JUNOS® Software

Interfaces and Routing Configuration Guide for J-series Services Routers and SRX-series Services Gateways

Release 9.4

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

Part Number: 530-027663-01, Revision 1

JUNOS Software Interfaces and Routing Configuration Guide
Release 9.4

Revision History
January 2009—Revision 1

The information in this document is current as of the date listed in the revision history.

YEAR 2000 NOTICE

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

SOFTWARE LICENSE

The terms and conditions for using this software are described in the software license contained in the acknowledgment to your purchase order or, to the extent applicable, to any reseller agreement or end-user purchase agreement executed between you and Juniper Networks. By using this software, you indicate that you understand and agree to be bound by those terms and conditions. Generally speaking, the software license restricts the manner in which you are permitted to use the software and may contain prohibitions against certain uses. The software license may state conditions under which the license is automatically terminated. You should consult the license for further details. For complete product documentation, please see the Juniper Networks Web site at www.juniper.net/techpubs.

END USER LICENSE AGREEMENT

READ THIS END USER LICENSE AGREEMENT ("AGREEMENT") BEFORE DOWNLOADING, INSTALLING, OR USING THE SOFTWARE. BY DOWNLOADING, INSTALLING, OR USING THE SOFTWARE OR OTHERWISE EXPRESSING YOUR AGREEMENT TO THE TERMS CONTAINED HEREIN, YOU (AS CUSTOMER OR IF YOU ARE NOT THE CUSTOMER, AS A REPRESENTATIVE/AGENT AUTHORIZED TO BIND THE CUSTOMER) CONSENT TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT OR CANNOT AGREE TO THE TERMS CONTAINED HEREIN, THEN (A) DO NOT DOWNLOAD, INSTALL, OR USE THE SOFTWARE, AND (B) YOU MAY CONTACT JUNIPER NETWORKS REGARDING LICENSE TERMS.

1. **The Parties.** The parties to this Agreement are (i) Juniper Networks, Inc. (if the Customer's principal office is located in the Americas) or Juniper Networks (Cayman) Limited (if the Customer's principal office is located outside the Americas) (such applicable entity being referred to herein as "Juniper"), and (ii) the person or organization that originally purchased from Juniper or an authorized Juniper reseller the applicable license(s) for use of the Software ("Customer") (collectively, the "Parties").

2. **The Software.** In this Agreement, "Software" means the program modules and features of the Juniper or Juniper-supplied software, for which Customer has paid the applicable license or support fees to Juniper or an authorized Juniper reseller, or which was embedded by Juniper in equipment which Customer purchased from Juniper or an authorized Juniper reseller. "Software" also includes updates, upgrades and new releases of such software. "Embedded Software" means Software which Juniper has embedded in or loaded onto the Juniper equipment and any updates, upgrades, additions or replacements which are subsequently embedded in or loaded onto the equipment.

3. **License Grant.** Subject to payment of the applicable fees and the limitations and restrictions set forth herein, Juniper grants to Customer a non-exclusive and non-transferable license, without right to sublicense, to use the Software, in executable form only, subject to the following use restrictions:

a. Customer shall use Embedded Software solely as embedded in, and for execution on, Juniper equipment originally purchased by Customer from Juniper or an authorized Juniper reseller.

b. Customer shall use the Software on a single hardware chassis having a single processing unit, or as many chassis or processing units for which Customer has paid the applicable license fees; provided, however, with respect to the Steel-Belted Radius or Odyssey Access Client software only, Customer shall use such Software on a single computer containing a single physical random access memory space and containing any number of processors. Use of the Steel-Belted Radius or IMS AAA software on multiple computers or virtual machines (e.g., Solaris zones) requires multiple licenses, regardless of whether such computers or virtualizations are physically contained on a single chassis.

c. Product purchase documents, paper or electronic user documentation, and/or the particular licenses purchased by Customer may specify limits to Customer's use of the Software. Such limits may restrict use to a maximum number of seats, registered endpoints, concurrent users, sessions, calls, connections, subscribers, clusters, nodes, realms, devices, links, ports or transactions, or require the purchase of separate licenses to use particular features, functionalities, services, applications, operations, or capabilities, or provide throughput, performance, configuration, bandwidth, interface, processing, temporal, or geographical limits. In addition, such limits may restrict the use of the Software to managing certain kinds of networks or require the Software to be used only in conjunction with other specific Software. Customer's use of the Software shall be subject to all such limitations and purchase of all applicable licenses.

d. For any trial copy of the Software, Customer's right to use the Software expires 30 days after download, installation or use of the Software. Customer may operate the Software after the 30-day trial period only if Customer pays for a license to do so. Customer may not extend or create an additional trial period by re-installing the Software after the 30-day trial period.

e. The Global Enterprise Edition of the Steel-Belted Radius software may be used by Customer only to manage access to Customer's enterprise network. Specifically, service provider customers are expressly prohibited from using the Global Enterprise Edition of the Steel-Belted Radius software to support any commercial network access services.

The foregoing license is not transferable or assignable by Customer. No license is granted herein to any user who did not originally purchase the applicable license(s) for the Software from Juniper or an authorized Juniper reseller.

4. **Use Prohibitions.** Notwithstanding the foregoing, the license provided herein does not permit the Customer to, and Customer agrees not to and shall not: (a) modify, unbundle, reverse engineer, or create derivative works based on the Software; (b) make unauthorized copies of the Software (except as necessary for backup purposes); (c) rent, sell, transfer, or grant any rights in and to any copy of the Software, in any form, to any third party; (d) remove any proprietary notices, labels, or marks on or in any copy of the Software or any product in which the Software is embedded; (e) distribute any copy of the Software to any third party, including as may be embedded in Juniper equipment sold in the secondhand market; (f) use any 'locked' or key-restricted feature, function, service, application, operation, or capability without first purchasing the applicable license(s) and obtaining a valid key from Juniper, even if such feature, function, service, application, operation, or capability is enabled without a key; (g) distribute any key for the Software provided by Juniper to any third party; (h) use the Software in any manner that extends or is broader than the uses purchased by Customer from Juniper or an authorized Juniper reseller; (i) use Embedded Software on non-Juniper equipment; (j) use Embedded Software (or make it available for use) on Juniper equipment that the Customer did not originally purchase from Juniper or an authorized Juniper reseller; (k) disclose the results of testing or benchmarking of the Software to any third party without the prior written consent of Juniper; or (l) use the Software in any manner other than as expressly provided herein.

5. **Audit.** Customer shall maintain accurate records as necessary to verify compliance with this Agreement. Upon request by Juniper, Customer shall furnish such records to Juniper and certify its compliance with this Agreement.

6. **Confidentiality.** The Parties agree that aspects of the Software and associated documentation are the confidential property of Juniper. As such, Customer shall exercise all reasonable commercial efforts to maintain the Software and associated documentation in confidence, which at a minimum includes restricting access to the Software to Customer employees and contractors having a need to use the Software for Customer's internal business purposes.

7. **Ownership.** Juniper and Juniper's licensors, respectively, retain ownership of all right, title, and interest (including copyright) in and to the Software, associated documentation, and all copies of the Software. Nothing in this Agreement constitutes a transfer or conveyance of any right, title, or interest in the Software or associated documentation, or a sale of the Software, associated documentation, or copies of the Software.

8. **Warranty, Limitation of Liability, Disclaimer of Warranty.** The warranty applicable to the Software shall be as set forth in the warranty statement that accompanies the Software (the "Warranty Statement"). Nothing in this Agreement shall give rise to any obligation to support the Software. Support services may be purchased separately. Any such support shall be governed by a separate, written support services agreement. TO THE MAXIMUM EXTENT PERMITTED BY LAW, JUNIPER SHALL NOT BE LIABLE FOR ANY LOST PROFITS, LOSS OF DATA, OR COSTS OR PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THIS AGREEMENT, THE SOFTWARE, OR ANY JUNIPER OR JUNIPER-SUPPLIED SOFTWARE. IN NO EVENT SHALL JUNIPER BE LIABLE FOR DAMAGES ARISING FROM UNAUTHORIZED OR IMPROPER USE OF ANY JUNIPER OR JUNIPER-SUPPLIED SOFTWARE. EXCEPT AS EXPRESSLY PROVIDED IN THE WARRANTY STATEMENT TO THE EXTENT PERMITTED BY LAW, JUNIPER DISCLAIMS ANY AND ALL WARRANTIES IN AND TO THE SOFTWARE (WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE), INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT DOES JUNIPER WARRANT THAT THE SOFTWARE, OR ANY EQUIPMENT OR NETWORK RUNNING THE SOFTWARE, WILL OPERATE WITHOUT ERROR OR INTERRUPTION, OR WILL BE FREE OF VULNERABILITY TO INTRUSION OR ATTACK. In no event shall Juniper's or its suppliers' or licensors' liability to Customer, whether in contract, tort (including negligence), breach of warranty, or otherwise, exceed the price paid by Customer for the Software that gave rise to the claim, or if the Software is embedded in another Juniper product, the price paid by Customer for such other product. Customer acknowledges and agrees that Juniper has set its prices and entered into this Agreement in reliance upon the disclaimers of warranty and the limitations of liability set forth herein, that the same reflect an allocation of risk between the Parties (including the risk that a contract remedy may fail of its essential purpose and cause consequential loss), and that the same form an essential basis of the bargain between the Parties.

9. **Termination.** Any breach of this Agreement or failure by Customer to pay any applicable fees due shall result in automatic termination of the license granted herein. Upon such termination, Customer shall destroy or return to Juniper all copies of the Software and related documentation in Customer's possession or control.

10. **Taxes.** All license fees payable under this agreement are exclusive of tax. Customer shall be responsible for paying Taxes arising from the purchase of the license, or importation or use of the Software. If applicable, valid exemption documentation for each taxing jurisdiction shall be provided to Juniper prior to invoicing, and Customer shall promptly notify Juniper if their exemption is revoked or modified. All payments made by Customer shall be net of any applicable withholding tax. Customer will provide reasonable assistance to Juniper in connection with such withholding taxes by promptly: providing Juniper with valid tax receipts and other required documentation showing Customer's payment of any withholding taxes; completing appropriate applications that would reduce the amount of withholding tax to be paid; and notifying and assisting Juniper in any audit or tax proceeding related to transactions hereunder. Customer shall comply with all applicable tax laws and regulations, and Customer will promptly pay or reimburse Juniper for all costs and damages related to any liability incurred by Juniper as a result of Customer's non-compliance or delay with its responsibilities herein. Customer's obligations under this Section shall survive termination or expiration of this Agreement.

11. **Export.** Customer agrees to comply with all applicable export laws and restrictions and regulations of any United States and any applicable foreign agency or authority, and not to export or re-export the Software or any direct product thereof in violation of any such restrictions, laws or regulations, or without all necessary approvals. Customer shall be liable for any such violations. The version of the Software supplied to Customer may contain encryption or other capabilities restricting Customer's ability to export the Software without an export license.

12. **Commercial Computer Software.** The Software is "commercial computer software" and is provided with restricted rights. Use, duplication, or disclosure by the United States government is subject to restrictions set forth in this Agreement and as provided in DFARS 227.7201 through 227.7202-4, FAR 12.212, FAR 27.405(b)(2), FAR 52.227-19, or FAR 52.227-14(ALT III) as applicable.

13. **Interface Information.** To the extent required by applicable law, and at Customer's written request, Juniper shall provide Customer with the interface information needed to achieve interoperability between the Software and another independently created program, on payment of applicable fee, if any. Customer shall observe strict obligations of confidentiality with respect to such information and shall use such information in compliance with any applicable terms and conditions upon which Juniper makes such information available.

14. **Third Party Software.** Any licensor of Juniper whose software is embedded in the Software and any supplier of Juniper whose products or technology are embedded in (or services are accessed by) the Software shall be a third party beneficiary with respect to this Agreement, and such licensor or vendor shall have the right to enforce this Agreement in its own name as if it were Juniper. In addition, certain third party software may be provided with the Software and is subject to the accompanying license(s), if any, of its respective owner(s). To the extent portions of the Software are distributed under and subject to open source licenses obligating Juniper to make the source code for such portions publicly available (such as the GNU General Public License ("GPL") or the GNU Library General Public License ("LGPL")), Juniper will make such source code portions (including Juniper modifications, as appropriate) available upon request for a period of up to three years from the date of distribution. Such request can be made in writing to Juniper Networks, Inc., 1194 N. Mathilda Ave., Sunnyvale, CA 94089, ATTN: General Counsel. You may obtain a copy of the GPL at http://www.gnu.org/licenses/gpl.html, and a copy of the LGPL at http://www.gnu.org/licenses/lgpl.html.

15. **Miscellaneous.** This Agreement shall be governed by the laws of the State of California without reference to its conflicts of laws principles. The provisions of the U.N. Convention for the International Sale of Goods shall not apply to this Agreement. For any disputes arising under this Agreement, the Parties hereby consent to the personal and exclusive jurisdiction of, and venue in, the state and federal courts within Santa Clara County, California. This Agreement constitutes the entire and sole agreement between Juniper and the Customer with respect to the Software, and supersedes all prior and contemporaneous

agreements relating to the Software, whether oral or written (including any inconsistent terms contained in a purchase order), except that the terms of a separate written agreement executed by an authorized Juniper representative and Customer shall govern to the extent such terms are inconsistent or conflict with terms contained herein. No modification to this Agreement nor any waiver of any rights hereunder shall be effective unless expressly assented to in writing by the party to be charged. If any portion of this Agreement is held invalid, the Parties agree that such invalidity shall not affect the validity of the remainder of this Agreement. This Agreement and associated documentation has been written in the English language, and the Parties agree that the English version will govern. (For Canada: Les parties aux présentés confirment leur volonté que cette convention de même que tous les documents y compris tout avis qui s'y rattaché, soient redigés en langue anglaise. (Translation: The parties confirm that this Agreement and all related documentation is and will be in the English language)).

Abbreviated Table of Contents

Table of Contents

Part 4 Configuring Private Communications over Public Networks with MPLS

Part 7 Index

About This Guide

This preface provides the following guidelines for using the *JUNOS Software Interfaces and Routing Configuration Guide*:

Objectives

This guide contains instructions for configuring the J-series and SRX-series interfaces for basic IP routing with standard routing protocols. It also shows how to create backup ISDN interfaces, configure digital subscriber line (DSL) connections and link services, create stateless firewall filters—also known as access control lists (ACLs)—and configure class-of-service (CoS) traffic classification.

NOTE: This manual documents Release 9.4 of JUNOS software. For additional information—either corrections to or information that might have been omitted from this manual—see the *JUNOS Software Release Notes* at http://www.juniper.net.

Audience

This manual is designed for anyone who installs, sets up, configures, monitors, or administers a J-series Services Router or an SRX-series services gateway running JUNOS software. The manual is intended for the following audiences:

- Customers with technical knowledge of and experience with networks and network security, the Internet, and Internet routing protocols
- Network administrators who install, configure, and manage Internet routers

Supported Routing Platforms

This manual describes features supported on J-series Services Routers and SRX-series services gateways running JUNOS software.

How to Use This Manual

This manual and the other manuals in this set explain how to install, configure, and manage:

- JUNOS software for J-series Services Routers
- JUNOS software for SRX-series services gateways

Table 1 on page xxx identifies the tasks required to configure and manage these devices and shows where to find task information and instructions.

For an annotated list of the documentation referred to in Table 1 on page xxx, see "List of Technical Publications" on page xxxiii. All documents are available at http://www.juniper.net/techpubs/.

Table 1: Tasks and Related Documentation

Task	Related Documentation
Basic Device Installation and Setup	
■ Reviewing safety warnings and compliance statements ■ Installing hardware and establishing basic connectivity ■ Initially setting up a device	J-series Services Routers: ■ *J-series Services Routers Quick Start* ■ *J-series Services Routers Hardware Guide* ■ *JUNOS Software Release Notes* SRX-series services gateways: the appropriate *Services Gateway Getting Started Guide*
Migration from ScreenOS or JUNOS Software (Legacy Services) to JUNOS Software (if necessary)	
■ Migrating from JUNOS software (legacy services) Release 8.3 or later to JUNOS software ■ Migrating from ScreenOS Release 5.4 or later to JUNOS software.	*JUNOS Software Migration Guide* (J-series Services Routers only)
Context—Changing to Secure Context or Router Context	
Changing the device from one context to another and understanding the factory default settings	*JUNOS Software Administration Guide*
Interface Configuration	
Configuring device interfaces	■ *JUNOS Software Interfaces and Routing Configuration Guide* ■ *JUNOS Software CLI Reference*
Deployment Planning and Configuration	

Table 1: Tasks and Related Documentation *(continued)*

Task	Related Documentation
■ Understanding and gathering information required to design network firewalls and IPsec VPNs ■ Implementing a JUNOS software firewall from a sample scenario ■ Implementing a policy-based IPsec VPN from a sample scenario	*JUNOS Software Design and Implementation Guide* (J-series Services Routers only)
Security Configuration	
Configuring and managing the following security services: ■ Stateful firewall policies ■ Zones and their interfaces and address books ■ IPsec VPNs ■ Firewall screens ■ Interface modes: Network Address Translation (NAT) mode and Router mode ■ Public Key Cryptography (PKI) ■ Application Layer Gateways (ALGs) ■ Chassis clusters ■ Intrusion Detection and Prevention (IDP)	■ *JUNOS Software Security Configuration Guide* ■ *JUNOS Software CLI Reference*
Routing Protocols and Services Configuration	
■ Configuring routing protocols, including static routes and the dynamic routing protocols RIP, OSPF, BGP, and IS-IS ■ Configuring class-of-service (CoS) features, including traffic shaping and policing ■ Configuring packet-based stateless firewall filters (access control lists) to control access and limit traffic rates ■ Configuring MPLS to control network traffic patterns	■ *JUNOS Software Interfaces and Routing Configuration Guide* ■ *JUNOS Software CLI Reference*
WAN Acceleration Module Installation (Optional)	
Installing and initially configuring a WXC Integrated Services Module (ISM 200)	*WXC Integrated Services Module Installation and Configuration Guide* (J-series Services Routers only)
User and System Administration	
■ Administering user authentication and access ■ Monitoring the device, routing protocols, and routing operations ■ Configuring and monitoring system alarms and events, real-time performance (RPM) probes, and performance ■ Monitoring the firewall and other security-related services ■ Managing system log files ■ Upgrading software ■ Diagnosing common problems	*JUNOS Software Administration Guide*
User Interfaces	

Table 1: Tasks and Related Documentation *(continued)*

Task	Related Documentation
■ Understanding and using the J-Web interface ■ Understanding and using the CLI configuration editor	■ *J-series Services Routers Quick Start* (J-series Services Routers only) ■ *JUNOS Software Administration Guide*

Document Conventions

Table 2 on page xxxii defines the notice icons used in this guide.

Table 2: 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.

Table 3 on page xxxii defines the text and syntax conventions used in this guide.

Table 3: 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`
Italic text like this	■ Introduces important new terms. ■ Identifies book names. ■ Identifies RFC and Internet draft titles.	■ A policy *term* is a named structure that defines match conditions and actions. ■ *JUNOS System Basics Configuration Guide* ■ RFC 1997, *BGP Communities Attribute*

Table 3: Text and Syntax Conventions *(continued)*

Convention	Description	Examples
Italic text like this	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** *domain-name*
Plain text like this	Represents names of configuration statements, commands, files, and directories; IP addresses; configuration hierarchy levels; or labels on routing platform components.	■ 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)	Enclose 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)	Enclose a variable for which you can substitute one or more values.	community name members [*community-ids*]

Indention and braces (**{ }**)	Identify a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  retain;
		  }
		  }
		}

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
J-Web GUI Conventions		
Bold text like this	Represents J-Web graphical user interface (GUI) items you click or select.	■ In the Logical Interfaces box, select **All Interfaces**. ■ To cancel the configuration, click **Cancel**.
> (bold right angle bracket)	Separates levels in a hierarchy of J-Web selections.	In the configuration editor hierarchy, select **Protocols > Ospf**.

List of Technical Publications

The following sections list hardware and software guides and release notes for SRX-series services gateways and J-series Services Routers running JUNOS software.

All documents are available at http://www.juniper.net/techpubs/.

Hardware Guides

- *SRX 3400 Services Gateway Hardware Guide*—Describes hardware components, installation, basic configuration, and basic troubleshooting procedures for the SRX 3400 services gateway. This guide explains how to prepare a site, unpack and install the device, replace device hardware, establish basic connectivity, and perform routine maintenance.
- *SRX 3600 Services Gateway Hardware Guide*—Describes hardware components, installation, basic configuration, and basic troubleshooting procedures for the SRX 3600 services gateway. This guide explains how to prepare a site, unpack and install the device, replace device hardware, establish basic connectivity, and perform routine maintenance.
- *SRX 5600 Services Gateway Hardware Guide*—Describes hardware components, installation, basic configuration, and basic troubleshooting procedures for the SRX 5600 services gateway. This guide explains how to prepare a site, unpack and install the device, replace device hardware, establish basic connectivity, and perform routine maintenance.
- *SRX 5800 Services Gateway Hardware Guide*—Describes hardware components, installation, basic configuration, and basic troubleshooting procedures for the SRX 5800 services gateway. This guide explains how to prepare a site, unpack and install the device, replace device hardware, establish basic connectivity, and perform routine maintenance.
- *J-series Services Routers Quick Start*—Explains how to quickly set up a J-series Services Router. This document contains router declarations of conformity.
- *J-series Services Routers Hardware Guide*—Provides an overview, basic instructions, and specifications for J-series Services Routers. This guide explains how to prepare a site, unpack and install the router, replace router hardware, and establish basic router connectivity. This guide contains hardware descriptions and specifications.

Software Guides

- *JUNOS Software Interfaces and Routing Configuration Guide*—Explains how to configure SRX-series and J-series interfaces for basic IP routing with standard routing protocols, ISDN service, firewall filters (access control lists), and class-of-service (CoS) traffic classification.
- *JUNOS Software Security Configuration Guide*—Explains how to configure and manage SRX-series and J-series security services such as stateful firewall policies, IPsec VPNs, firewall screens, Network Address Translation (NAT), Public Key Cryptography, chassis clusters, Application Layer Gateways (ALGs), and Intrusion Detection and Prevention (IDP).
- *JUNOS Software Administration Guide*—Shows how to monitor SRX-series and J-series devices and routing operations, firewall and security services, system alarms and events, and network performance. This guide also shows how to administer user authentication and access, upgrade software, and diagnose common problems.
- *JUNOS Software CLI Reference*—Provides the complete configuration hierarchy available on SRX-series and J-series devices. This guide also describes the configuration statements and operational mode commands unique to these devices.

- *JUNOS Network Management Configuration Guide*—Describes enterprise-specific MIBs for JUNOS software. The information in this guide is applicable to M-series, T-series, EX-series, SRX-series, and J-series devices.
- *JUNOS System Log Messages Reference*—Describes how to access and interpret system log messages generated by JUNOS software modules and provides a reference page for each message. The information in this guide is applicable to M-series, T-series, EX-series, SRX-series, and J-series devices.
- *JUNOS Software Design and Implementation Guide*—Provides guidelines and examples for designing and implementing IPsec VPNs, firewalls, and routing on J-series Services Routers running JUNOS software.
- *JUNOS Software Migration Guide*—Provides instructions for migrating an SSG device running ScreenOS software to JUNOS software or upgrading a J-series Services Router to a later version of JUNOS software.
- *WXC Integrated Services Module Installation and Configuration Guide*—Explains how to install and initially configure a WXC Integrated Services Module in a J-series Services Router for application acceleration.

Release Notes

- *JUNOS Software Release Notes*—Summarize new features and known problems for a particular release of JUNOS software, including JUNOS software for J-series and SRX-series devices.The release notes also contain corrections and updates to the manuals and software upgrade and downgrade instructions for JUNOS software.

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 name
- Document part number
- Page number
- Software release version (not required for *Network Operations Guides [NOGs])*

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 postsales 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/customers/support/downloads/710059.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: https://www.juniper.net/alerts/
- 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 located at 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, visit us at http://www.juniper.net/support/requesting-support.html.

Part 1
Support Overview for Interface and Routing Features

Chapter 1

Interface and Routing Features on SRX 3400/3600/5600/5800 Services Gateways

The following tables list interface and routing features that are supported on SRX 3400, SRX 3600, SRX 5600, and SRX 5800 services gateways.

Table 4: Support Information: Interfaces

Feature	More Information
Ethernet interface	"Ethernet Interface Overview" on page 24
Gigabit Ethernet interface	"Interfaces Terms" on page 11
Loopback Interface	"Loopback Interface" on page 70
Management interface	"Management Interface" on page 70

Table 5: Support Information: Routing Options

Feature	More Information
IPv4 options and broadcast Internet diagrams	"Routing Overview" on page 299
Static routing	"Static Routing Overview" on page 333
RIP v1, v2	"RIP Overview" on page 310
OSPF v2	"OSPF Overview" on page 315
BGP	"BGP Overview" on page 322
Multiple virtual routers	*JUNOS Routing Protocols Configuration Guide*
Network Time Protocol (NTP)	*JUNOS System Basics and Services Command Reference*
Virtual Router Redundancy Protocol (VRRP)	*JUNOS Software Network Interfaces Configuration Guide*

Table 6: Support Information: MPLS

Feature	More Information
Equal-cost multipath (ECMP)	*JUNOS Software MPLS Applications Configuration Guide*
Filter-based forwarding (FBF) and forwarding table filters (FTFs)	*JUNOS Software VPNs Configuration Guide*

Table 7: Stateless Firewall Filters

Feature	More Information
Stateless firewall filters (ACLs)	"Stateless Firewall Filter Overview" on page 522

Table 8: Support Information: CoS

Feature	More Information
Code-point aliases	"Code-Point Aliases" on page 557
Classifiers	"Classifiers" on page 557
Forwarding classes	"Forwarding Classes" on page 560
Transmission queues (SRX 5600 and SRX 5800 only)	"Transmission Queues" on page 561
Schedulers: ■ Transmission rate ■ Delay buffer size ■ Shaping rate ■ Red drop profiles	"Schedulers" on page 561 **NOTE:** For hardware differences that affect scheduling and shaping in the SRX 3400 and SRX 3600 series devices, see "SRX 3400 and SRX 3600 Hardware Capabilities and Limitations" on page 654
Virtual channels (SRX 5600 and SRX 5800 only)	"Virtual Channels" on page 564
Tunnels (SRX 5600 and SRX 5800 only)	"CoS Queuing for Tunnels" on page 576
Policing (SRX 5600 and SRX 5800 only)	"Policers for Traffic Classes" on page 565

Chapter 2

Interface and Routing Features on J-series Services Routers

The following tables list interface and routing features that are supported on J-series Services Routers.

Table 9: Support Information: Interfaces

Feature	More Information
Asymmetric digital subscriber line (ADSL) interface	“ADSL Interface Overview” on page 43
Channelized E1 interface	“Channelized T1/E1/ISDN PRI Interfaces Overview” on page 31
Channelized ISDN PRI interface	“Channelized T1/E1/ISDN PRI Interfaces Overview” on page 31
Channelized T1 interface	“Channelized T1/E1/ISDN PRI Interfaces Overview” on page 31
Class-of-service support interface	“Special Interfaces” on page 67
Discard interface	“Discard Interface” on page 70
Ethernet interface	“Ethernet Interface Overview” on page 24
E1 interface	“E1 Overview” on page 28
E3 interface	“T3 and E3 Interfaces Overview” on page 32
Fast Ethernet interface	“Interfaces Terms” on page 11
Generic routing encapsulation (GRE) interface	“Special Interfaces” on page 67
Gigabit Ethernet interface	“Interfaces Terms” on page 11
Internally configured interface used by the system as a control path between the WXC Integrated Services Module and the Routing Engine	“Special Interfaces” on page 67
Internally generated GRE interface	“Special Interfaces” on page 67
Internally generated link services interface	“Special Interfaces” on page 67
Internally generated IP-over-IP interface	“Special Interfaces” on page 67

Table 9: Support Information: Interfaces *(continued)*

Feature	More Information
Internally generated Protocol Independent Multicast de-encapsulation interface	"Special Interfaces" on page 67
Internally generated Protocol Independent Multicast encapsulation interface	"Special Interfaces" on page 67
IP-over-IP encapsulation interface	"Special Interfaces" on page 67
ISDN interface	"ISDN Interface Overview" on page 45
Link services interface	"Services Interfaces" on page 71
Loopback Interface	"Loopback Interface" on page 70
Management interface	"Management Interface" on page 70
Passive monitoring interface	"Special Interfaces" on page 67
Point-to-Point Protocol over Ethernet (PPPoE) interface	"Configuring Point-to-Point Protocol over Ethernet" on page 157
Protocol Independent Multicast de-encapsulation interface	"Special Interfaces" on page 67
Protocol Independent Multicast encapsulation interface	"Special Interfaces" on page 67
Secure tunnel interface	"Special Interfaces" on page 67
Serial interface	"Serial Interface Overview" on page 37
Symmetric high-speed DSL (SHDSL) interface	"SHDSL Interface Overview" on page 44
T1 interface	"T1 Overview" on page 28
T3 interface	"T3 and E3 Interfaces Overview" on page 32
Universal serial bus (USB) model physical interface	"Special Interfaces" on page 67

Table 10: Support Information: Routing Options

Feature	More Information
IPv4 options and broadcast Internet diagrams	"Routing Overview" on page 299
IPv6 routing, forwarding, global address configuration, and Internet Control Message Protocol (ICMP)	"Routing Overview" on page 299
Static routing	"Static Routing Overview" on page 333
RIP v1, v2	"RIP Overview" on page 310
RIP next generation (RIPng)	"RIPng Overview" on page 314
OSPF v2	"OSPF Overview" on page 315

Table 10: Support Information: Routing Options *(continued)*

Feature	More Information
OSPF v3	"OSPF Overview" on page 315
IS-IS	"IS-IS Overview" on page 320
BGP	"BGP Overview" on page 322
BGP extensions for IPv6	"BGP Overview" on page 322
Neighbor Discovery Protocol and Secure Neighbor Discovery Protocol	*JUNOS Routing Protocols Configuration Guide*
Multiple virtual routers	*JUNOS Routing Protocols Configuration Guide*
Internet Group Management Protocol (IGMP)	*JUNOS Multicast Protocols Configuration Guide*
Protocol Independent Multicast (PIM)	*JUNOS Multicast Protocols Configuration Guide*
Distance Vector Multicast Routing Protocol (DVMRP)	*JUNOS Multicast Protocols Configuration Guide*
Single-source multicast	*JUNOS Multicast Protocols Configuration Guide*
Multicast Source Discovery Protocol (MSDP)	*JUNOS Multicast Protocols Configuration Guide*
Session Announcement Protocol (SAP) and Session Description	*JUNOS Multicast Protocols Configuration Guide*
Network Time Protocol (NTP)	*JUNOS System Basics and Services Command Reference*
Compressed Real-Time Transport Protocol (CRTP)	*JUNOS System Basics and Services Command Reference*
Virtual Router Redundancy Protocol (VRRP)	*JUNOS Software Network Interfaces Configuration Guide*

Table 11: Support Information: MPLS

Feature	More Information
Secondary and standby label-switched paths (LSPs)	"MPLS Overview" on page 408
Point-to-multipoint connections	"MPLS Overview" on page 408
MPLS virtual private networks (VPNs) with VPN routing and forwarding (VRF) tables on customer edge (CE) routers	"VPN Overview" on page 418
Layer 3 MPLS VPNs	"VPN Overview" on page 418
Layer 2 VPNs for Ethernet connections	"VPN Overview" on page 418
Circuit cross-connect (CCC) and translational cross-connect (TCC)	"VPN Configuration Overview" on page 439
LDP	"Signaling Protocols Overview" on page 414
RSVP	"Signaling Protocols Overview" on page 414

Table 11: Support Information: MPLS *(continued)*

Feature	More Information
Connectionless Network Service (CLNS)	"CLNS Overview" on page 464
Virtual private LAN service (VPLS)	"VPLS Overview" on page 475
OSPF and IS-IS traffic engineering extensions	*JUNOS Software MPLS Applications Configuration Guide*
Equal-cost multipath (ECMP)	*JUNOS Software MPLS Applications Configuration Guide*
Interprovider and carrier-of-carriers VPNs	*JUNOS Software VPNs Configuration Guide*
Standards-based fast reroute	*JUNOS Software VPNs Configuration Guide*
Filter-based forwarding (FBF) and forwarding table filters (FTFs)	*JUNOS Software VPNs Configuration Guide*
Multicast VPNs	*JUNOS Software VPNs Configuration Guide*

Table 12: Stateless Firewall Filters

Feature	More Information
Stateless firewall filters (ACLs)	"Stateless Firewall Filter Overview" on page 522

Table 13: Support Information: CoS

Feature	More Information
Code-point aliases	"Code-Point Aliases" on page 557
Classifiers	"Classifiers" on page 557
Forwarding classes	"Forwarding Classes" on page 560
Transmission queues	"Transmission Queues" on page 561
Schedulers: ■ Transmission rate (no exact knob rate) ■ Delay buffer size ■ Shaping rate ■ Red drop profiles	"Schedulers" on page 561
Virtual channels	"Virtual Channels" on page 564
Tunnels	"CoS Queuing for Tunnels" on page 576
Policing	"Policers for Traffic Classes" on page 565

Part 2
Configuring Router Interfaces

Chapter 3

Interfaces Overview

J-series Services Routers and SRX-series service gateways support a variety of interface types, as explained in Table 4 on page 3 and Table 9 on page 5.

To configure and monitor J-series or SRX-series device interfaces, you need to understand their media characteristics, as well as physical and logical properties such as IP addressing, link-layer protocols, and link encapsulation.

This chapter contains the following topics. For more information about interfaces, see the *JUNOS Network Interfaces Configuration Guide*, the *JUNOS Services Interfaces Configuration Guide*, and the *JUNOS Interfaces Command Reference*.

Interfaces Terms

To understand interfaces, become familiar with the terms defined in Table 14 on page 12.

Table 14: Network Interfaces Terms

Term	Definition
alternate mark inversion (AMI)	Original method of formatting T1 and E1 data streams.
asymmetric digital subscriber line (ADSL) interface	Physical WAN interface for connecting a J-series device to a digital subscriber line access multiplexer (DSLAM). An ADSL interface allocates line bandwidth asymmetrically with downstream (provider-to-customer) data rates of up to 8 Mbps for ADSL, 12 Mbps for ADSL2, and 25 Mbps for ADSL2+, and upstream (customer-to-provider) rates of up to 800 Kbps for ADSL and 1 Mbps for ADSL2 and ADSL2+, depending on the implementation.
ADSL2 interface	An ADSL interface that supports ITU-T Standards G.992.3 and G.992.4 and allocates downstream (provider-to-customer) data rates of up to 12 Mbps and upstream (customer-to-provider) rates of up to 1 Mbps.
ADSL2+ interface	An ADSL interface that supports ITU-T Standard G.992.5 and allocates downstream (provider-to-customer) data rates of up to 25 Mbps and upstream (customer-to-provider) rates of up to 1 Mbps.
Annex A	ITU-T Standard G.992.1 that defines how ADSL works over plain old telephone service (POTS) lines.
Annex B	ITU-T Standard G.992.1 that defines how ADSL works over Integrated Services Digital Network (ISDN) lines.
binary 8-zero substitution (B8ZS)	Improved method of formatting T1 and E1 data streams, in which a special code is substituted whenever 8 consecutive zeros are sent over the link.
Challenge Handshake Authentication Protocol (CHAP)	Protocol that authenticates remote users. CHAP is a server-driven, three-step authentication method that depends on a shared secret password residing on both the server and the client.
checksum	See *frame checksum sequence*.
channel group	Combination of DS0 interfaces partitioned from a channelized interface into a single logical bundle.
channel service unit (CSU)	Unit that connects a digital telephone line to a multiplexer or other signal service.
channelized E1	2.048-Mbps interface that can be configured as a single clear-channel E1 interface or channelized into as many as 31 discrete DS0 interfaces, or up to 30 ISDN PRI B-channels and 1 D-channel. On J-series channelized T1/E1/ISDN PRI interfaces, time slots are numbered from 1 through 31, and time slot 1 is reserved for framing. When the interface is configured for ISDN PRI service, time slot 16 is reserved for the D-channel.
channelized interface	Interface that is a subdivision of a larger interface, minimizing the number of Physical Interface Modules (PIMs) that an installation requires. On a channelized PIM, each port can be configured as a single clear channel or partitioned into multiple discrete T1, E1, and DS0 interfaces.
channelized T1	1.544-Mbps interface that can be configured as a single clear-channel T1 interface or channelized into as many as 24 discrete DS0 interfaces, or up to 23 ISDN PRI B-channels and 1 D-channel. When the interface is configured for ISDN PRI service, time slot 24 is reserved for the D-channel.

Table 14: Network Interfaces Terms *(continued)*

Term	Definition
Cisco HDLC	Cisco High-level Data Link Control protocol. Proprietary Cisco encapsulation for transmitting LAN protocols over a WAN. HDLC specifies a data encapsulation method on synchronous serial links by means of frame characters and checksums. Cisco HDLC enables the transmission of multiple protocols.
clock source	Source of the consistent, periodic signal used by a device to synchronize data communication and processing tasks.
CSU compatibility mode	Subrate on an E3 or T3 interface that allows a J-series device to connect to a channel service unit (CSU) with proprietary multiplexing at the remote end of the line. Subrating an E3 or T3 interface reduces the maximum allowable peak rate by limiting the payload encapsulated by the High-level Data Link Control protocol (HDLC).
data-link connection identifier (DLCI)	Identifier for a Frame Relay virtual connection, also called a logical interface.
data service unit (DSU)	Unit that connects a data terminal equipment (DTE) device—in this case, a J-series Services Router or an SRX–series services gateway— to a digital telephone line.
data terminal equipment (DTE)	RS-232 interface that a Juniper Networks device uses to exchange information with a serial device.
DS1	Digital signal 1, another name for a T1 interface.
DS3 interface	Digital signal 3, another name for a T3 interface.
data inversion	Transmission of all data bits in the data stream so that zeros are transmitted as ones and ones are transmitted as zeros. Data inversion is normally used only in alternate mark inversion (AMI) mode to guarantee ones density in the transmitted stream.
E1 interface	Physical WAN interface for transmitting signals in European digital transmission (E1) format. The E1 signal format carries information at a rate of 2.048 Mbps and can carry 32 channels of 64 Kbps each.
E3 interface	Physical WAN interface for transmitting 16 E1 circuits over copper wires using time-division multiplexing. E3 is widely used outside of North America and transfers traffic at the rate of 34.368 Mbps.
encapsulation type	Type of protocol header in which data is wrapped for transmission.
Fast Ethernet interface	Physical LAN interface for transmitting data at 100 Mbps. Fast Ethernet, also called 100Base-T, additionally supports standard 10Base-T Ethernet transmission. Fast Ethernet is available on both dual-port and 4-port PIMs for the J4350 and J6350 devices.
FPC	Logical identifier for a Physical Interface Module (PIM) installed on a J-series device. The FPC number used in the JUNOS command-line interface (CLI) and displayed in command output represents the chassis slot in which a PIM is installed.
fractional E1	Interface that contains one or more of the 32 DS0 time slots that can be reserved from an E1 interface. (Time slot 0 is reserved.)
fractional T1	Interface that contains one or more of the 24 DS0 time slots that can be reserved from a T1 interface. (Time slot 0 is reserved.)

Table 14: Network Interfaces Terms *(continued)*

Term	Definition
frame check sequence (FCS)	Calculation that is added to a frame to control errors in High-level Data Link Control (HDLC), Frame Relay, and other data link layer protocols.
Frame Relay	An efficient WAN protocol that does not require explicit acknowledgement of each frame of data. Frame Relay allows private networks to reduce costs by sharing facilities between the endpoint switches of a network managed by a Frame Relay service provider. Individual data link connection identifiers (DLCIs) are assigned to ensure that customers receive only their own traffic.
Gigabit Ethernet interface	Physical LAN or WAN interface for transmitting data at 1000 Mbps. The four built-in ports on J4350 and J6350 devices are Gigabit Ethernet interfaces. Gigabit Ethernet is also available in a single-port copper or optical PIM for these devices. Gigabit Ethernet is also supported in SRX-series services gateways.
High-Level Data Link Control (HDLC)	International Telecommunication Union (ITU) standard for a bit-oriented data link layer protocol on which most other bit-oriented protocols are based.
hostname	Name assigned to the device during initial configuration.
ITU-T G.991.2	International Telecommunication Union standard describing a data transmission method for symmetric high-speed digital subscriber line (SHDSL) as a means for data transport in telecommunications access networks. The standard also describes the functionality required for interoperability of equipment from various manufacturers.
ITU-T G.992.1	International Telecommunication Union standard that requires the downstream (provider-to-customer) data transmission to consist of full-duplex low-speed bearer channels and simplex high-speed bearer channels. In the upstream (customer-to-provider) transmissions, only low-speed bearer channels are provided.
ITU-T G.994.1	International Telecommunication Union standard describing the types of signals, messages, and procedures exchanged between digital subscriber line (DSL) equipment when the operational modes of equipment need to be automatically established and selected.
ITU-T G.997.1	International Telecommunication Union standard describing the physical layer management for asymmetric digital subscriber line (ADSL) transmission systems. The standard specifies the means of communication on a transport transmission channel defined in the physical layer recommendations. In addition, the standard describes the content and syntax of network elements for configuration, fault management, and performance management.
logical interface	Virtual interface that you create on a physical interface to identify its connection. Creating multiple logical interfaces allows you to associate multiple virtual circuits, data line connections, or virtual LANs (VLANs) with a single interface device.
maximum transmission unit (MTU)	Maximum or largest segment size that a network can transmit.
Multilink Frame Relay (MLFR)	Protocol that allows multiple Frame Relay links to be aggregated by inverse multiplexing.
Multilink Point-to-Point Protocol (MLPPP)	Protocol that allows you to bundle multiple Point-to-Point Protocol (PPP) links into a single logical unit. MLPPP improves bandwidth efficiency and fault tolerance and reduces latency.

Table 14: Network Interfaces Terms *(continued)*

Term	Definition
Password Authentication Protocol (PAP)	Authentication protocol that uses a simple 2-way handshake to establish identity.
Physical Interface Module (PIM)	Network interface card that is fixed or can be interchangeably installed on a J-series device to provide the physical connections to a LAN or WAN, receiving incoming packets and transmitting outgoing packets. A PIM contains *one* of the following interfaces or sets of interfaces: ■ Single Gigabit Ethernet LAN or WAN interface ■ Two or four Fast Ethernet LAN interfaces ■ Two T1 or two E1 WAN interfaces ■ Single E3 or T3 (DS3) WAN interface (J4350 and J6350 models only) ■ Single asynchronous digital subscriber line (ADSL) WAN interface—Annex A to support ADSL over plain old telephone service (POTS) lines or Annex B to support ADSL over ISDN ■ Four ISDN BRI S/T or U interfaces ■ Two channelized T1/E1/ISDN PRI interfaces ■ Two serial interfaces ■ Symmetric high-speed digital subscriber line (SHDSL) WAN interface—Annex A or Annex B to support ATM-over-SHDSL connections
Point-to-Point Protocol (PPP)	Link-layer protocol that provides multiprotocol encapsulation. PPP is used for link-layer and network-layer configuration.
serial interface	Physical LAN interface for transmitting data between computing devices. A J-series device has two types of serial interfaces: ■ Asynchronous serial interface—Console port, with speeds up to 110.5 Kbps. The console port supports an RS-232 (EIA-232) standard serial cable with a 25-pin (DB-25) connector. ■ Synchronous serial interface—Port that transmits packets to and from, for example, a T1 device or microwave link, at speeds up to 8 Mbps. You cannot use this serial interface to connect a console. J-series device synchronous serial interfaces support RS-232 (EIA-232), RS-422/449 (EIA-449), RS-530 (EIA-530), V.35, and X.21 cable types. For details, see “Serial Line Protocols” on page 40. For cable details, see the *J-series Services Routers Hardware Guide*.
symmetric high-speed digital subscriber line (G.SHDSL)	Physical WAN symmetric DSL interface capable of sending and receiving high-speed symmetrical data streams over a single pair of copper wires at rates between 192 Kbps and 2.31 Mbps. G.SHDSL incorporates features of other DSL technologies such as asymmetric DSL and transports T1, E1, ISDN, Asynchronous Transfer Mode (ATM), and IP signals.
symmetric high-speed digital subscriber line (SHDSL) transceiver unit–remote (STU–R)	Equipment that provides symmetric high-speed digital subscriber line (SHDSL) connections to remote user terminals such as data terminals or telecommunications equipment.
T1 interface	Physical WAN interface for transmitting digital signals in the T-carrier system used in the United States, Japan, and Canada. The T1 signal format carries 24 pulse code modulation (PCM) signals using time-division multiplexing (TDM) at an overall rate of 1.544 Mbps.

Table 14: Network Interfaces Terms *(continued)*

Term	Definition
T3 interface	Physical WAN interface for transmitting digital signals in the T-carrier system used in the United States, Japan, and Canada. T3 signals are formatted like T1 signals, but carry information at the higher rate of 44.736 Mbps. T3 is also called DS3.

Network Interfaces

All Juniper Networks devices use network interfaces to make physical connections to other devices. A connection takes place along media-specific physical wires through a port on a Physical Interface Module (PIM) installed in the J-series Services Router or an Input/Output Card (IOC) in the SRX-series services gateway. Each device interface has a unique name that follows a naming convention.

This section contains the following topics:

Media Types

Each type of interface on a J-series or SRX-series device uses a particular medium to transmit data. The physical wires and data link layer protocols used by a medium determine how traffic is sent. See Table 4 on page 3 and Table 9 on page 5 for a list of media supported on each type of device.

You must configure each network interface before it can operate on the device. Configuring an interface can define both the physical properties of the link and the logical properties of a logical interface on the link.

Network Interface Naming

The interfaces on the J-series and SRX-series devices are used for networking and services. Most interfaces are configurable, but some internally generated interfaces are not configurable. If you are familiar with Juniper Networks M-series and T-series routing platforms, be aware that device interface names are similar to but not identical with the interface names on those routing platforms.

This section contains the following topics:

Interface Naming Conventions

The unique name of each network interface identifies its type and location and indicates whether it is a physical interface or an optional logical unit created on a physical interface:

- The name of each network interface has the following format to identify the physical device that corresponds to a single physical network connector:

  type-slot/pim-or-ioc/port

- Network interfaces that are fractionalized into time slots include a channel number in the name, preceded by a colon (:):

  type-slot/pim-or-ioc/port:*channel*

- Each logical interface has an additional logical unit identifier, preceded by a period (.):

  type-slot/pim-or-ioc/port:<*channel*>.*unit*

The parts of an interface name are summarized in Table 15 on page 18.

Table 15: Network Interface Names

Name Part	Meaning	Possible Values
type	Type of network medium that can connect to this interface.	at—ATM-over-ADSL or ATM-over-SHDSL WAN interface bc—Bearer channel on an ISDN interface br—Basic Rate Interface for establishing ISDN connections ce1—Channelized E1 interface ct1—Channelized T1 interface dc—Delta channel on an ISDN interface dl—Dialer interface for initiating ISDN and USB modem connections e1—E1 WAN interface e3—E3 WAN interface fe—Fast Ethernet interface ge—Gigabit Ethernet interface reth—For chassis cluster configurations only, redundant Ethernet interface se—Serial interface (either RS-232, RS-422/499, RS-530, V.35, or X.21) t1—T1 (also called DS1) WAN interface t3—T3 (also called DS3) WAN interface wx—WXC Integrated Services Module (ISM 200) interface for WAN acceleration xe—10-Gigabit Ethernet interface In addition to these network interfaces, devices can have the following special interfaces: dsc, gr and gre, ip and ipip, lo, ls and lsi, lt, pd and pimd, pc, pe and pime, pp0, st, tap, and umd0. For more information, see “Special Interfaces” on page 67.

Table 15: Network Interface Names *(continued)*

Name Part	Meaning	Possible Values
slot	Number of the chassis slot in which a PIM or IOC is installed.	J-series Services Router: The slot number begins at 1 and increases as follows from top to bottom, left to right: ■ J2320 router—Slots 1 to 3 ■ J2350 router—Slots 1 to 5 ■ J4350 or J6350 router—PIM slots 1 to 6 The slot number 0 is reserved for the out-of-band management ports. (See "Management Interface" on page 70.) SRX 5600 and 5800 services gateways: The slot number begins at 0 and increases as follows from left to right, bottom to top: ■ SRX 5600 services gateway—Slots 0 to 5 ■ SRX 5800 services gateway—Slots 0 to 5, 7 to 11 SRX 3400 and 3600 services gateways: The Switch Fabric Board (SFB) is always 0. Slot numbers increase as follows from top to bottom, left to right: ■ SRX 3400 services gateway—Slots 0 to 4 ■ SRX 3600 services gateway—Slots 0 to 6
pim-or-ioc	Number of the PIM or IOC on which the physical interface is located.	J-series Services Routers: This number is always 0. Only one PIM can be installed in a slot. SRX 5600 and 5800 services gateways: For 40-port Gigabit Ethernet IOCs or 4–port 10-Gigabit Ethernet IOCs, this number can be 0, 1, 2, or 3. SRX 3400 and 3600 services gateways: This number is always 0. Only one IOC can be installed in a slot.

Table 15: Network Interface Names *(continued)*

Name Part	Meaning	Possible Values
port	Number of the port on a PIM or IOC on which the physical interface is located.	J-series Services Routers: ■ On a single-port PIM, always 0. ■ On a multiple-port PIM, this number begins at 0 and increases from left to right, bottom to top, to a maximum of 3. On SRX 5400 and 5800 services gateways: ■ For 40-port Gigabit Ethernet IOCs, this number begins at 0 and increases from left to right to a maximum of 9. ■ For 4-port 10-Gigabit Ethernet IOCs, this number is always 0. On SRX 3400 and 3600 services gateways: ■ For the SFB built-in copper Gigabit Ethernet ports, this number begins at 0 and increases from top to bottom, left to right, to a maximum of 7. For the SFB built-in fiber Gigabit Ethernet ports, this number begins at 8 and increases from left to right to a maximum of 11. ■ For 16-port Gigabit Ethernet IOCs, this number begins at 0 to a maximum of 15. ■ For 2-port 10-Gigabit Ethernet IOCs, this number is 0 or 1. Port numbers appear on the PIM or IOC faceplate.
channel	Number of the channel (time slot) on a fractional or channelized T1 or E1 interface.	■ On an E1 interface, a value from 1 through 31. The 1 time slot is reserved. ■ On a T1 interface, a value from 1 through 24.
unit	Number of the logical interface created on a physical interface.	A value from 0 through 16384. If no logical interface number is specified, unit 0 is the default, but must be explicitly configured. For more information about logical interfaces, see “Interface Logical Properties” on page 59.

For example, the interface name e1-5/0/0:15.0 on a J-series Services Router represents the following information:

- E1 WAN interface
- PIM slot 5
- PIM number 0 (always 0)
- Port 0
- Channel 15
- Logical interface, or unit, 0

Understanding CLI Output for Interfaces

The JUNOS software that operates on J-series Services Routers and SRX-series services gateways was originally developed for Juniper Networks routing platforms that support many ports, on interface cards called Physical Interface Cards (PICs). On these larger platforms, PICs are installed into slots on Flexible PIC Concentrators (FPCs), and FPCs are installed into slots in the router chassis.

For J-series Services Routers and SRX-series services gateways, PIM and IOC slots are detected internally by the JUNOS software as FPC slots, and the PIM or IOC in each slot is identified as a "PIC." For example, in the following output, the three PIMs located in slots 0, 2, and 5 are reported as FPC 0, FPC 2, and FPC 5, and PIM 0 is reported as PIC 0:

```
user@host> show chassis hardware
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                JN000192AB        J4350
Midplane         REV 02.04 710-010001  CORE99563
System IO        REV 02.03 710-010003  CORE100885        System IO board
Routing Engine   RevX2.6  750-010005   IWGS40735451      RE-J.2
FPC 0                                                   FPC
  PIC 0                                                 2x FE
FPC 2            RevX2.1  750-010355   CORE100458        FPC
  PIC 0                                                 2x T1
FPC 5            REV 04   750-010353   AF04451744        FPC
  PIC 0                                                 2x FE
```

To understand the abbreviations for PICs that appear in JUNOS CLI output, see Table 16 on page 21. For details, see the *J-series Services Routers Hardware Guide*.

Table 16: PIC Abbreviations and Full Names

PIC Abbreviation in JUNOS CLI	PIC or IOC Name
2x FE	Dual-Port Fast Ethernet PIM
4x FE	4-Port Fast Ethernet ePIM
1x GE Copper	Copper Gigabit Ethernet ePIM (1 10-Mbps, 100-Mbps, or 1000-Mbps port)
1x GE SFP	SFP Gigabit Ethernet ePIM (1 fiber port)
1x SFP uPIM	1-Port Gigabit Ethernet uPIM
6x GE SFP uPIM	6-Port SFP Gigabit Ethernet uPIM
8x GE uPIM	8-Port Gigabit Ethernet uPIM
16x GE uPIM	16-Port Gigabit Ethernet uPIM
4x GE Base PIC	4 built-in Gigabit Ethernet ports on a chassis (fixed PIM)
1x 10GE	1-Port 10-Gigabit Ethernet IOC

Table 16: PIC Abbreviations and Full Names *(continued)*

PIC Abbreviation in JUNOS CLI	PIC or IOC Name
10x 1GE	10-Port Gigabit Ethernet IOC
2x Serial	Dual-Port Serial PIM
2x T1	Dual-Port T1 PIM
2x E1	Dual-Port E1 PIM
2x CT1E1 /PRI	Dual-Port Channelized T1/E1/ISDN PRI PIM
1x T3	T3 PIM (1 port)
1x E3	E3 PIM (1 port)
4x BRI S/T	4-Port ISDN BRI S/T PIM
4x BRI U	4-Port ISDN BRI U PIM
1x ADSL Annex A	ADSL 2/2 + Annex A PIM (1 port, for POTS)
1x ADSL Annex B	ADSL 2/2 + Annex B PIM (1 port, for ISDN)
2x SHDSL (ATM)	G.SHDSL PIM (2-port two-wire mode or 1-port four-wire mode)
Integrated Services Module	WXC Integrated Services Module (ISM 200)

Data Link Layer Overview

The data link layer is Layer 2 in the Open Systems Interconnection (OSI) model. The data link layer is responsible for transmitting data across a physical network link. Each physical medium has link-layer specifications for network and link-layer protocol characteristics such as physical addressing, network topology, error notification, frame sequencing, and flow control.

Physical Addressing

Physical addressing is different from network addressing. Network addresses differentiate between nodes or devices in a network, allowing traffic to be routed or switched through the network. In contrast, physical addressing identifies devices at the link-layer level, differentiating between individual devices on the same physical medium. The primary form of physical addressing is the media access control (MAC) address.

Network Topology

Network topology specifications identify how devices are linked in a network. Some media allow devices to be connected by a bus topology, while others require a ring

topology. The bus topology is used by Ethernet technologies, which are supported on Juniper Networks devices.

Error Notification

The data link layer provides error notifications that alert higher-layer protocols that an error has occurred on the physical link. Examples of link-level errors include the loss of a signal, the loss of a clocking signal across serial connections, or the loss of the remote endpoint on a T1 or T3 link.

Frame Sequencing

The frame sequencing capabilities of the data link layer allow frames that are transmitted out of sequence to be reordered on the receiving end of a transmission. The integrity of the packet can then be verified by means of the bits in the Layer 2 header, which is transmitted along with the data payload.

Flow Control

Flow control within the data link layer allows receiving devices on a link to detect congestion and notify their upstream and downstream neighbors. The neighbor devices relay the congestion information to their higher-layer protocols so that the flow of traffic can be altered or rerouted.

Data Link Sublayers

The data link layer is divided into two sublayers: Logical Link Control (LLC) and Media Access Control (MAC). The LLC sublayer manages communications between devices over a single link of a network. This sublayer supports fields in link-layer frames that enable multiple higher-layer protocols to share a single physical link.

The MAC sublayer governs protocol access to the physical network medium. Through the MAC addresses that are typically assigned to all ports on a device, multiple devices on the same physical link can uniquely identify one another at the data link layer. MAC addresses are used in addition to the network addresses that are typically configured manually on ports within a network.

MAC Addressing

A MAC address is the serial number permanently stored in a device adapter to uniquely identify the device. MAC addresses operate at the data link layer, while IP addresses operate at the network layer. The IP address of a device can change as the device is moved around a network to different IP subnets, but the MAC address remains the same, because it is physically tied to the device.

Within an IP network, devices match each MAC address to its corresponding configured IP address by means of the Address Resolution Protocol (ARP). ARP maintains a table with a mapping for each MAC address in the network.

Most Layer 2 networks use one of three primary numbering spaces—MAC-48, EUI-48 (Extended Unique Identifier), and EUI-64—which are all globally unique. MAC-48

and EUI-48 spaces each use 48-bit addresses, and EUI-64 spaces use a 64-bit addresses, but all three use the same numbering format. MAC-48 addresses identify network hardware, and EUI-48 addresses identify other devices and software.

The Ethernet and ATM technologies supported on devices use the MAC-48 address space. IPv6 uses the EUI-64 address space.

MAC-48 addresses are the most commonly used MAC addresses in most networks. These addresses are 12-digit hexadecimal numbers (48 bits in length) that typically appear in one of the following formats:

- *MM:MM:MM:SS:SS:SS*
- *MM-MM-MM-SS-SS-SS*

The first three octets (*MM:MM:MM* or *MM-MM-MM*) are the ID number of the hardware manufacturer. Manufacturer ID numbers are assigned by the Institute of Electrical and Electronics Engineers (IEEE). The last three octets (*SS:SS:SS* or *SS-SS-SS*) make up the serial number for the device, which is assigned by the manufacturer. For example, an Ethernet interface card might have a MAC address of 00:05:85:c1:a6:a0.

Ethernet Interface Overview

Ethernet is a Layer 2 technology that operates in a shared bus topology. Ethernet supports broadcast transmission, uses best-effort delivery, and has distributed access control. Ethernet is a point-to-multipoint technology.

In a shared bus topology, all devices connect to a single, shared physical link through which all data transmissions are sent. All traffic is broadcast, so that all devices within the topology receive every transmission. The devices within a single Ethernet topology make up a broadcast domain.

Ethernet uses best-effort delivery to broadcast traffic. The physical hardware provides no information to the sender about whether the traffic was received. If the receiving host is offline, traffic to the host is lost. Although the Ethernet data link protocol does not inform the sender about lost packets, higher-layer protocols like TCP/IP might provide this type of notification.

This section contains the following topics:

Ethernet Access Control and Transmission

Ethernet's access control is distributed, because Ethernet has no central mechanism that grants access to the physical medium within the network. Instead, Ethernet uses carrier sense multiple access with collision detection (CSMA/CD). Because multiple

devices on an Ethernet network can access the physical medium, or wire, simultaneously, each device must determine whether the physical medium is in use. Each host listens on the wire to determine if a message is being transmitted. If it detects no transmission, the host begins transmitting its own data.

The length of each transmission is determined by fixed Ethernet packet sizes. By fixing the length of each transmission and enforcing a minimum idle time between transmissions, Ethernet ensures that no pair of communicating devices on the network can monopolize the wire and block others from sending and receiving traffic.

Collisions and Detection

When a device on an Ethernet network begins transmitting data, the data takes a finite amount of time to reach all hosts on the network. Because of this delay, or latency, in transmitting traffic, a device might detect an idle state on the wire just as another device initially begins its transmission. As a result, two devices might send traffic across a single wire at the same time. When the two electrical signals collide, they become scrambled so that both transmissions are effectively lost.

Collision Detection

To handle collisions, Ethernet devices monitor the link while they are transmitting data. The monitoring process is known as collision detection. If a device detects a foreign signal while it is transmitting, it terminates the transmission and attempts to transmit again only after detecting an idle state on the wire. Collisions continue to occur if two colliding devices both wait the same amount of time before retransmitting. To avoid this condition, Ethernet devices use a binary exponential backoff algorithm.

Backoff Algorithm

To use the binary exponential backoff algorithm, each device that sent a colliding transmission randomly selects a value within a range. The value represents the number of transmission times that the device must wait before retransmitting its data. If another collision occurs, the range of values is doubled and retransmission takes place again. Each time a collision occurs, the range of values doubles, to reduce the likelihood that two hosts on the same network can select the same retransmission time. Table 17 on page 25 shows collision rounds up to round 10.

Table 17: Collision Backoff Algorithm Rounds

Round	Size of Set	Elements in the Set
1	2	{0,1}
2	4	{0,1,2,3}
3	8	{0,1,2,3,...,7}
4	16	{0,1,2,3,4,...,15}
5	32	{0,1,2,3,4,5,...,31}

Table 17: Collision Backoff Algorithm Rounds *(continued)*

Round	Size of Set	Elements in the Set
6	64	{0,1,2,3,4,5,6,...,63}
7	128	{0,1,2,3,4,5,6,7,...,127}
8	256	{0,1,2,3,4,5,6,7,8,...,255}
9	512	{0,1,2,3,4,5,6,7,8,9,...,511}
10	1024	{0,1,2,3,4,5,6,7,8,9,10,...,1023}

Collision Domains and LAN Segments

Collisions are confined to a physical wire over which data is broadcast. Because the physical wires are subject to signal collisions, individual LAN segments are known as collision domains. Although the physical limitations on the length of an Ethernet cable restrict the length of a LAN segment, multiple collision domains can be interconnected by repeaters, bridges, and switches.

Repeaters

Repeaters are electronic devices that act on analog signals. Repeaters relay all electronic signals from one wire to another. A single repeater can double the distance between two devices on an Ethernet network. However, the Ethernet specification restricts the number of repeaters between any two devices on an Ethernet network to two, because collision detection with latencies increases in complexity as the wire length and number of repeaters increase.

Bridges and Switches

Bridges and switches combine LAN segments into a single Ethernet network by using multiple ports to connect the physical wires in each segment. Although bridges and switches are fundamentally the same, bridges generally provide more management and more interface ports. As Ethernet packets flow through a bridge, the bridge tracks the source MAC address of the packets and stores the addresses and their associated input ports in an interface table. As it receives subsequent packets, the bridge examines its interface table and takes one of the following actions:

- If the destination address does not match an address in the interface table, the bridge transmits the packet to all hosts on the network using the Ethernet broadcast address.
- If the destination address maps to the port through which the packet was received, the bridge or switch discards the packet. Because the other devices on the LAN segment also received the packet, the bridge does not need to retransmit it.
- If the destination address maps to a port other than the one through which the packet was received, the bridge transmits the packet through the appropriate port to the corresponding LAN segment.

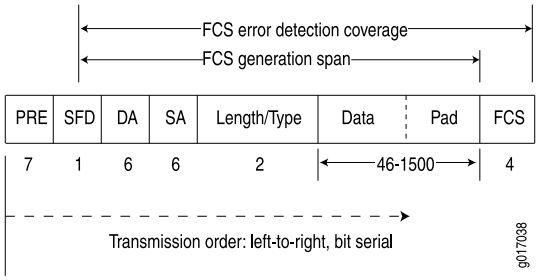
Broadcast Domains

The combination of all the LAN segments within an Ethernet network is called a broadcast domain. In the absence of any signaling devices such as a repeater, bridge, or switch, the broadcast domain is simply the physical wire that makes up the connections in the network. If a bridge or switch is used, the broadcast domain consists of the entire LAN.

Ethernet Frames

Data is transmitted through an Ethernet network in frames. The frames are of variable length, ranging from 64 octets to 1518 octets, including the header, payload, and cyclic redundancy check (CRC) value. Figure 1 on page 27 shows the Ethernet frame format.

Figure 1: Ethernet Frame Format



Ethernet frames have the following fields:

- The preamble (PRE) in the frame is 7 octets of alternating 0s and 1s. The predictable format in the preamble allows receiving interfaces to synchronize themselves to the data being sent. The preamble is followed by a 1-octet start-of-frame delimiter (SFD).
- The destination address (DA) and source address (SA) fields contain the 6-octet (48-bit) MAC addresses for the destination and source ports on the network. These Layer 2 addresses uniquely identify the devices on the LAN.
- The length/type field is a 2-octet field that either indicates the length of the frame's data field or identifies the protocol stack associated with the frame. Following are some common frame types:
  - AppleTalk—0x809B
  - AppleTalk ARP—0x80F3
  - DECnet—0x6003
  - IP—0x0800
  - IPX—0x8137
  - Loopback—0x9000
  - XNS—0x0600

- The frame data is the packet payload.
- The frame check sequence (FCS) field is a 4-octet field that contains the calculated CRC value. This value is calculated by the originating host and appended to the frame. When it receives the frames, the receiving host calculates the CRC and checks it against this appended value to verify the integrity of the received frame.

T1 and E1 Interfaces Overview

T1 and E1 are equivalent digital data transmission formats that carry DS1 signals. T1 and E1 lines can be interconnected for international use. This section contains the following topics:

T1 Overview

T1 is a digital data transmission medium capable of handling 24 simultaneous connections running at a combined 1.544 Mbps. T1 combines these 24 separate connections, called channels or time slots, onto a single link. T1 is also called DS1.

The T1 data stream is broken into frames. Each frame consists of a single framing bit and 24 8-bit channels, totalling 193 bits per T1 frame. Frames are transmitted 8,000 times per second, at a data transmission rate of 1.544 Mbps (8,000 x 193 = 1.544 Mbps).

As each frame is received and processed, the data in each 8-bit channel is maintained with the channel data from previous frames, enabling T1 traffic to be separated into 24 separate flows across a single medium. For example, in the following set of 4-channel frames (without a framing bit), the data in channel 1 consists of the first octet of each frame, the data in channel 2 consists of the second octet of each frame, and so on:

```
           Chan. 1   Chan. 2   Chan. 3   Chan. 4
Frame 1  [10001100][00110001][11111000][10101010]
Frame 2  [11100101][01110110][10001000][11001010]
Frame 3  [00010100][00101111][11000001][00000001]
```

E1 Overview

E1 is the European format for DS1 digital transmission. E1 links are similar to T1 links except that they carry signals at 2.048 Mbps. Each signal has 32 channels, and each channel transmits at 64 Kbps. E1 links have higher bandwidth than T1 links because they use all 8 bits of a channel. T1 links use 1 bit in each channel for overhead.

T1 and E1 Signals

T1 and E1 interfaces consist of two pairs of wires—a transmit data pair and a receive data pair. Clock signals, which determine when the transmitted data is sampled, are embedded in T1 and E1 transmissions.

Typical digital signals operate by sending either zeros (0s) or ones (1s), which are usually represented by the absence or presence of a voltage on the line. The receiving device need only detect the presence of the voltage on the line at the particular sampling edge to determine if the signal is 0 or 1. T1 and E1, however, use bipolar electrical pulses. Signals are represented by no voltage (0), positive voltage (1), or negative voltage (1). The bipolar signal allows T1 and E1 receivers to detect error conditions in the line, depending on the type of encoding that is being used. For more information, see "Encoding" on page 29.

Encoding

Following are common T1 and E1 encoding techniques:

- Alternate mark inversion (AMI)—T1 and E1
- Bipolar with 8-zero substitution (B8ZS)—T1 only
- High-density bipolar 3 code (HDB3)—E1 only

AMI Encoding

AMI encoding forces the 1s signals on a T1 or E1 line to alternate between positive and negative voltages for each successive 1 transmission, as in this sample data transmission:

```
1 1 0 1 0 1 0 1
+ - 0 + 0 - 0 +
```

When AMI encoding is used, a data transmission with a long sequence of 0s has no voltage transitions on the line. In this situation, devices have difficulty maintaining clock synchronization, because they rely on the voltage fluctuations to constantly synchronize with the transmitting clock. To counter this effect, the number of consecutive 0s in a data stream is restricted to 15. This restriction is called the 1s density requirement, because it requires a certain number of 1s for every 15 0s that are transmitted.

On an AMI-encoded line, two consecutive pulses of the same polarity—either positive or negative—are called a bipolar violation (BPV), which is generally flagged as an error.

B8ZS and HDB3 Encoding

Both B8ZS and HDB3 encoding do not restrict the number of 0s that can be transmitted on a line. Instead, these encoding methods detect sequences of 0s and substitute bit patterns in their place to provide the signal oscillations required to maintain timing on the link.

The B8ZS encoding method for T1 lines detects sequences of eight consecutive 0 transmissions and substitutes a pattern of two consecutive BPVs (11110000). Because the receiving end uses the same encoding, it detects the BPVs as 0s substitutions, and no BPV error is flagged. A single BPV, which does not match the 11110000 substitution bit sequence, is likely to generate an error, depending on the configuration of the device.

The HDB3 encoding method for E1 lines detects sequences of four consecutive 0 transmissions and substitutes a single BPV (1100). Similar to B8ZS encoding, the receiving device detects the 0s substitutions and does not generate a BPV error.

T1 and E1 Framing

J-series Services Router T1 interfaces use two types of framing: superframe (D4) and extended superframe (ESF). E1 interfaces use G.704 framing or G.704 with no CRC4 framing, or can be in unframed mode.

Superframe (D4) Framing for T1

A D4 frame consists of 192 data bits: 24 8-bit channels and a single framing bit. The single framing bit is part of a 12-bit framing sequence. The 193rd bit in each T1 frame is set to a value, and every 12 consecutive frames are examined to determine the framing bit pattern for the 12-bit superframe.

The following sample 12-frame sequence shows the framing pattern for D4 framing:

```
[data bits][framing bit]
[xxxxxxxxx][1]
[xxxxxxxxx][0]
[xxxxxxxxx][0]
[xxxxxxxxx][0]
[xxxxxxxxx][1]
[xxxxxxxxx][1]
[xxxxxxxxx][0]
[xxxxxxxxx][1]
[xxxxxxxxx][1]
[xxxxxxxxx][1]
[xxxxxxxxx][0]
[xxxxxxxxx][0]
```

The 100011011100 12-bit pattern is repeated in each successive superframe. The receiving device detects these bits to synchronize with the incoming data stream and determine when the framing pattern begins and ends.

D4 framing requires the 8th bit of every byte (of every channel) within the frame to be set to 1, a process known as bit robbing. The bit-robbing requirement ensures that the 1s density requirements are met, regardless of the data contents of the frames, but it reduces the bandwidth on the T1 link by an eighth.

Extended Superframe (ESF) Framing for T1

ESF extends the D4 superframe from 12 frames to 24 frames. By expanding the size of the superframe, ESF increases the number of bits in the superframe framing

pattern from 12 to 24. The extra bits are used for frame synchronization, error detection, and maintenance communications through the facilities data link (FDL).

The ESF pattern for synchronization bits is 001011. Only the framing bits from frames 4, 8, 12, 16, 20, and 24 in the superframe sequence are used to create the synchronization pattern.

The framing bits from frames 2, 6, 10, 14, 18, and 22 are used to pass a CRC code for each superframe block. The CRC code verifies the integrity of the received superframe and detects bit errors with a CRC6 algorithm.

The framing bits for frames 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, and 23 are used for the data link channel. These 12 bits enable the operators at the network control center to query the remote equipment for information about the performance of the link.

T1 and E1 Loopback Signals

The control signal on a T1 or E1 link is the loopback signal. Using the loopback signal, the operators at the network control center can force the device at the remote end of a link to retransmit its received signals back onto the transmit path. The transmitting device can then verify that the received signals match the transmitted signals, to perform end-to-end checking on the link.

Two loopback signals are used to perform the end-to-end testing:

- The loop-up command signal sets the link into loopback mode, with the following command pattern:

  ```
  ....100001000010000100...
  ```

- The loop-down signal returns the link to its normal mode, with the following command pattern:

  ```
  ....100100100100100100....
  ```

While the link is in loopback mode, the operator can insert test equipment onto the line to test its operation.

Channelized T1/E1/ISDN PRI Interfaces Overview

Channelization enables devices to provide IP services to users with different access speeds and bandwidth requirements. Users share an interface that has been divided into discrete time slots, by transmitting in only their own time slot. On J-series devices, a single channelized T1/E1/ISDN PRI interface can be partitioned into the following numbers of DS0 or ISDN PRI time slots, by means of software configuration:

- T1 interface—Up to 24 DS0 time slots (channels 1 through 24).
- E1 interface—Up to 31 DS0 time slots (channels 1 through 31).

- ISDN PRI—Up to 23 ISDN PRI B-channels and 1 D-channel when the parent interface is channelized T1, and up to 30 ISDN PRI B-channels and 1 D channel when the parent interface is channelized E1. Time slots on the interface unused by ISDN PRI can operate normally as DS0 interfaces.

  For more information about ISDN, see "ISDN Interface Overview" on page 45.

NOTE: You cannot configure the channelized T1/E1/ISDN PRI PIM through a J-Web Quick Configuration page.

You can aggregate the channels on a channelized interface into bundles called channel groups to aggregate customer traffic.

A single channelized T1/E1/ISDN PRI interface also supports drop-and-insert multiplexing, to integrate voice and data channels on a single T1 or E1 link. The drop-and-insert feature allows you to remove the DS0 time slots of one T1 or E1 port and replace them by inserting the time slots of another T1 or E1 interface.

T3 and E3 Interfaces Overview

T3 is a high-speed data-transmission medium formed by multiplexing 28 DS1 signals into seven separate DS2 signals, and combining the DS2 signals into a single DS3 signal. T3 links operate at 43.736 Mbps. T3 is also called DS3.

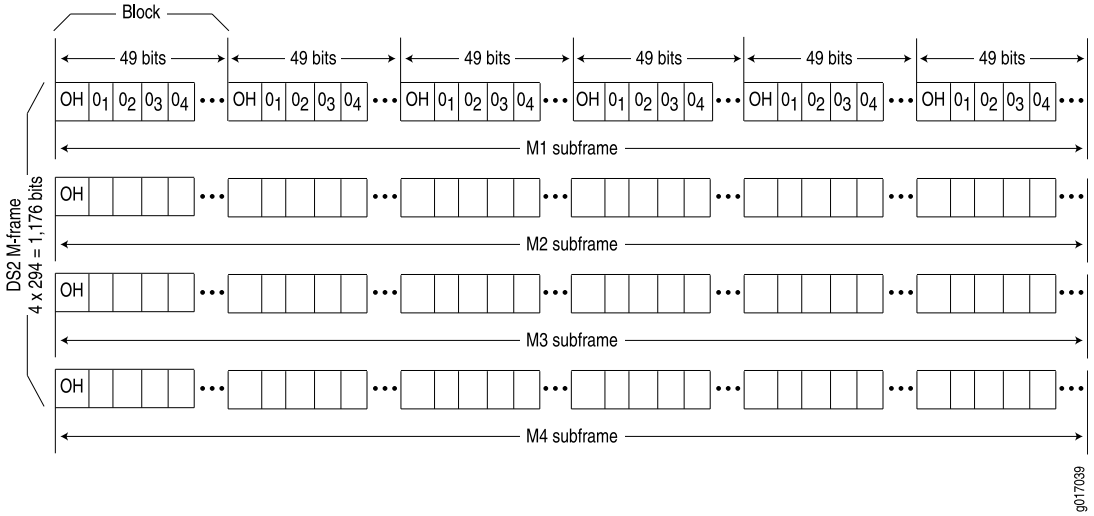
E3 is the equivalent European transmission format. E3 links are similar to T3 (DS3) links, but carry signals at 34.368 Mbps. Each signal has 16 E1 channels, and each channel transmits at 2.048 Mbps. E3 links use all 8 bits of a channel, whereas T3 links use 1 bit in each channel for overhead.

Multiplexing DS1 Signals

Four DS1 signals combine to form a single DS2 signal. The four DS1 signals form a single DS2 M-frame, which includes subframes M1 through M4. Each subframe has six 49-bit blocks, for a total of 294 bits per subframe. The first bit in each block is a DS2 overhead (OH) bit. The remaining 48 bits are DS1 information bits.

Figure 2 on page 33 shows the DS2 M-frame format.

Figure 2: DS2 M-Frame Format



The four DS2 subframes are not four DS1 channels. Instead, the DS1 data bits within the subframes are formed by data interleaved from the DS1 channels. The 0_n values designate time slots devoted to DS1 inputs as part of the bit-by-bit interleaving process. After every 48 DS1 information bits (12 bits from each signal), a DS2 OH bit is inserted to indicate the start of a subframe.

DS2 Bit Stuffing

Because the four DS1 signals are asynchronous signals, they might operate at different line rates. To synchronize the asynchronous streams, the multiplexers on the line use bit stuffing.

A DS2 connection requires a nominal transmit rate of 6.304 Mbps. However, because multiplexers increase the overall output rate to the intermediate rate of 6.312 Mbps, the output rate is higher than individual input rates on DS1 signals. The extra bandwidth is used to stuff the incoming DS1 signals with extra bits until the output rate of each signal equals the increased intermediate rate. These stuffed bits are inserted at fixed locations in the DS2 M-frame. When DS2 frames are received and the signal is demultiplexed, the stuffing bits are identified and removed.

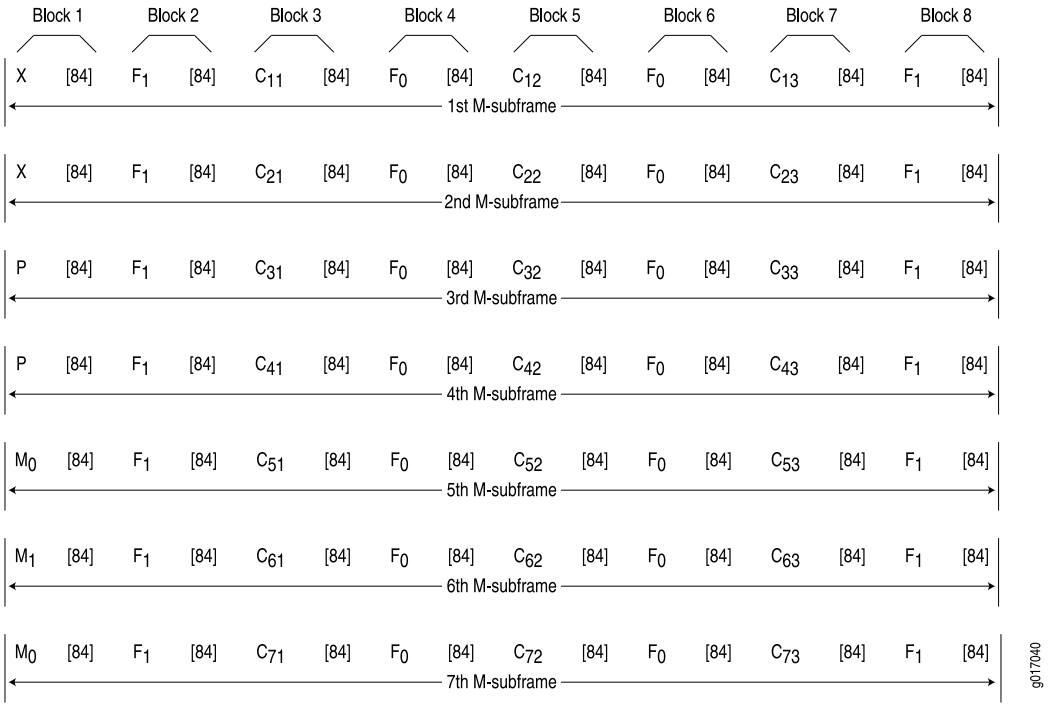
DS3 Framing

A set of four DS1 signals is multiplexed into seven DS2 signals, which are multiplexed into a single DS3 signal. The multiplexing occurs just as with DS1-to-DS2 multiplexing. The resulting DS3 signal uses either the standard M13 asynchronous framing format or the C-bit parity framing format. Although the two framing formats differ in their use of control and message bits, the basic frame structures are identical. The DS3 frame structures are shown in Figure 3 on page 34 and Figure 4 on page 35.

M13 Asynchronous Framing

A DS3 M-frame includes seven subframes, formed by DS2 data bits interleaved from the seven multiplexed DS2 signals. Each subframe has eight 85-bit blocks—a DS3 OH bit plus 84 data bits. The meaning of an OH bit depends on the block it precedes. Standard DS3 M13 asynchronous framing format is shown in Figure 3 on page 34.

Figure 3: DS3 M13 Frame Format



A DS3 M13 M-frame contains the following types of OH bits:

- Framing bits (F-bits)—Make up a frame alignment signal that synchronizes DS3 subframes. Each DS3 frame contains 28 F-bits (4 bits per subframe). F-bits are located at the beginning of blocks 2, 4, 6, and 8 of each subframe. When combined, the frame alignment pattern for each subframe is 1001. The pattern can be examined to detect bit errors in the transmission.
- Multiframing bits (M-bits)—Make up a multiframe alignment signal that synchronizes the M-frames in a DS3 signal. Each DS3 frame contains 3 M-bits, which are located at the beginning of subframes 5, 6, and 7. When combined, the multiframe alignment patter for each M-frame is 010.
- Bit stuffing control bits (C-bits)—Serve as bit stuffing indicators for each DS2 input. For example, C_{11}, C_{12}, and C_{13} are indicators for DS2 input 1. Their values indicate whether DS3 bit stuffing has occurred at the multiplexer. If the three C-bits in a subframe are all 0s, no stuffing was performed for the DS2 input. If the three C-bits are all 1s, stuffing was performed.
- Message bits (X-bits)—Used by DS3 transmitters to embed asynchronous in-service messages in the data transmission. Each DS3 frame contains 2 X-bits,

which are located at the beginning of subframes 1 and 2. Within an DS3 M-frame, both X-bits must be identical.

- Parity bits (P-bits)—Compute parity over all but 1 bit of the M-frame. (The first X-bit is not included.) Each DS3 frame contains 2 P-bits, which are located at the beginning of subframes 3 and 4. Both P-bits must be identical.

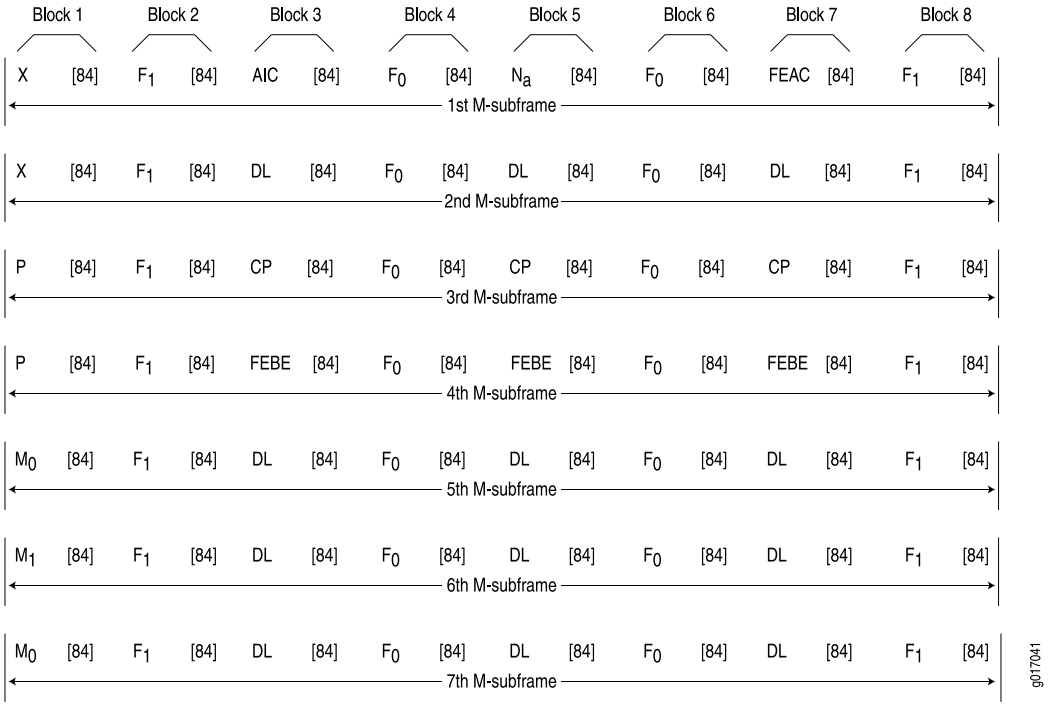
  If the previous DS3 frame contained an odd number of 1s, both P-bits are set to 1. If the previous DS3 contained an even number of 1s, both P-bits are set to 0. If, on the receiving side, the number of 1s for a given frame does not match the P-bits in the following frame, it indicates one or more bit errors in the transmission.

C-Bit Parity Framing

In M13 framing, every C-bit in a DS3 frame is used for bit stuffing. However, because multiplexers first use bit stuffing when multiplexing DS1 signals into DS2 signals, the incoming DS2 signals are already synchronized. Therefore, the bit stuffing that occurs when DS2 signals are multiplexed is redundant.

C-bit parity framing format redefines the function of C-bits and X-bits, using them to monitor end-to-end path performance and provide in-band data links. The C-bit parity framing structure is shown in Figure 4 on page 35.

Figure 4: DS3 C-Bit Parity Framing



In C-bit parity framing, the X-bits transmit error conditions from the far end of the link to the near end. If no error conditions exist, both X-bits are set to 1. If an out-of-frame (OOF) or alarm indication signal (AIS) error is detected, both X-bits are

set to 0 in the upstream direction for 1 second to notify the other end of the link about the condition.

The C-bits that control bit stuffing in M13 frames are typically used in the following ways by C-bit parity framing:

- Application identification channel (AIC)—The first C-bit in the first subframe identifies the type of DS3 framing used. A value of 1 indicates that C-bit parity framing is in use.
- N_a—A reserved network application bit.
- Far-end alarm and control (FEAC) channel—The third C-bit in the first subframe is used for the FEAC channel. In normal transmissions, the FEAC C-bit transmits all 1s. When an alarm condition is present, the FEAC C-bit transmits a code word in the format 0*xxxxxxx* 11111111, in which *x* can be either 1 or 0. Bits are transmitted from right to left.

  Table 18 on page 36 lists some C-bit code words and the alarm or status condition indicated.

Table 18: FEAC C-Bit Condition Indicators

Alarm or Status Condition	C-Bit Code Word
DS3 equipment failure requires immediate attention.	00110010 11111111
DS3 equipment failure occurred—such as suspended, not activated, or unavailable service—that is non-service-affecting.	00011110 11111111
DS3 loss of signal.	00011100 11111111
DS3 out of frame.	00000000 11111111
DS3 alarm indication signal (AIS) received.	00101100 11111111
DS3 idle received.	00110100 11111111
Common equipment failure occurred that is non-service-affecting.	00011101 11111111
Multiple DS1 loss of signal.	00101010 11111111
DS1 equipment failure occurred that requires immediate attention.	00001010 11111111
DS1 equipment failure occurred that is non-service-affecting.	00000110 11111111
Single DS1 loss of signal.	00111100 11111111

- Data links—The 12 C-bits in subframes 2, 5, 6, and 7 are data link (DL) bits for applications and terminal-to-terminal path maintenance.
- DS3 parity—The 3 C-bits in the third subframe are DS3 parity C-bits (also called CP-bits). When a DS3 frame is transmitted, the sending device sets the CP-bits to the same value as the P-bits. When the receiving device processes the frame, it calculates the parity of the M-frame and compares this value to the parity in

the CP-bits of the following M-frame. If no bit errors have occurred, the two values are typically the same.

- Far–end block errors (FEBEs)—The 3 C-bits in the fourth subframe make up the far-end block error (FEBE) bits. If a framing or parity error is detected in an incoming M-frame (via the CP-bits), the receiving device generates a C-bit parity error and sends an error notification to the transmitting (far-end) device. If an error is generated, the FEBE bits are set to 000. If no error occurred, the bits are set to 111.

Serial Interface Overview

Serial links are simple, bidirectional links that require very few control signals. In a basic serial setup, data communications equipment (DCE) installed in a user's premises is responsible for establishing, maintaining, and terminating a connection. A modem is a typical DCE device.

A serial cable connects the DCE to a telephony network where, ultimately, a link is established with data terminal equipment (DTE). DTE is typically where a serial link terminates.

The distinction between DCE and DTE is important because it affects the cable pinouts on a serial cable. A DTE cable uses a male 9-pin or 25-pin connector, and a DCE cable uses a female 9-pin or 25-pin connector.

To form a serial link, the cables are connected to each other. However, if the pins are identical, each side's transmit and receive lines are connected, which makes data transport impossible. To address this problem, each cable is connected to a null modem cable, which crosses the transmit and receive lines in the cable.

Serial Transmissions

In basic serial communications, nine signals are critical to the transmission. Each signal is associated with a pin in either the 9-pin or 25-pin connector. Table 19 on page 37 lists and defines serial signals and their sources.

Table 19: Serial Transmission Signals

Signal Name	Definition	Signal Source
TD	Transmitted data	DTE
RD	Received data	DCE
RTS	Request to send	DTE
CTS	Clear to send	DCE
DSR	Data set ready	DCE
Signal Ground	Grounding signal	–
CD	Carrier detect	–

Table 19: Serial Transmission Signals *(continued)*

Signal Name	Definition	Signal Source
DTR	Data terminal ready	DTE
RI	Ring indicator	–

When a serial connection is made, a serial line protocol—such as EIA-530, X.21, RS-422/449, RS-232, or V.35—begins controlling the transmission of signals across the line as follows:

1. The DCE transmits a DSR signal to the DTE, which responds with a DTR signal. After this handshake, the link is established and traffic can pass.
2. When the DTE device is ready to receive data, it sets its RTS signal to a marked state (all 1s) to indicate to the DCE that it can transmit data. (If the DTE is not able to receive data—because of buffer conditions, for example—it sets the RTS signal to all 0s.)
3. When the DCE device is ready to receive data, it sets its CTS signal to a marked state to indicate to the DTE that it can transmit data. (If the DCE is not able to receive data, it sets the CTS signal to all 0s.)
4. When the negotiation to send information has taken place, data is transmitted across the transmitted data (TD) and received data (RD) lines:
   - TD line—Line through which data from a DTE device is transmitted to a DCE device
   - RD line—Line through which data from a DCE device is transmitted to a DTE device

   The name of the wire does not indicate the direction of data flow.

The DTR and DSR signals were originally designed to operate as a handshake mechanism. When a serial port is opened, the DTE device sets its DTR signal to a marked state. Similarly, the DCE sets its DSR signal to a marked state. However, because of the negotiation that takes place with the RTS and CTS signals, the DTR and DSR signals are not commonly used.

The carrier detect and ring indicator signals are used to detect connections with remote modems. These signals are not commonly used.

Signal Polarity

Serial interfaces use a balanced (also called differential) protocol signaling technique. Two serial signals are associated with a circuit: the A signal and the B signal. The A signal is denoted with a plus sign (for example, DTR+), and the B signal is denoted with a minus sign (for example, DTR–). If DTR is low, then DTR+ is negative with respect to DTR–. If DTR is high, then DTR+ is positive with respect to DTR–.

By default, all signal polarities are positive, but sometimes they might be reversed. For example, signals might be miswired as a result of reversed polarities.

Serial Clocking Modes

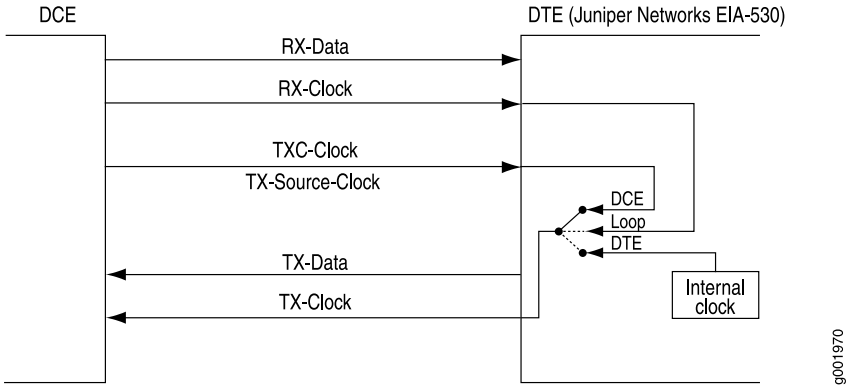
By default, a serial interface uses loop clocking to determine its timing source. For EIA-530 and V.35 interfaces, you can set each port independently to use one of the following clocking modes. X.21 interfaces can use only loop clocking mode.

- Loop clocking mode—Uses the DCE's receive (RX) clock to clock data from the DCE to the DTE.
- DCE clocking mode—Uses the transmit (TXC) clock, generated by the DCE specifically to be used by the DTE as the DTE's transmit clock.
- Internal clocking mode—Uses an internally generated clock. The speed of this clock is configured locally. Internal clocking mode is also known as line timing.

Both loop clocking mode and DCE clocking mode use external clocks generated by the DCE.

Figure 5 on page 39 shows the clock sources for loop, DCE, and internal clocking modes.

Figure 5: Serial Interface Clocking Modes



Serial Interface Transmit Clock Inversion

When an externally timed clocking mode (DCE or loop) is used, long cables might introduce a phase shift of the DTE-transmitted clock and data. At high speeds, this phase shift might cause errors. Inverting the transmit clock corrects the phase shift, thereby reducing error rates.

DTE Clock Rate Reduction

Although the serial interface is intended for use at the default clock rate of 16.384 MHz, you might need to use a slower rate under any of the following conditions:

- The interconnecting cable is too long for effective operation.
- The interconnecting cable is exposed to an extraneous noise source that might cause an unwanted voltage in excess of +1 volt.

The voltage must be measured differentially between the signal conductor and the point in the circuit from which all voltages are measured (“circuit common”) at the load end of the cable, with a 50-ohm resistor substituted for the generator.

- Interference with other signals must be minimized.
- Signals must be inverted.

Serial Line Protocols

Serial interfaces support the following line protocols:

- EIA-530 on page 40
- RS-232 on page 40
- RS-422/449 on page 41
- V.35 on page 42
- X.21 on page 42

EIA-530

EIA-530 is an Electronic Industries Association (EIA) standard for the interconnection of DTE and DCE using serial binary data interchange with control information exchanged on separate control circuits. EIA-530 is also known as RS-530.

The EIA-530 line protocol is a specification for a serial interface that uses a DB-25 connector and balanced equivalents of the RS-232 signals—also called V.24. The EIA-530 line protocol is equivalent to the RS-422 and RS-423 interfaces implemented on a 25-pin connector.

The EIA-530 line protocol supports both balanced and unbalanced modes. In unbalanced transmissions, voltages are transmitted over a single wire. Because only a single signal is transmitted, differences in ground potential can cause fluctuations in the measured voltage across the link. For example, if a 3V signal is sent from one endpoint to another, and the receiving endpoint has a ground potential 1V higher than the transmitter, the signal on the receiving end is measured as a 2V signal.

Balanced transmissions use two wires instead of one. Rather than sending a single signal across the wire and having the receiving end measure the voltage, the transmitting device sends two separate signals across two separate wires. The receiving device measures the difference in voltage of the two signals (balanced sampling) and uses that calculation to evaluate the signal. Any differences in ground potential affect both wires equally, and the difference in the signals is still the same.

The EIA-530 interface supports asynchronous and synchronous transmissions at rates ranging from 20 Kbps to 2 Mbps.

RS-232

RS-232 is a Recommended Standard (RS) describing the most widely used type of serial communication. The RS-232 protocol is used for asynchronous data transfer

as well as synchronous transfers using HDLC, Frame Relay, and X.25. RS-232 is also known as EIA-232.

The RS-232 line protocol is very popular for low-speed data signals. RS-232 signals are carried as single voltages referred to a common ground signal. The voltage output level of these signals varies between –12V and +12V. Within this range, voltages between –3V and +3V are considered inoperative and are used to absorb line noise. Control signals are considered operative when the voltage ranges from +3 to +25V.

The RS-232 line protocol is an unbalanced protocol, because it uses only one wire, and is susceptible to signal degradation. Degradation can be extremely disruptive, particularly when a difference in ground potential exists between the transmitting and receiving ends of a link.

The RS-232 interface is implemented in a 25-pin D-shell connector and supports line rates up to 200 Kbps over lines shorter than 98 feet (30 meters).

NOTE: RS-232 serial interfaces cannot function error-free with a clock rate greater than 200 KHz.

RS-422/449

RS-422 is a Recommended Standard (RS) describing the electrical characteristics of balanced voltage digital interface circuits that support higher bandwidths than traditional serial protocols like RS-232. RS-422 is also known as EIA-422.

The RS-449 standard (also known as EIA-449) is compatible with RS-422 signal levels. The EIA created RS-449 to detail the DB-37 connector pinout and define a set of modem control signals for regulating flow control and line status.

The RS-422/499 line protocol runs in balanced mode, allowing serial communications to extend over distances of up to 4,000 feet (1.2 km) and at very fast speeds of up to 10 Mbps.

In an RS-422/499-based system, a single master device can communicate with up to 10 slave devices in the system. To accommodate this configuration, RS-422/499 supports the following kinds of transmission:

- Half-duplex transmission—In half-duplex transmission mode, transmissions occur in only one direction at a time. Each transmission requires a proper handshake before it is sent. This operation is typical of a balanced system in which two devices are connected by a single connection.
- Full-duplex transmission—In full duplex transmission mode, multiple transmissions can occur simultaneously so that devices can transmit and receive at the same time. This operation is essential when a single master in a point-to-multipoint system must communicate with multiple receivers.
- Multipoint transmission—RS-422/449 allows only a single master in a multipoint system. The master can communicate to all points in a multipoint system, and the other points must communicate with each other through the master.

V.35

V.35 is an ITU-T standard describing a synchronous, physical-layer protocol used for communications between a network access device and a packet network. V.35 is most commonly used in the United States and Europe.

The V.35 line protocol is a mixture of balanced (RS-422) and common ground (RS-232) signal interfaces. The V.35 control signals DTR, DSR, DCD, RTS, and CTS are single-wire common ground signals that are essentially identical to their RS-232 equivalents. Unbalanced signaling for these control signals is sufficient, because the control signals are mostly constant, varying at very low frequency, which makes single-wire transmission suitable. Higher-frequency data and clock signals are sent over balanced wires.

V.35 interfaces operate at line rates of 20 Kbps and above.

X.21

X.21 is an ITU-T standard for serial communications over synchronous digital lines. The X.21 protocol is used primarily in Europe and Japan.

The X.21 line protocol is a state-driven protocol that sets up a circuit-switched network using call setup. X.21 interfaces use a 15-pin connector with the following eight signals:

- Signal ground (G)—Reference signal used to evaluate the logic states of the other signals. This signal can be connected to the protective earth (ground).
- DTE common return (Ga)—Reference ground signal for the DCE interface. This signal is used only in unbalanced mode.
- Transmit (T)—Binary signal that carries the data from the DTE to the DCE. This signal can be used for data transfer or in call-control phases such as Call Connect or Call Disconnect.
- Receive (R)—Binary signal that carries the data from the DCE to the DTE. This signal can be used for data transfer or in call-control phases such as Call Connect or Call Disconnect.
- Control (C)—DTE-controlled signal that controls the transmission on an X.21 link. This signal must be on during data transfer, and can be on or off during call-control phases.
- Indication (I)—DCE-controlled signal that controls the transmission on an X.21 link. This signal must be on during data transfer, and can be on or off during call-control phases.
- Signal Element Timing (S)—Clocking signal that is generated by the DCE. This signal specifies when sampling on the line must occur.
- Byte Timing (B)—Binary signal that is on when data or call-control information is being sampled. When an 8-byte transmission is over, this signal switches to off.

Transmissions across an X.21 link require both the DCE and DTE devices to be in a ready state, indicated by an all 1s transmission on the T and R signals.

ADSL Interface Overview

Asymmetric digital subscriber line (ADSL) technology is part of the *x*DSL family of modem technologies that use existing twisted-pair telephone lines to transport high-bandwidth data. ADSL lines connect service provider networks and customer sites over the "last mile" of the network—the loop between the service provider and the customer site.

ADSL transmission is asymmetric because the downstream bandwidth is typically greater than the upstream bandwidth. A typical ADSL circuit uses bandwidths of 1.5 Mbps to 2.0 Mbps downstream and 16 Kbps upstream. Depending on the length of the copper wire, an ADSL link can have up to 6.1 Mbps downstream and 64 Kbps upstream.

All J-series Services Routers support ADSL, ADSL2, and ADSL2 + , which comply with the following standards:

- For Annex A and B—ITU G.992.1 (ADSL)
- For Annex A only—ANSI T1.413 Issue II, ITU G.992.3 (ADSL2) and ITU G.992.5 (ADSL2 +)
- For Annex B only—ETSI TS 101 388 V1.3

NOTE: J-series devices with ADSL PIMs can use PPP over Ethernet (PPPoE) and PPP over ATM (PPPoA) to connect through ADSL lines only, not for direct ATM connections.

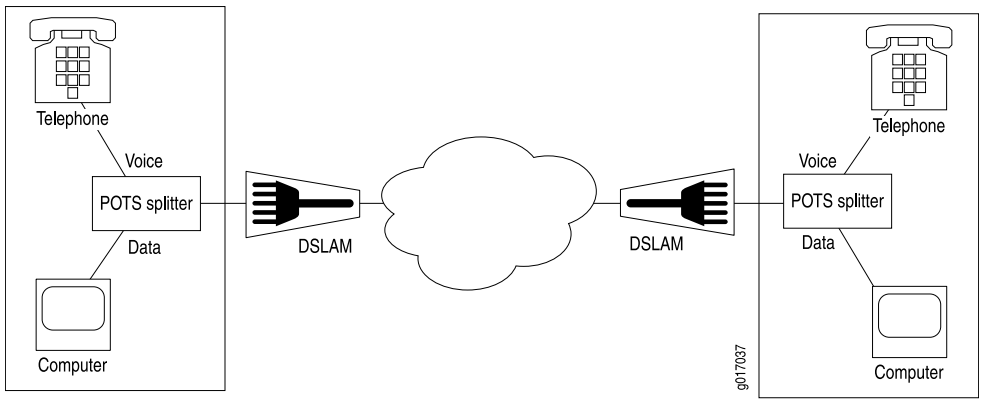
ADSL Systems

ADSL links run across twisted-pair telephone wires. When ADSL modems are connected to each end of a telephone wire, a dual-purpose ADSL circuit can be created. Once established, the circuit can transmit lower-frequency voice traffic and higher-frequency data traffic.

To accommodate both types of traffic, ADSL modems are connected to plain old telephone service (POTS) splitters that filter out the lower-bandwidth voice traffic and the higher-bandwidth data traffic. The voice traffic can be directed as normal telephone voice traffic. The data traffic is directed to the ADSL modem, which is typically connected to the data network.

Because twisted-pair wiring has a length limit, ADSL modems are typically connected to multiplexing devices. DSL access multiplexers (DSLAMs) can process and route traffic from multiple splitters. This typical ADSL configuration is shown in Figure 6 on page 44.

Figure 6: Typical ADSL Topology



ADSL2 and ADSL2+

The ADSL2 and ADSL2 + standards were adopted by the ITU in July 2002. ADSL2 improves the data rate and reach performance, diagnostics, standby mode, and interoperability of ADSL modems.

ADSL2 + doubles the possible downstream data bandwidth, enabling rates of 20 Mbps on telephone lines shorter than 5,000 feet (1.5 km).

First-generation ADSL standards require fixed 32-bit overhead framing on all ADSL packets. On long lines with low rates of 128 Kbps, the overhead represents 25 percent of the available bandwidth. ADSL2 standards allow the overhead per frame to be a programmable value between 4 Kbps and 32 Kbps, to provide up to 28 Kbps more bandwidth for payload data.

ADSL2 uses seamless rate adaptation (SRA) to change the data rate of a connection during operation with no interruptions or bit errors. The ADSL2 transceiver detects changes in channel conditions—for example, the failure of another transceiver in a multicarrier link—and sends a message to the transmitter to initiate a data rate change. The message includes data transmission parameters such as the number of bits modulated and the power on each channel. When the transmitter receives the information, it transitions to the new transmission rate.

Asynchronous Transfer Mode

On a J-series device, the ADSL link is employed over an Asynchronous Transfer Mode (ATM)-over-ADSL interface. Although the interface type is **at**, the physical interface is ADSL. ATM-over-ADSL and ATM-over-SHDSL interfaces can be configured with the properties associated with traditional ATM interfaces, including virtual circuit and path information and ATM encapsulation.

SHDSL Interface Overview

SHDSL interfaces on J-series device support a symmetric, high-speed digital subscriber line (SHDSL) multirate technology for data transfer between a single customer

premises equipment (CPE) subscriber and a central office (CO). ITU-T G.991.2 is the officially designated standard describing SHDSL, also known as G.SHDSL.

Unlike ADSL, which delivers more bandwidth downstream than available upstream, SHDSL is symmetrical and delivers a bandwidth of up to 2.3 Mbps in both directions. Because business applications require higher-speed digital transportation methods, SHDSL is becoming very popular and gaining wide acceptance in the industry. Additionally, SHDSL is compatible with ADSL and therefore causes very little, if any, interference between cables.

SHDSL is deployed on a network in much the same manner as ADSL.

ISDN Interface Overview

The Integrated Services Digital Network (ISDN) technology is a design for a completely digital telecommunications network. ISDN can carry voice, data, images, and video across a telephony network, using a single interface for all transmissions.

ISDN Channels

ISDN uses separate channels to transmit voice and data over the network. Channels operate at bandwidths of either 64 Kbps or 16 Kbps, depending on the type of channel.

Bearer channels (B-channels) use 64 Kbps to transmit voice, data, video, or multimedia information. This bandwidth is derived from the fact that analog voice lines are sampled at a rate of 64 Kbps (8,000 samples per second using 8 bits per sample).

Delta channels (D-channels) are control channels that operate at either 16 Kbps or 64 Kbps. D-channels are used primarily for ISDN signaling between switching equipment in an ISDN network.

ISDN Interfaces

ISDN provides two basic types of service, Basic Rate Interface (BRI) and Primary Rate Interface (PRI). J-series devices support both ISDN BRI and ISDN PRI.

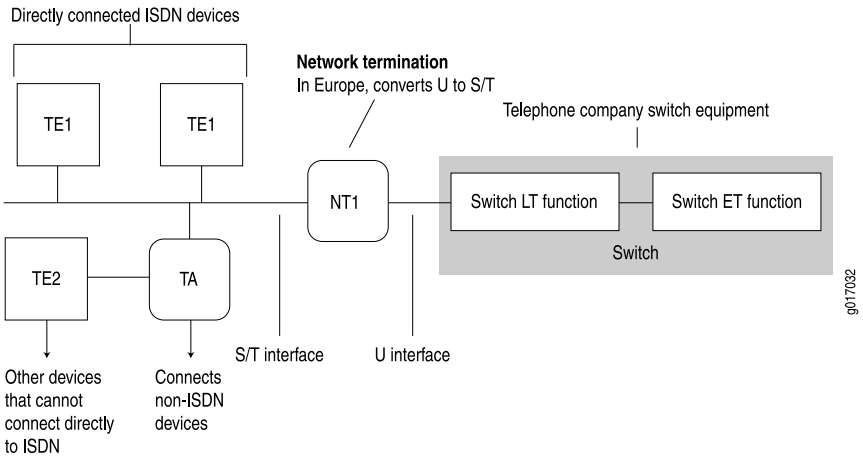
ISDN BRI is designed for high-bandwidth data transmissions through existing telephone lines. The copper wires that make up much of the existing telephony infrastructure can support approximately 160 Kbps, which provides enough bandwidth for two B-channels and a D-channel, leaving 16 Kbps for any data framing, maintenance, and link control overhead.

ISDN PRI is designed for users with greater capacity requirements than can be met with ISDN BRI. In the United States, the most common PRI supports 23 B-channels and 1 D-channel, totalling 1,536 Kbps, which is roughly equivalent to a T1 link. In Europe, the most common PRI supports 30 B-channels and 1 D-channel, totalling 1,984 Kbps, which is roughly equivalent to an E1 link.

Typical ISDN Network

Figure 7 on page 46 shows a typical ISDN network.

Figure 7: ISDN Network



In Figure 7 on page 46, two types of end-user devices are connected to the ISDN network:

- Terminal equipment type 1 (TE1) device—Designed to connect directly through an ISDN telephone line.
- Terminal equipment type 2 (TE2) device—Not designed for ISDN. TE2 devices—for example, analog telephones or modems—must connect to the ISDN network through a terminal adapter (TA).

A terminal adapter allows non-ISDN devices on the ISDN network.

NT Devices and S and T Interfaces

The interface between the ISDN network and a TE1 device or terminal adapter is called an S interface. The S interface connects to a network termination type 2 (NT2) device such as a PBX, or directly to the TE1 device or terminal adapter, as shown in Figure 7 on page 46. The NT2 device is then connected to a network termination type 1 (NT1) device through a T interface. The S and T interfaces are electrically equivalent.

An NT1 device is a physical layer device that connects a home telephone network to a service provider carrier network. ISDN devices that connect to an NT1 device from the home network side use a 4-wire S/T interface. The NT1 device converts the 4-wire S/T interface into the 2-wire U interface that telephone carriers use as their plain old telephone service (POTS) lines.

In the United States, NT1 devices are user owned. In many other countries, NT1 devices are owned by the telephone service providers.

U Interface

The U interface connects the ISDN network into the telephone switch through line termination (LT) equipment. The connection from LT equipment to other switches within the telephone network is called the exchange termination (ET).

ISDN Call Setup

Before traffic can pass through an ISDN network, an ISDN call must be set up. ISDN call setup requires a Layer 2 connection to be initialized and then a Layer 3 session to be established over the connection.

To specify the services and features to be provided by the service provider switch, you must set service profile identifiers (SPIDs) on TE1 devices before call setup and initialization. If you define SPIDs for features that are not available on the ISDN link, Layer 2 initialization takes place, but a Layer 3 connection is not established.

Layer 2 ISDN Connection Initialization

The TE device and the telephone network initialize a Layer 2 connection for ISDN as follows:

1. The TE device and the telephone network exchange Receive Ready (RR) frames, to indicate that they are available for data transmission. A call cannot be set up if either the TE device or telephone network does not transmit RR frames.
2. If both ends of the ISDN connection are available to receive data, the TE device sends an Unnumbered Information (UI) frame to all devices on the ISDN link.
3. When it receives the UI frame, the network responds with a message containing a unique terminal endpoint identifier (TEI) that identifies the endpoint on the ISDN link for all subsequent data transmissions.
4. When the TE device receives the TEI message, it sends out a call setup message.
5. The network sends an acknowledgement of the call setup message.
6. When the TE device receives the acknowledgement, a Layer 2 connection is initialized on the ISDN link.

Layer 3 ISDN Session Establishment

The caller, switch, and receiver establish a Layer 3 ISDN connection as follows:

1. When a Layer 2 connection is initialized, the caller sends a SETUP message to the switch in the telephone network.
2. If the setup is message is valid, the switch responds with a call proceeding (CALL PROC) message to the caller and a SETUP message to the receiver.
3. When the receiver receives the SETUP message, it responds with an ALERTING message to the telephone switch.
4. This ALERTING message is then forwarded to the caller.
5. The receiver then accepts the connection by sending a CONNECT message to the switch.

6. The switch forwards the CONNECT message to the caller.
7. The caller responds with an acknowledgement message (CONNECT ACK).
8. When the CONNECT ACK message is received by the receiver, the ISDN call is set up and traffic can pass.

Interface Physical Properties

The physical properties of a network interface are the characteristics associated with the physical link that affect the transmission of either link-layer signals or the data across the links. Physical properties include clocking properties, transmission properties, such as the maximum transmission unit (MTU), and encapsulation methods, such as point-to-point and Frame Relay encapsulation.

The default property values for an interface are usually sufficient to successfully enable a bidirectional link. However, if you configure a set of physical properties on an interface, those same properties must be set on all adjacent interfaces to which a direct connection is made.

Table 20 on page 48 summarizes some key physical properties of device interfaces.

Table 20: Interface Physical Properties

Physical Property	Description
bert-error-rate	Bit error rate (BER). The error rate specifies the number of bit errors in a particular bit error rate test (BERT) period required to generate a BERT error condition. See "Bit Error Rate Testing" on page 49.
bert-period	Bit error rate test (BERT) time period over which bit errors are sampled. See "Bit Error Rate Testing" on page 49.
chap	Challenge Handshake Authentication Protocol (CHAP). Specifying chap enables CHAP authentication on the interface. See "PPP Authentication" on page 55.
clocking	Clock source for the link. Clocking can be provided by the local system (internal) or a remote endpoint on the link (external). By default, all interfaces use the internal clocking mode. If an interface is configured to accept an external clock source, one adjacent interface must be configured to act as a clock source. Under this configuration, the interface operates in a loop timing mode, in which the clocking signal is unique for that individual network segment or loop. See "Interface Clocking" on page 49.
description	A user-defined text description of the interface, often used to describe the interface's purpose.
disable	Administratively disables the interface.
encapsulation	Type of encapsulation on the interface. Common encapsulation types include PPP, Frame Relay, Cisco HDLC, and PPP over Ethernet (PPPoE). See "Physical Encapsulation on an Interface" on page 52.
fcs	Frame check sequence (FCS). FCS is an error-detection scheme that appends parity bits to a digital signal and uses decoding algorithms that detect errors in the received digital signal. See "Frame Check Sequences" on page 50.

Table 20: Interface Physical Properties *(continued)*

Physical Property	Description
mtu	Maximum transmission unit (MTU) size. The MTU is the largest size packet or frame, specified in bytes or octets, that can be sent in a packet-based or frame-based network. The Transmission Control Protocol (TCP) uses the MTU to determine the maximum size of each packet in any transmission. For MTU values on J-series interfaces, see "MTU Default and Maximum Values" on page 51.
no-keepalives	Disabling of keepalive messages across a physical link. A keepalive message is sent between network devices to indicate that they are still active. Keepalives help determine whether the interface is operating correctly. Except for ATM-over-ADSL interfaces, all interfaces use keepalives by default.
pap	Password Authentication Protocol (PAP). Specifying pap enables PAP authentication on the interface. To configure PAP, use the CLI or J-Web configuration editor. PAP is not available in the J-Web Quick Configuration pages.
payload-scrambler	Scrambling of traffic transmitted out the interface. Payload scrambling randomizes the data payload of transmitted packets. Scrambling eliminates nonvariable bit patterns (strings of all 1s or all 0s) that generate link-layer errors across some physical links.

Bit Error Rate Testing

In telecommunication transmission, the bit error rate (BER) is the percentage of bits that have errors compared to the total number of bits received in a transmission, usually expressed as 10 to a negative power. For example, a transmission with a BER of 10^{-6} received 1 errored bit in 1,000,000 bits transmitted. The BER indicates how often a packet or other data unit must be retransmitted because of an error. If the BER is too high, a slower data rate might improve the overall transmission time for a given amount of data if it reduces the BER and thereby lowers the number of resent packets.

A bit error rate test (BERT) is a procedure or device that measures the BER for a given transmission. You can configure a device to act as a BERT device by configuring the interface with a bit error rate and a testing period. When the interface receives a BERT request from a BER tester, it generates a response in a well-known BERT pattern. The initiating device checks the BERT-patterned response to determine the number of bit errors.

Interface Clocking

Clocking determines how individual routing nodes or entire networks sample transmitted data. As streams of information are received by a device in a network, a clock source specifies when to sample the data. In asynchronous networks, the clock source is derived locally, and synchronous networks use a central, external clock source. Interface clocking indicates whether the device uses asynchronous or synchronous clocking.

NOTE: Because truly synchronous networks are difficult to design and maintain, most synchronous networks are really plesiochronous networks. In a plesiochronous network, different timing regions are controlled by local clocks that are synchronized (with very narrow constraints). Such networks approach synchronicity and are generally known as synchronous networks.

Most networks are designed to operate as asynchronous networks. Each device generates its own clock signal, or devices use clocks from more than one clock source. The clocks within the network are not synchronized to a single clock source. By default, devices generate their own clock signals to send and receive traffic.

The system clock allows the device to sample (or detect) and transmit data being received and transmitted through its interfaces. Clocking enables the device to detect and transmit the 0s and 1s that make up digital traffic through the interface. Failure to detect the bits within a data flow results in dropped traffic.

Short-term fluctuations in the clock signal are known as clock jitter. Long-term variations in the signal are known as clock wander.

Asynchronous clocking can either derive the clock signal from the data stream or transmit the clocking signal explicitly.

Data Stream Clocking

Common in T1 links, data stream clocking occurs when separate clock signals are not transmitted within the network. Instead, devices must extract the clock signal from the data stream. As bits are transmitted across the network, each bit has a time slot of 648 nanoseconds. Within a time slot, pulses are transmitted with alternating voltage peaks and drops. The receiving device uses the period of alternating voltages to determine the clock rate for the data stream.

Explicit Clocking Signal Transmission

Clock signals that are shared by hosts across a data link must be transmitted by one or both endpoints on the link. In a serial connection, for example, one host operates as a clock master and the other operates as a clock slave. The clock master internally generates a clock signal that is transmitted across the data link. The clock slave receives the clock signal and uses its period to determine when to sample data and how to transmit data across the link.

This type of clock signal controls only the connection on which it is active and is not visible to the rest of the network. An explicit clock signal does not control how other devices or even other interfaces on the same device sample or transmit data.

Frame Check Sequences

All packets or frames within a network can be damaged by crosstalk or interference in the network's physical wires. The frame check sequence (FCS) is an extra field in each transmitted frame that can be analyzed to determine if errors have occurred.

The FCS uses cyclic redundancy checks (CRCs), checksums, and two-dimensional parity bits to detect errors in the transmitted frames.

Cyclic Redundancy Checks and Checksums

On a link that uses CRCs for frame checking, the data source uses a predefined polynomial algorithm to calculate a CRC number from the data it is transmitting. The result is included in the FCS field of the frame and transmitted with the data. On the receiving end, the destination host performs the same calculation on the data it receives.

If the result of the second calculation matches the contents of the FCS field, the packet was sent and received without bit errors. If the values do not match, an FCS error is generated, the frame is discarded and the originating host is notified of the error.

Checksums function similarly to CRCs, but use a different algorithm.

Two-Dimensional Parity

On a link that uses two-dimensional parity bits for frame checking, the sending and receiving hosts examine each frame in the total packet transmission and create a parity byte that is evaluated to detect transmission errors.

For example, a host can create the parity byte for the following frame sequence by summing up each column (each bit position in the frame) and keeping only the least-significant bit:

```
Frame 1       0 1 0 1 0 0 1
Frame 2       1 1 0 1 0 0 1
Frame 3       1 0 1 1 1 1 0
Frame 4       0 0 0 1 1 1 0
Frame 5       0 1 1 0 1 0 0
Frame 6       1 0 1 1 1 1 1

Parity Byte   1 1 1 1 0 1 1
```

If the sum of the bit values in a bit position is even, the parity bit for the position is 0. If the sum is odd, the parity bit is 1. This method is called even parity. Matching parity bytes on the originating and receiving hosts indicate that the packet was received without error.

MTU Default and Maximum Values

Table 21 on page 51 lists MTU values for J-series devices.

Table 21: MTU Values for J2320, J2350, J4350, and J6350 Interfaces

J4350 and J6350 Interfaces	Default Media MTU (bytes)	Maximum MTU (bytes)	Default IP MTU (bytes)
Gigabit Ethernet (10/100/1000) built-in interface	1514	9018	1500

Table 21: MTU Values for J2320, J2350, J4350, and J6350 Interfaces *(continued)*

J4350 and J6350 Interfaces	Default Media MTU (bytes)	Maximum MTU (bytes)	Default IP MTU (bytes)
6–Port, 8–Port, and 16–Port Gigabit Ethernet uPIMs	1514	9014	1500
Gigabit Ethernet (10/100/1000) ePIM	1514	9018	1500
Gigabit Ethernet (10/100/1000) SFP ePIM	1514	9018	1500
4-Port Fast Ethernet (10/100) ePIM	1514	1514	1500
Dual-Port Fast Ethernet (10/100) PIM	1514	9192	1500
Dual-Port Serial PIM	1504	9150	1500
Dual-Port T1 or E1 PIM	1504	9192	1500
Dual-Port Channelized T1/E1/ISDN PRI PIM (channelized to DS0s)	1504	4500	1500
Dual-Port Channelized T1/E1/ISDN PRI PIM (clear-channel T1 or E1)	1504	9150	1500
Dual-Port Channelized T1/E1/ISDN PRI PIM (ISDN PRI dialer interface)	1504	4098	1500
T3 (DS3) or E3 PIM	4474	9192	4470
4-Port ISDN BRI PIM	1504	4092	1500
ADSL + 2 PIM	4482	9150	4470
G.SHDSL PIM	4482	9150	4470

Physical Encapsulation on an Interface

Encapsulation is the process by which a lower-level protocol accepts a message from a higher-level protocol and places it in the data portion of the lower-level frame. As a result, datagrams transmitted through a physical network have a sequence of headers: the first header for the physical network (or data link layer) protocol, the second header for the network layer protocol (IP, for example), the third header for the transport protocol, and so on.

The following encapsulation protocols are supported on device physical interfaces:

- Frame Relay on page 53
- Point-to-Point Protocol on page 54
- Point-to-Point Protocol over Ethernet on page 57
- High-Level Data Link Control on page 58

Frame Relay

The Frame Relay packet-switching protocol operates at the physical and data link layers in a network to optimize packet transmissions by creating virtual circuits between hosts. Figure 8 on page 53 shows a typical Frame Relay network.

Figure 8: Frame Relay Network

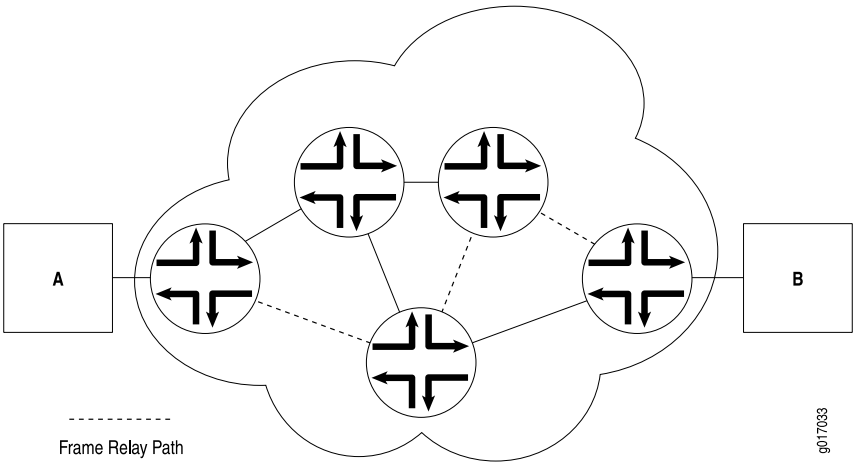


Figure 8 on page 53 shows multiple paths from Host A to Host B. In a typical routed network, traffic is sent from device to device with each device making routing decisions based on its own routing table. In a packet-switched network, the paths are predefined. Devices switch a packet through the network according to predetermined next-hops established when the virtual circuit is set up.

Virtual Circuits

A virtual circuit is a bidirectional path between two hosts in a network. Frame Relay virtual circuits are logical connections between two hosts that are established either by a call setup mechanism or by explicit configuration.

A virtual circuit created through a call setup mechanism is known as a switched virtual circuit (SVC). A virtual circuit created through explicit configuration is called a permanent virtual circuit (PVC).

Switched and Permanent Virtual Circuits

Before data can be transmitted across an SVC, a signaling protocol like ISDN must set up a call by the exchange of setup messages across the network. When a connection is established, data is transmitted across the SVC. After data transmission, the circuit is torn down and the connection is lost. For additional traffic to pass between the same two hosts, a subsequent SVC must be established, maintained, and terminated.

Because PVCs are explicitly configured, they do not require the setup and teardown of SVCs. Data can be switched across the PVC whenever a host is ready to transmit.

SVCs are useful in networks where data transmission is sporadic and a permanent circuit is not needed.

Data-Link Connection Identifiers

An established virtual circuit is identified by a data-link connection identifier (DLCI). The DLCI is a value from 16 through 1022. (Values 1 through 15 are reserved.) The DLCI uniquely identifies a virtual circuit locally so that devices can switch packets to the appropriate next-hop address in the circuit. Multiple paths that pass through the same transit devices have different DLCIs and associated next-hop addresses.

Congestion Control and Discard Eligibility

Frame Relay uses the following types of congestion notification to control traffic within a Frame Relay network. Both are controlled by a single bit in the Frame Relay header.

- Forward-explicit congestion notification (FECN)
- Backward-explicit congestion notification (BECN)

Traffic congestion is typically defined in the buffer queues on a device. When the queues reach a predefined level of saturation, traffic is determined to be congested. When traffic congestion occurs in a virtual circuit, the device experiencing congestion sets the congestion bits in the Frame Relay header to 1. As a result, transmitted traffic has the FECN bit set to 1, and return traffic on the same virtual circuit has the BECN bit set to 1.

When the FECN and BECN bits are set to 1, they provide a congestion notification to the source and destination devices. The devices can respond in either of two ways: to control traffic on the circuit by sending it through other routes, or to reduce the load on the circuit by discarding packets.

If devices discard packets as a means of congestion (flow) control, Frame Relay uses the discard eligibility (DE) bit to give preference to some packets in discard decisions. A DE value of 1 indicates that the frame is of lower importance than other frames and more likely to be dropped during congestion. Critical data (such as signaling protocol messages) without the DE bit set is less likely to be dropped.

Point-to-Point Protocol

The Point-to-Point Protocol (PPP) is an encapsulation protocol for transporting IP traffic across point-to-point links. PPP is made up of three primary components:

- Link control protocol (LCP)—Establishes working connections between two points.
- Authentication protocols—Enable secure connections between two points.
- Network control protocols (NCPs)—Initialize the PPP protocol stack to handle multiple network layer protocols, such as IPv4, IPv6, and Connectionless Network Protocol (CLNP).

Link Control Protocol

LCP is responsible for establishing, maintaining, and tearing down a connection between two endpoints. LCP also tests the link and determines whether it is active. LCP establishes a point-to-point connection as follows:

1. LCP must first detect a clocking signal on each endpoint. However, because the clocking signal can be generated by a network clock and shared with devices on the network, the presence of a clocking signal is only a preliminary indication that the link might be functioning.
2. When a clocking signal is detected, a PPP host begins transmitting PPP Configure-Request packets.
3. If the remote endpoint on the point-to-point link receives the Configure-Request packet, it transmits a Configure-Acknowledgement packet to the source of the request.
4. After receiving the acknowledgement, the initiating endpoint identifies the link as established. At the same time, the remote endpoint sends its own request packets and processes the acknowledgement packets. In a functioning network, both endpoints treat the connection as established.

During connection establishment, LCP also negotiates connection parameters such as FCS and HDLC framing. By default, PPP uses a 16-bit FCS, but you can configure PPP to use either a 32-bit FCS or a 0-bit FCS (no FCS). Alternatively, you can enable HDLC encapsulation across the PPP connection.

After a connection is established, PPP hosts generate Echo-Request and Echo-Response packets to maintain a PPP link.

PPP Authentication

PPP's authentication layer uses a protocol to help ensure that the endpoint of a PPP link is a valid device. Authentication protocols include the Password Authentication Protocol (PAP), the Extensible Authentication Protocol (EAP), and the Challenge Handshake Authentication Protocol (CHAP). CHAP is the most commonly used.

NOTE: EAP is not currently supported on J-series devices. PAP is supported, but must be configured from the CLI or J-Web configuration editor. PAP is not configurable from the J-Web Quick Configuration pages.

CHAP ensures secure connections across PPP links. After a PPP link is established by LCP, the PPP hosts at either end of the link initiate a three-way CHAP handshake. Two separate CHAP handshakes are required before both sides identify the PPP link as established.

CHAP configuration requires each endpoint on a PPP link to use a shared secret (password) to authenticate challenges. The shared secret is never transmitted over the wire. Instead, the hosts on the PPP connection exchange information that enables both to determine that they share the same secret. Challenges consist of a hash function calculated from the secret, a numeric identifier, and a randomly chosen

challenge value that changes with each challenge. If the response value matches the challenge value, authentication is successful. Because the secret is never transmitted and is required to calculate the challenge response, CHAP is considered very secure.

PAP authentication protocol uses a simple 2-way handshake to establish identity. PAP is used after the link establishment phase (LCP up), during the authentication phase. JUNOS software can support PAP in one direction (egress or ingress), and CHAP in the other.

Network Control Protocols

After authentication is completed, the PPP connection is fully established. At this point, any higher-level protocols (for example, IP protocols) can initialize and perform their own negotiations and authentication.

PPP NCPs include support for the following protocols. IPCP and IPV6CP are the most widely used on J-series devices.

- ATCP—AppleTalk Control Protocol
- BCP—Bridging Control Protocol
- BVCP—Banyan Vines Control Protocol
- DNCP—DECnet Phase IV Control Protocol
- IPCP—IP Control Protocol
- IPV6CP—IPv6 Control Protocol
- IPXCP—Novell IPX Control Protocol
- LECP—LAN Extension Control Protocol
- NBFCP—NetBIOS Frames Control Protocol
- OSINLCP—OSI Network Layer Control Protocol (includes IS-IS, ES-IS, CLNP, and IDRP)
- SDTP—Serial Data Transport Protocol
- SNACP—Systems Network Architecture (SNA) Control Protocol
- XNSCP—Xerox Network Systems (XNS) Internet Datagram Protocol (IDP) Control Protocol

Magic Numbers

Hosts running PPP can create “magic” numbers for diagnosing the health of a connection. A PPP host generates a random 32-bit number and sends it to the remote endpoint during LCP negotiation and echo exchanges.

In a typical network, each host's magic number is different. A magic number mismatch in an LCP message informs a host that the connection is not in loopback mode and traffic is being exchanged bidirectionally. If the magic number in the LCP message is the same as the configured magic number, the host determines that the connection is in loopback mode, with traffic looped back to the transmitting host.

Looping traffic back to the originating host is a valuable way to diagnose network health between the host and the loopback location. To enable loopback testing, telecommunications equipment typically supports channel service unit/data service unit (CSU/DSU) devices.

CSU/DSU Devices

A channel service unit (CSU) connects a terminal to a digital line. A data service unit (DSU) performs protective and diagnostic functions for a telecommunications line. Typically, the two devices are packaged as a single unit. A CSU/DSU device is required for both ends of a T1 or T3 connection, and the units at both ends must be set to the same communications standard.

A CSU/DSU device enables frames sent along a link to be looped back to the originating host. Receipt of the transmitted frames indicates that the link is functioning correctly up to the point of loopback. By configuring CSU/DSU devices to loop back at different points in a connection, network operators can diagnose and troubleshoot individual segments in a circuit.

Point-to-Point Protocol over Ethernet

Point-to-Point Protocol over Ethernet (PPPoE) combines PPP, which is typically run over broadband connections, with the Ethernet link-layer protocol that allows users to connect to a network of hosts over a bridge or access concentrator. PPPoE enables service providers to maintain access control through PPP connections and also manage multiple hosts at a remote site.

To provide a PPPoE connection, each PPP session must learn the Ethernet address of the remote peer and establish a unique session identifier during the PPPoE discovery and session stages.

PPPoE Discovery

To initiate a PPPoE session, a host must first identify the Ethernet MAC address of the remote peer and establish a unique PPPoE session ID for the session. Learning the remote Ethernet MAC address is called PPPoE discovery.

During the PPPoE discovery process, the host does not discover a remote endpoint on the Ethernet network. Instead, the host discovers the access concentrator through which all PPPoE sessions are established. Discovery is a client/server relationship, with the host (a J-series device) acting as the client and the access concentrator acting as the server.

The PPPoE discovery stage consists of the following steps:

1. PPPoE Active Discovery Initiation (PADI)—The client initiates a session by broadcasting a PADI packet to the LAN, to request a service.
2. PPPoE Active Discovery Offer (PADO)—Any access concentrator that can provide the service requested by the client in the PADI packet replies with a PADO packet that contains it own name, the unicast address of the client, and the service requested. An access concentrator can also use the PADO packet to offer other services to the client.

3. PPPoE Active Discovery Request (PADR)—From the PADOs it receives, the client selects one access concentrator based on its name or the services offered and sends it a PADR packet to indicate the service or services needed.
4. PPPoE Active Discovery Session-Confirmation (PADS)—When the selected access concentrator receives the PADR packet, it accepts or rejects the PPPoE session:
   - To accept the session, the access concentrator sends the client a PADS packet with a unique session ID for a PPPoE session and a service name that identifies the service under which it accepts the session.
   - To reject the session, the access concentrator sends the client a PADS packet with a service name error and resets the session ID to zero.

PPPoE Sessions

The PPPoE session stage starts after the PPPoE discovery stage is over. Each PPPoE session is uniquely identified by the Ethernet address of the peer and the session ID. After the PPPoE session is established, data is sent as in any other PPP encapsulation. The PPPoE information is encapsulated within an Ethernet frame and is sent to a unicast address. Magic numbers, echo requests, and all other PPP traffic behave exactly as in normal PPP sessions. In this stage, both the client and the server must allocate resources for the PPPoE logical interface.

After a session is established, the client or the access concentrator can send a PPPoE Active Discovery Termination (PADT) packet anytime to terminate the session. The PADT packet contains the destination address of the peer and the session ID of the session to be terminated. After this packet is sent, the session is closed to PPPoE traffic.

High-Level Data Link Control

High-Level Data Link Control (HDLC) is a bit-oriented, switched and nonswitched link-layer protocol. HDLC is widely used because it supports half-duplex and full-duplex connections, point-to-point and point-to-multipoint networks, and switched and nonswitched channels.

HDLC Stations

Nodes within a network running HDLC are called stations. HDLC supports three types of stations for data link control:

- Primary stations—Responsible for controlling the secondary and combined other stations on the link. Depending on the HDLC mode, the primary station is responsible for issuing acknowledgement packets to allow data transmission from secondary stations.
- Secondary stations—Controlled by the primary station. Under normal circumstances, secondary stations cannot control data transmission across the link with the primary station, are active only when requested by the primary station, and can respond to the primary station only (not to other secondary stations). All secondary station frames are response frames.

- Combined stations—A combination of primary and secondary stations. On an HDLC link, all combined stations can send and receive commands and responses without any permission from any other stations on the link and cannot be controlled by any other station.

HDLC Operational Modes

HDLC runs in three separate modes:

- Normal Response Mode (NRM)—The primary station on the HDLC link initiates all information transfers with secondary stations. A secondary station on the link can transmit a response of one or more information frames only when it receives explicit permission from the primary station. When the last frame is transmitted, the secondary station must wait for explicit permission before it can transmit more frames.

  NRM is used most widely for point-to-multipoint links, in which a single primary station controls many secondary stations.

- Asynchronous Response Mode (ARM)—The secondary station can transmit either data or control traffic at any time, without explicit permission from the primary station. The primary station is responsible for error recovery and link setup, but the secondary station can transmit information at any time.

  ARM is used most commonly with point-to-point links, because it reduces the overhead on the link by eliminating the need for control packets.

- Asynchronous Balance Mode (ABM)—All stations are combined stations. Because no other station can control a combined station, all stations can transmit information without explicit permission from any other station. ABM is not a widely used HDLC mode.

Interface Logical Properties

The logical properties of an interface are the characteristics that do not apply to the physical interface or the wires connected to it. Logical properties include the protocol families running on the interface (including any protocol-specific MTUs), the IP address or addresses associated with the interface, virtual LAN (VLAN) tagging, and any firewall filters or routing policies that are operating on the interface.

The IP specification requires a unique address on every interface of each system attached to an IP network, so that traffic can be correctly routed. Individual hosts such as home computers must have a single IP address assigned. Devices must have a unique IP address for every interface.

This section contains the following topics:

Protocol Families

A protocol family is a group of logical properties within an interface configuration. Protocol families include all the protocols that make up a protocol suite. To use a protocol within a particular suite, you must configure the entire protocol family as a logical property for an interface. The protocol families include common and not-so-common protocol suites.

Common Protocol Suites

JUNOS protocol families include the following common protocol suites:

- Inet—Supports IP protocol traffic, including Open Shortest Path First (OSPF), Border Gateway Protocol (BGP), and Internet Control Message Protocol (ICMP).
- Inet6—Supports IPv6 protocol traffic, including Routing Information Protocol for IPv6 (RIPng), Intermediate System-to-Intermediate System (IS-IS), and BGP.
- ISO—Supports IS-IS traffic.
- MPLS—Supports Multiprotocol Label Switching (MPLS).

If your device is operating in secure context, the Inet6, ISO, and MPLS protocol families are disabled on the device by default. You must enable these protocol families for a device in secure mode to forward IPv6, IS-IS, and MPLS packets. For more information, see "Enabling IPv6 in Secure Context" on page 65, "Configuring the IS-IS Protocol" on page 379, and "Enabling MPLS" on page 423.

CAUTION: Because MPLS is operating in packet mode, security services are not available.

NOTE: JUNOS software security processing is not applied to IPv6 or IS-IS packets forwarded by the device.

Other Protocol Suites

In addition to the common protocol suites, JUNOS protocol families sometimes use the following protocol suites:

- ccc—Circuit cross-connect (CCC).
- mlfr-uni-nni—Multilink Frame Relay (MLFR) FRF.16 user-to-network network-to-network (UNI NNI).
- mlfr-end-to-end—Multilink Frame Relay end-to-end.
- mlppp—Mulitlink Point-to-Point Protocol.
- tcc—Translational cross-connect (TCC).
- tnp—Trivial Network Protocol. This Juniper Networks proprietary protocol provides communication between the Routing Engine and the device's packet forwarding

components. The JUNOS software automatically configures this protocol family on the device's internal interfaces only.

IPv4 Addressing

IPv4 addresses are 32-bit numbers that are typically displayed in dotted decimal notation. A 32-bit address contains two primary parts: the network prefix and the host number.

All hosts within a single network share the same network address. Each host also has an address that uniquely identifies it. Depending on the scope of the network and the type of device, the address is either globally or locally unique. Devices that are visible to users outside the network (Web servers, for example) must have a globally unique IP address. Devices that are visible only within the network (J-series devices, for example) must have locally unique IP addresses.

IP addresses are assigned by a central numbering authority called the Internet Assigned Numbers Authority (IANA). IANA ensures that addresses are globally unique where needed and has a large address space reserved for use by devices not visible outside their own networks.

IPv4 Classful Addressing

To provide flexibility in the number of addresses distributed to networks of different sizes, 4-octet (32-bit) IP addresses were originally divided into three different categories or classes: class A, class B, and class C. Each address class specifies a different number of bits for its network prefix and host number:

- Class A addresses use only the first byte (octet) to specify the network prefix, leaving 3 bytes to define individual host numbers.
- Class B addresses use the first 2 bytes to specify the network prefix, leaving 2 bytes to define host addresses.
- Class C addresses use the first 3 bytes to specify the network prefix, leaving only the last byte to identify hosts.

In binary format, with an x representing each bit in the host number, the three address classes can be represented as follows:

```
00000000 xxxxxxxx xxxxxxxx xxxxxxxx (Class A)
00000000 00000000 xxxxxxxx xxxxxxxx (Class B)
00000000 00000000 00000000 xxxxxxxx (Class C)
```

Because each bit (x) in a host number can have a 0 or 1 value, each represents a power of 2. For example, if only 3 bits are available for specifying the host number, only the following host numbers are possible:

```
111 110 101 100 011 010 001 000
```

In each IP address class, the number of host-number bits raised to the power of 2 indicates how many host numbers can be created for a particular network prefix. Class A addresses have 2^{24} (or 16,777,216) possible host numbers, class B addresses

have 2^{16} (or 65,536) host numbers, and class C addresses have 2^{8} (or 256) possible host numbers.

IPv4 Dotted Decimal Notation

The 32-bit IPv4 addresses are most often expressed in dotted decimal notation, in which each octet (or byte) is treated as a separate number. Within an octet, the rightmost bit represents 2^{0} (or 1), increasing to the left until the first bit in the octet is 2^{7} (or 128). Following are IP addresses in binary format and their dotted decimal equivalents:

```
11010000 01100010 11000000 10101010 = 208.98.192.170
01110110 00001111 11110000 01010101 = 118.15.240.85
00110011 11001100 00111100 00111011 = 51.204.60.59
```

IPv4 Subnetting

Because of the physical and architectural limitations on the size of networks, you often must break large networks into smaller subnetworks. Within a network, each wire or ring requires its own network number and identifying subnet address.

Figure 9 on page 62 shows two subnets in a network.

Figure 9: Subnets in a Network

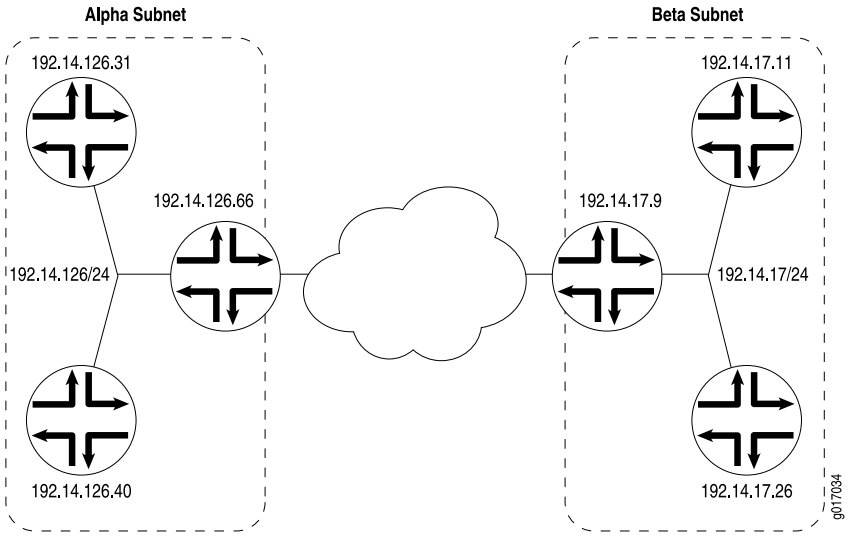


Figure 9 on page 62 shows three devices connected to one subnet and three more devices connected to a second subnet. Collectively, the six devices and two subnets make up the larger network. In this example, the network is assigned the network prefix 192.14.0.0, a class B address. Each device has an IP address that falls within this network prefix.

In addition to sharing a network prefix (the first two octets), the devices on each subnet share a third octet. The third octet identifies the subnet. All devices on a

subnet must have the same subnet address. In this case, the alpha subnet has the IP address 192.14.126.0 and the beta subnet has the IP address 192.14.17.0.

The subnet address 192.14.17.0 can be represented as follows in binary notation:

```
11000000 . 00001110 . 00010001 . xxxxxxxx
```

Because the first 24 bits in the 32-bit address identify the subnet, the last 8 bits are not significant. To indicate the subnet, the address is written as 192.14.17.0/24 (or just 192.14.17/24). The /24 is the subnet mask (sometimes shown as 255.255.255.0).

IPv4 Variable-Length Subnet Masks

Traditionally, subnets were divided by address class. Subnets had either 8, 16, or 24 significant bits, corresponding to 2^{24}, 2^{16}, or 2^{8} possible hosts. As a result, an entire /16 subnet had to be allocated for a network that required only 400 addresses, wasting 65,136 ($2^{16} - 400 = 65{,}136$) addresses.

To help allocate address spaces more efficiently, variable-length subnet masks (VLSMs) were introduced. Using VLSM, network architects can allocate more precisely the number of addresses required for a particular subnet.

For example, suppose a network with the prefix 192.14.17/24 is divided into two smaller subnets, one consisting of 18 devices and the other of 46 devices.

To accommodate 18 devices, the first subnet must have 2^{5} (32) host numbers. Having 5 bits assigned to the host number leaves 27 bits of the 32-bit address for the subnet. The IP address of the first subnet is therefore 192.14.17.128/27, or the following in binary notation:

```
11000000 . 00001110 . 00010001 . 100xxxxx
```

The subnet mask includes 27 significant digits.

To create the second subnet of 46 devices, the network must accommodate 2^{6} (64) host numbers. The IP address of the second subnet is 192.14.17.64/26, or

```
11000000 . 00001110 . 00010001 . 01xxxxxx
```

By assigning address bits within the larger /24 subnet mask, you create two smaller subnets that use the allocated address space more efficiently.

IPv6 Addressing

To create a much larger address space and relieve a projected future shortage of IP addresses, IPv6 was created. IPv6 addresses consist of 128 bits, instead of 32 bits, and include a scope field that identifies the type of application suitable for the address. IPv6 does not support broadcast addresses, but instead uses multicast addresses for broadcast. In addition, IPv6 defines a new type of address called anycast.

On devices in secure context, IPv6 is disabled and must be explicitly enabled.

IPv6 Address Representation

IPv6 addresses consist of 8 groups of 16-bit hexadecimal values separated by colons (:). IPv6 addresses have the following format:

```
aaaa:aaaa:aaaa:aaaa:aaaa:aaaa:aaaa:aaaa
```

Each aaaa is a 16-bit hexadecimal value, and each a is a 4-bit hexadecimal value. Following is a sample IPv6 address:

```
3FFE:0000:0000:0001:0200:F8FF:FE75:50DF
```

You can omit the leading zeros of each 16-bit group, as follows:

```
3FFE:0:0:1:200:F8FF:FE75:50DF
```

You can compress 16-bit groups of zeros to double colons (::) as shown in the following example, but only once per address:

```
3FFE::1:200:F8FF:FE75:50DF
```

IPv6 Address Types

IPv6 has three types of addresses:

- Unicast—For a single interface.
- Multicast—For a set of interfaces on the same physical medium. A packet is sent to all interfaces associated with the address.
- Anycast—For a set of interfaces on different physical media. A packet is sent to only one of the interfaces associated with this address, not to all the interfaces.

IPv6 Address Scope

Unicast and multicast IPv6 addresses support feature called address scoping that identifies the application suitable for the address.

Unicast addresses support global address scope and two types of local address scope:

- Link-local unicast addresses—Used only on a single network link. The first 10 bits of the prefix identify the address as a link-local address. Link-local addresses cannot be used outside the link.

- Site-local unicast addresses—Used only within a site or intranet. A site consists of multiple network links. Site-local addresses identify nodes inside the intranet and cannot be used outside the site.

Multicast addresses support 16 different types of address scope, including node, link, site, organization, and global scope. A 4-bit field in the prefix identifies the address scope.

IPv6 Address Structure

Unicast addresses identify a single interface. Each unicast address consists of *n* bits for the prefix, and 128 – *n* bits for the interface ID.

Multicast addresses identify a set of interfaces. Each multicast address consists of the first 8 bits of all 1s, a 4-bit flags field, a 4-bit scope field, and a 112-bit group ID:

```
11111111 | flgs | scop | group ID
```

The first octet of 1s identifies the address as a multicast address. The flags field identifies whether the multicast address is a well-known address or a transient multicast address. The scope field identifies the scope of the multicast address. The 112-bit group ID identifies the multicast group.

Similar to multicast addresses, anycast addresses identify a set of interfaces. However, packets are sent to only one of the interfaces, not to all interfaces. Anycast addresses are allocated from the normal unicast address space and cannot be distinguished from a unicast address in format. Therefore, each member of an anycast group must be configured to recognize certain addresses as anycast addresses.

Enabling IPv6 in Secure Context

If your device is in secure context, you must explicitly enable IPv6. By default in secure context, the device drops IPv6 packets. You can enable the device in one of the following ways to forward IPv6 packets:

- In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. To reach the correct J-Web page, select **Configure** or **Edit** next to Security, Forwarding options, Family, and finally Inet6. Next to Mode, select **packet-based**. Click **OK**.
- From configuration mode in the CLI, enter the command set security forwarding-options family inet6 mode packet-based.

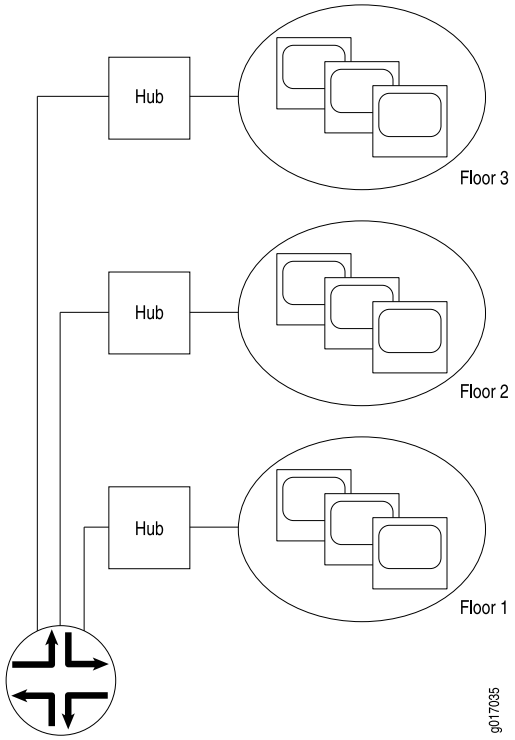
NOTE: JUNOS software security processing is not applied to IPv6 packets forwarded by the device.

Virtual LANs

A local area network (LAN) is a single broadcast domain. When traffic is broadcast, all hosts within the LAN receive the broadcast traffic. A LAN is determined by the physical connectivity of devices within the domain.

Within a traditional LAN, hosts are connected by a hub or repeater that propagates any incoming traffic throughout the network. Each host and its connecting hubs or repeaters make up a LAN segment. LAN segments are connected through switches and bridges to form the broadcast domain of the LAN. Figure 10 on page 66 shows a typical LAN topology.

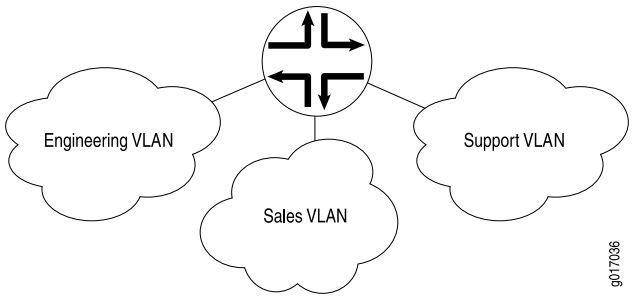
Figure 10: Typical LAN



Virtual LANs (VLANs) allow network architects to segment LANs into different broadcast domains based on logical groupings. Because the groupings are logical, the broadcast domains are not determined by the physical connectivity of the devices in the network. Hosts can be grouped according to a logical function, to limit the traffic broadcast within the VLAN to only the devices for which the traffic is intended.

Suppose a corporate network has three major organizations: engineering, sales, and support. Using VLAN tagging, hosts within each organization can be tagged with a different VLAN identifier. Traffic sent to the broadcast domain is then checked against the VLAN identifier and broadcast to only the devices in the appropriate VLAN. Figure 11 on page 67 shows a typical VLAN topology.

Figure 11: Typical VLAN



Special Interfaces

In addition to the configured network interfaces associated with the physical ports and wires that make up much of the network, devices have special interfaces. Table 22 on page 67 lists each special interface and briefly describes its use.

For information about interface names, see "Network Interface Naming" on page 16.

Table 22: Special Interfaces

Interface Name	Description
dsc	Discard interface. See "Discard Interface" on page 70.
fxp0	In a J-series Services Router chassis cluster configuration, configurable management interfaces are created from built-in interfaces on the connected J-series chassis. The fxp0 interface is the management port, and fxp1 is used as the control link interface in a chassis cluster. In an SRX-series services gateway, the fxp0 management interface is a dedicated port located on the Routing Engine. In an SRX-series services gateway chassis cluster configuration, the control link interface must be port 0 on an SPC. For each node in the chassis cluster, you must configure the SPC that is used for the control link interface. For more information about chassis clusters, see the *JUNOS Software Security Configuration Guide*. For more information about the device management port interfaces, see "Management Interface" on page 70.
gr-0/0/0	Configurable generic routing encapsulation (GRE) interface. GRE allows the encapsulation of one routing protocol over another routing protocol. Within a J-series device, packets are routed to this internal interface, where they are first encapsulated with a GRE packet and then re-encapsulated with another protocol packet to complete the GRE. The GRE interface is an internal interface only and is not associated with a physical medium or PIM. You must configure the interface for it to perform GRE.
gre	Internally generated GRE interface. This interface is generated by the JUNOS software to handle GRE. It is not a configurable interface.

Table 22: Special Interfaces *(continued)*

Interface Name	Description
ip-0/0/0	Configurable IP-over-IP encapsulation (also called IP tunneling) interface. IP tunneling allows the encapsulation of one IP packet over another IP packet. Generally, IP routing allows packets to be routed directly to a particular address. However, in some instances you might need to route an IP packet to one address and then encapsulate it for forwarding to a different address. In a mobile environment in which the location of the end device changes, a different IP address might be used as the end device migrates between networks. Within a J-series device, packets are routed to this internal interface where they are encapsulated with an IP packet and then forwarded to the encapsulating packet's destination address. The IP-IP interface is an internal interface only and is not associated with a physical medium or PIM. You must configure the interface for it to perform IP tunneling.
ipip	Internally generated IP-over-IP interface. This interface is generated by the JUNOS software to handle IP-over-IP encapsulation. It is not a configurable interface.
lo0	Loopback address. The loopback address has several uses, depending on the particular JUNOS feature being configured. See “Loopback Interface” on page 70.
lo0.16384	Internal loopback address. The internal loopback address is a particular instance of the loopback address with the logical unit number 16384. It is created by the JUNOS software as the loopback interface for the internal routing instance. This interface prevents any filter on lo0.0 from disrupting internal traffic.
ls-0/0/0	Configurable link services interface. Link services include the multilink services MLPPP, MLFR, and Compressed Real-Time Transport Protocol (CRTP). Within a J-series device, packets are routed to this internal interface for link bundling or compression. The link services interface is an internal interface only and is not associated with a physical medium or PIM. You must configure the interface for it to perform multilink services. For more information about multilink services, see “Services Interfaces” on page 71.
lsi	Internally generated link services interface. This interface is generated by the JUNOS software to handle multilink services like MLPPP, MLFR, and CRTP. It is not a configurable interface.
lt-0/0/0	Interface used to provide class-of-service (CoS) support for real-time performance monitoring (RPM) probe packets. Within a J-series device, packets are routed to this internal interface for services. The lt interface is an internal interface only and is not associated with a physical medium or PIM. You must configure the interface for it to perform CoS for RPM services. **NOTE:** The lt interface on the M-series and T-series routing platforms supports configuration of logical devices—the capability to partition a single physical device into multiple logical devices that perform independent routing tasks. However, the lt interface on the J-series device does not support logical devices.
pc-*pim*/0/0	Internally configured interface used by the system as a control path between the WXC Integrated Services Module and the Routing Engine.

Table 22: Special Interfaces *(continued)*

Interface Name	Description
pd-0/0/0	Configurable Protocol Independent Multicast (PIM) de-encapsulation interface. In PIM sparse mode, the first-hop routing platform encapsulates packets destined for the rendezvous point device. The packets are encapsulated with a unicast header and are forwarded through a unicast tunnel to the rendezvous point. The rendezvous point then de-encapsulates the packets and transmits them through its multicast tree. Within a J-series device, packets are routed to this internal interface for de-encapsulation. The PIM de-encapsulation interface is an internal interface only and is not associated with a physical medium or Physical Interface Module (PIM). You must configure the interface for it to perform PIM de-encapsulation.
pe-0/0/0	Configurable Protocol Independent Multicast (PIM) encapsulation interface. In PIM sparse mode, the first-hop routing platform encapsulates packets destined for the rendezvous point device. The packets are encapsulated with a unicast header and are forwarded through a unicast tunnel to the rendezvous point. The rendezvous point then de-encapsulates the packets and transmits them through its multicast tree. Within a J-series device, packets are routed to this internal interface for encapsulation. The PIM encapsulation interface is an internal interface only and is not associated with a physical medium or Physical Interface Module (PIM). You must configure the interface for it to perform PIM encapsulation.
pimd	Internally generated Protocol Independent Multicast (PIM) de-encapsulation interface. This interface is generated by the JUNOS software to handle PIM de-encapsulation. It is not a configurable interface.
pime	Internally generated Protocol Independent Multicast (PIM) encapsulation interface. This interface is generated by the JUNOS software to handle PIM encapsulation. It is not a configurable interface.
pp0	Configurable PPPoE encapsulation interface. PPP packets being routed in an Ethernet network use PPPoE encapsulation. Within a J-series device, packets are routed to this internal interface for PPPoE encapsulation. The PPPoE encapsulation interface is an internal interface only and is not associated with a physical medium or PIM. You must configure the interface for it to forward PPPoE traffic. For more information about PPPoE interfaces, see “Configuring Point-to-Point Protocol over Ethernet” on page 157.
st0	Secure tunnel interface used for IPSec VPNs.
tap	Internally generated interface. This interface is generated by the JUNOS software to monitor and record traffic during passive monitoring. When packets are discarded by the Packet Forwarding Engine, they are placed on this interface. It is not a configurable interface.
umd0	Configurable USB modem physical interface. This interface is detected when an USB modem is connected to the USB port on the device. **NOTE:** The J4350 and J6350 devices have two USB ports. However, you can connect only one USB modem to the USB ports on these devices. If you connect USB modems to both the USB ports, only the first USB modem connected to the device is recognized.

Discard Interface

The discard (dsc) interface is not a physical interface, but a virtual interface that discards packets. You can configure one discard interface. This interface allows you to identify the ingress (inbound) point of a denial-of-service (DoS) attack. When your network is under attack, the target host IP address is identified, and the local policy forwards attacking packets to the discard interface. Traffic routed out the discard interface is silently discarded.

Loopback Interface

The Internet Protocol (IP) specifies a loopback network with the (IPv4) address 127.0.0.0/8. Most IP implementations support a loopback interface (lo0) to represent the loopback facility. Any traffic that a computer program sends on the loopback network is addressed to the same computer. The most commonly used IP address on the loopback network is 127.0.0.1 for IPv4 and ::1 for IPv6. The standard domain name for the address is localhost.

The loopback interface can perform the following functions:

- Device identification—The loopback interface is used to identify the device. While any interface address can be used to determine if the device is online, the loopback address is the preferred method. Whereas interfaces might be removed or addresses changed based on network topology changes, the loopback address *never changes*.

  When you ping an individual interface address, the results do not always indicate the health of the device. For example, a subnet mismatch in the configuration of two endpoints on a point-to-point link makes the link appear to be inoperable. Pinging the interface to determine whether the devicer is online provides a misleading result. An interface might be unavailable because of a problem unrelated to the device's configuration or operation.

- Routing information—The loopback address is used by protocols such as OSPF to determine protocol-specific properties for the device or network. Further, some commands such as ping mpls require a loopback address to function correctly.

- Packet filtering—Stateless firewall filters can be applied to the loopback address to filter packets originating from, or destined for, the Routing Engine.

Management Interface

Management interfaces are the primary interfaces for accessing the device remotely. Typically, a management interface is not connected to the in-band network, but is connected instead to the device's internal network. Through a management interface you can access the device over the network using utilities such as ssh and telnet and configure it from anywhere, regardless of its physical location. Simple Network Management Protocol (SNMP) can use the management interface to gather statistics from the device.

Management interfaces vary based on device type:

- The J-series Services Routers include four built-in Gigabit Ethernet interfaces located on the front panel of the router chassis named ge-0/0/0, ge-0/0/1, ge-0/0/2, and ge-0/0/3 from left to right. These are not physically dedicated management interfaces, although the factory configuration for these routers automatically enables the J-Web user interface on these interfaces. You can use them to pass traffic or you can segregate one off and place it in the management zone to be used as a management interface. To use a built-in interface as a management Ethernet interface, configure it with a valid IP address.
- The SRX 5600 and SRX 5800 services gateways include a 10/100-Mbps Ethernet port on the Routing Engine (RE). This port, which is labeled ETHERNET, is a dedicated out-of-band management interface for the device. The JUNOS software automatically creates the device's management interface fxp0. To use fxp0 as a management port, you must configure its logical port fxp0.0 with a valid IP address. While you can use fxp0 to connect to a management network, you cannot place it into the management zone.

NOTE: On the SRX 5600 and SRX 5800 services gateways, you must first connect to the device through the serial console port before assigning a unique IP address to the management interface.

As a security feature, users cannot log in as root through a management interface. To access the device as root, you must use the console port.

Services Interfaces

On Juniper Networks M-series and T-series routing platforms, individual services such as IP-over-IP encapsulation, link services such as multilink protocols, adaptive services such as stateful firewall filters and NAT, and sampling and logging capabilities are implemented by services Physical Interface Cards (PICs). On a J-series Services Router, these same features are implemented by the general-purpose CPU on the main circuit board.

Although the same JUNOS software image supports the services features across all routing platforms, on a J-series device no Physical Interface Module (PIM) is associated with services features.

To configure services on a J-series device, you must configure one or more internal interfaces by specifying PIM slot 0 and port 0—for example, gr-0/0/0 for GRE.

J-series devices support multilink protocol services on the ls-0/0/0 interface. At the logical level, the ls-0/0/0 interface supports the Multilink Point-to-Point Protocol (MLPPP) and Multilink Frame Relay (MLFR) FRF.15 encapsulation types, and at the physical level, the interface supports the MLRF FRF.16 encapsulation type and Compressed Real-Time Transport Protocol (CRTP).

MLPPP and MLFR

Multilink Point-to-Point Protocol (MLPPP) is a protocol for aggregating multiple constituent links into one larger PPP bundle. Multilink Frame Relay (MLFR) allows you to aggregate multiple Frame Relay links by inverse multiplexing. MLPPP and MLFR provide service options between low-speed T1 and E1 services. In addition to providing additional bandwidth, bundling multiple links can add a level of fault tolerance to your dedicated access service. Because you can implement bundling across multiple interfaces, you can protect users against loss of access when a single interface fails.

MLFR Frame Relay Forum

JUNOS supports FRF.12 fragmentation header formats for both FRF.15 (MLFR) and FRF.16 (MFR).

MLFR Frame Relay Forum 15 (FRF.15) combines multiple permanent virtual circuits (PVCs) into one aggregated virtual circuit (AVC). This process provides fragmentation over multiple PVCs on one end and reassembly of the AVC on the other end. MLFR FRF.15 is supported on the ls-0/0/0 interface.

MLFR FRF.16 is supported on the ls-0/0/0:*channel*, which carries a single MLFR FRF.16 bundle. MLFR FRF.16 combines multiple links to form one logical link. Packet fragmentation and reassembly occur on each virtual circuit. Each bundle can support multiple virtual circuits.

NOTE: If you configure a permanent virtual circuit (PVC) between T1, E1, T3, or E3 interfaces in J-series device and another vendor, and the other vendor does not have the same FRF.12 support or supports FRF.12 in a different way, the devices interface might discard a fragmented packet containing FRF.12 headers and count it as a "Policed Discard." Therefore, when you configure a PVC between T1, E1, T3, or E3 interfaces in the devices and another vendor, you should configure multilink bundles on both peers and configure fragmentation thresholds on the multilink bundle.

CRTP

Real-Time Transport Protocol (RTP) can help achieve interoperability among different implementations of network audio and video applications. However, the header can be too large a payload for networks using low-speed lines such as dial-up modems. Compressed Real-Time Transport Protocol (CRTP) can reduce network overhead on a low-speed link. On a J-series device, CRTP can operate on a T1 or E1 interface with PPP encapsulation.

Chapter 4

Configuring Ethernet, DS1, DS3, and Serial Interfaces

Juniper Networks devices can use network interfaces such as DS1, DS3, Fast Ethernet, Gigabit Ethernet, and serial interfaces to transmit and receive network traffic. For network interfaces to operate, you must configure properties such as logical interfaces, the encapsulation type, and certain settings specific to the interface type.

In most cases, you can use either J-Web Quick Configuration or a configuration editor to configure network interfaces.

NOTE: You cannot configure channelized T1 or E1 interfaces through a J-Web Quick Configuration page. You must use the J-Web or CLI configuration editor. Even after configuration, channelized interfaces do not appear on the Quick Configuration Interfaces page.

This chapter includes the following topics. For more information about interfaces, see "Interfaces Overview" on page 11 and the *JUNOS Network Interfaces Configuration Guide*. To configure channelized interfaces, see "Configuring Channelized T1/E1/ISDN PRI Interfaces" on page 109. To configure DSL interfaces, see "Configuring Digital Subscriber Line Interfaces" on page 125. To configure PPPoE interfaces, see "Configuring Point-to-Point Protocol over Ethernet" on page 157. To configure ISDN interfaces, see "Configuring ISDN" on page 177.

Before You Begin

Before you configure network interfaces, you need to perform the following tasks:

- Install your Juniper Networks device. For more information, see the Hardware Guide for your device.
- Establish basic connectivity. For more information, see the Getting Started Guide for your device.

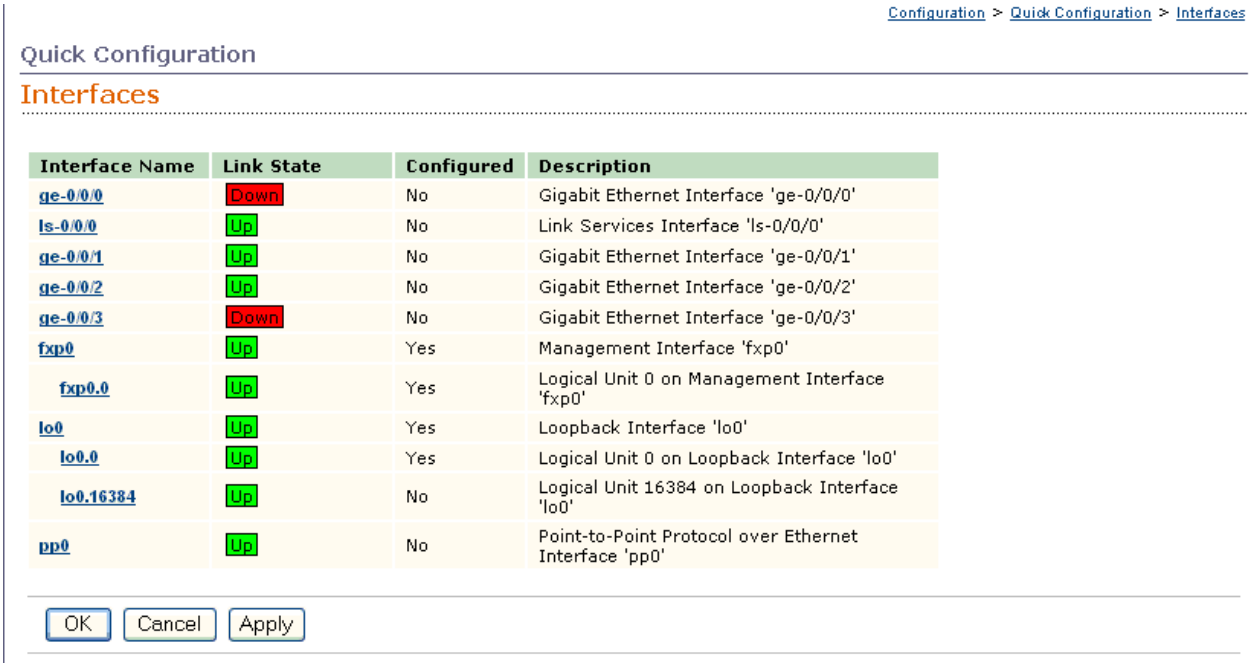
- If you do not already have a basic understanding of physical and logical interfaces and Juniper Networks interface conventions, read “Interfaces Overview” on page 11.

Although it is not a requirement, you might also want to plan how you are going to use the various network interfaces before you start configuring them. You can see a list of the physical interfaces installed on the device by displaying the Quick Configuration page, as shown in Figure 12 on page 74.

Configuring Interfaces—Quick Configuration

You can use J-Web Quick Configuration to quickly configure most network interfaces, as shown in Figure 12 on page 74.

Figure 12: Quick Configuration Interfaces Page



To configure a network interface with Quick Configuration:

1. Select **Configuration > Quick Configuration > Interfaces**. For information about interface names, see “Network Interface Naming” on page 16.

   A list of the network interfaces available on the routing platform appears, as shown in Figure 12 on page 74. The third column indicates whether the interface has been configured.

NOTE: Channelized T1 and E1 interfaces are not displayed in the list of interfaces on the J-Web Quick Configuration Interfaces page. However, you can configure and view channelized T1/E1/ISDN PRI interfaces with the J-Web configuration editor. For details, see "Configuring Channelized T1/E1/ISDN PRI Interfaces" on page 109..

2. Configure properties for a network interface by selecting the interface name and following the instructions in one of the following topics.

- Configuring an E1 Interface with Quick Configuration on page 76
- Configuring an E3 Interface with Quick Configuration on page 79
- Configuring a Fast Ethernet Interface with Quick Configuration on page 82
- Configuring Gigabit Ethernet Interfaces—Quick Configuration on page 86
- Configuring T1 Interfaces with Quick Configuration on page 89
- Configuring T3 Interfaces with Quick Configuration on page 92
- Configuring Serial Interfaces with Quick Configuration on page 95
- Configuring Redundant Ethernet Interfaces—Quick Configuration on page 99

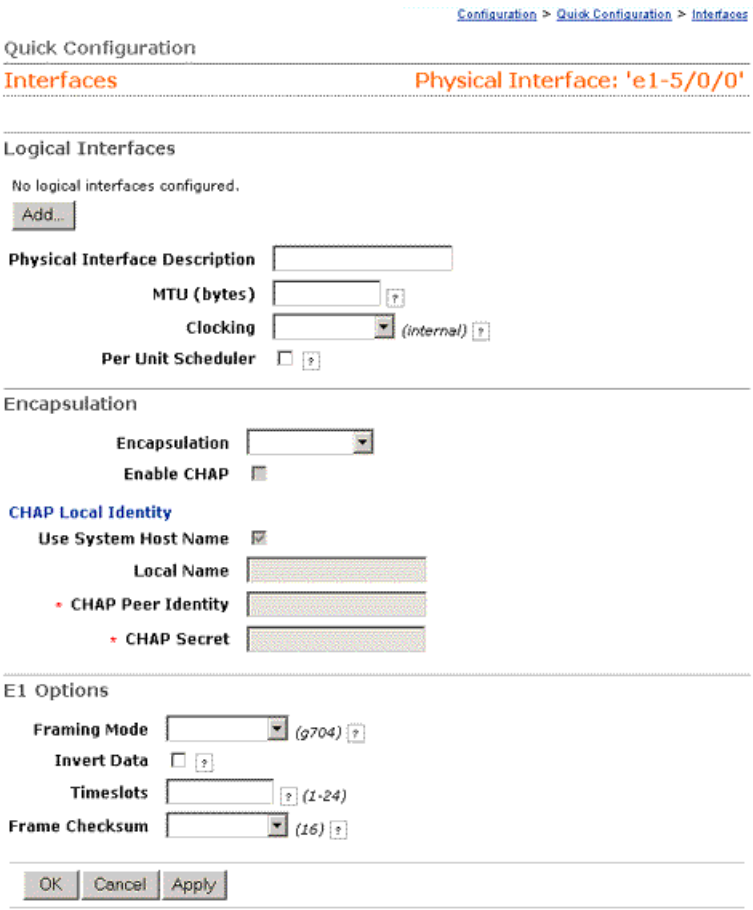
Configuring an E1 Interface with Quick Configuration

To configure properties on an E1 interface:

1. From the Quick Configuration page, as shown in Figure 12 on page 74, select the E1 interface you want to configure.

   The properties you can configure on an E1 interface are displayed, as shown in Figure 13 on page 76. (see "Network Interface Naming" on page 16.)

Figure 13: E1 Interfaces Quick Configuration Page



2. Enter information into the Quick Configuration page, as described in Table 23 on page 77.
3. Click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration page, click **Apply**.
   - To apply the configuration and return to the main configuration page, click **OK**.

- To cancel your entries and return to the main page, click **Cancel**.

4. To verify that the E1 interface is configured correctly, see "Verifying Interface Configuration" on page 106.

Table 23: E1 Quick Configuration Summary

Field	Function	Your Action
Logical Interfaces		
Add logical interfaces	Defines one or more logical units that you connect to this physical E1 interface. You must define at least one logical unit for an E1 interface. You can define multiple units if the encapsulation type is Frame Relay.	Click **Add**.
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to more clearly identify it in monitoring displays.
IPv4 Addresses and Prefixes	Specifies one or more IPv4 addresses for the interface.	1. Type one or more IPv4 addresses and prefixes. For example: 10.10.10.10/24 2. Click **Add**. 3. Click **OK**.
Physical Interface Description	(Optional) Adds supplementary information about the physical E1 interface.	Type a text description of the E1 interface to more clearly identify it in monitoring displays.
MTU (bytes)	Specifies the maximum transmission unit size for the E1 interface.	Type a value between 256 and 9192 bytes. The default MTU for E1 interfaces is 1504.
Clocking	Specifies the transmit clock source for the E1 line.	From the list, select one of the following: ■ **internal**—Device's own system clock (the default) ■ **external**—Clock received from the E1 interface
Per unit scheduler	Enables scheduling on logical interfaces. Allows you to configure multiple output queues on a logical interface and associate an output scheduler and shaping rate with the queues.	■ To enable scheduling, select the check box. ■ To disable scheduling, clear the check box.
Encapsulation		

Table 23: E1 Quick Configuration Summary *(continued)*

Field	Function	Your Action
Encapsulation	Specifies the encapsulation type for traffic on the interface.	From the list, select the encapsulation for this E1 interface: ■ **PPP** ■ **Frame Relay** ■ **Cisco HDLC**
Enable CHAP	Enables or disables CHAP authentication on an E1 interface with PPP encapsulation only.	■ To enable CHAP, select the check box. ■ To disable CHAP, clear the check box.
CHAP Local Identity (available if CHAP is enabled)		
Use System Host Name	Specifies that the E1 interface uses the device's system hostname in CHAP challenge and response packets.	■ To enable, select the check box (the default). ■ To disable, clear the check box.
Local Name	If Use System Host Name is disabled, specifies the local name for CHAP to use.	Type a local name for this E1 interface.
CHAP Peer Identity	Identifies the client or peer with which the device communicates on this E1 interface.	Type the CHAP client name.
CHAP Secret	Specifies the secret password for CHAP authentication, known to both sides of the connection.	Type a password that is known to the other side of the connection. Use a combination of letters and numbers that is difficult for others to guess.
E1 Options		
Framing Mode	Specifies the framing mode for the E1 line.	From the list, select one of the following: ■ **g704**—The default ■ **g704–no-crc4**—G704 without cyclic redundancy check 4 (CRC4) ■ **unframed**—Unframed transmission format
Invert Data	Enables or disables data inversion. Data inversion is normally used only in alternate mark inversion (AMI) mode.	■ To enable, select the check box. ■ To disable, clear the check box.
Timeslots	Specifies the number of time slots allocated to a fractional E1 interface. By default, an E1 interface uses all the time slots.	Type numeric values from **2** through **32**. Separate discontinuous entries with commas, and use hyphens to indicate ranges. For example: 2,4,7–9
Frame Checksum	Specifies the number of bits in the frame checksum. A 32–bit checksum provides more reliable packet verification, but is not supported by some older equipment.	Select **16** or **32**. The default checksum is **16**.

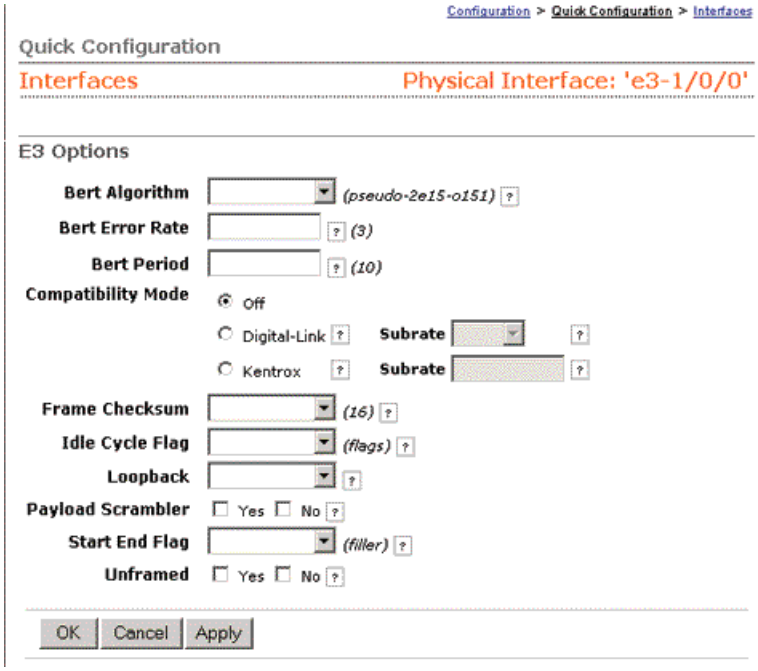
Configuring an E3 Interface with Quick Configuration

To configure properties on an E3 interface:

1. From the Quick Configuration page, as shown in Figure 12 on page 74, select the E3 interface you want to configure.

   The properties you can configure on an E3 interface are displayed, as shown in Figure 14 on page 79. (see "Network Interface Naming" on page 16.)

Figure 14: E3 Interfaces Quick Configuration Page



2. Enter information into the Quick Configuration page, as described in Table 24 on page 80.
3. Click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration page, click **Apply**.
   - To apply the configuration and return to the main configuration page, click **OK**.
   - To cancel your entries and return to the main page, click **Cancel**.
4. To verify that the E3 interface is configured correctly, see "Verifying Interface Configuration" on page 106.

Table 24: E3 Quick Configuration Summary

Field	Function	Your Action
Logical Interfaces		
Add logical interfaces	Defines one or more logical units that you connect to this physical E3 interface. You must define at least one logical unit for an E3 interface. You can define multiple units if the encapsulation type is Frame Relay.	Click **Add**.
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to more clearly identify it in monitoring displays.
IPv4 Addresses and Prefixes	Specifies one or more IPv4 addresses for the interface.	1. Type one or more IPv4 addresses and prefixes. For example: 10.10.10.10/24 2. Click **Add**. 3. Click **OK**.
Physical Interface Description	(Optional) Adds supplementary information about the physical E3 interface.	Type a text description of the E3 interface to more clearly identify it in monitoring displays.
MTU (bytes)	Specifies the maximum transmission unit size for the E3 interface.	Type a value between 256 and 9192 bytes. The default MTU for E3 interfaces is 4474.
Clocking	Specifies the transmit clock source for the E3 line.	From the list, select one of the following: ■ **internal**—Device's own system clock (the default) ■ **external**—Clock received from the E3 interface
Encapsulation		
Encapsulation	Specifies the encapsulation type for traffic on the interface.	From the list, select the encapsulation for this E3 interface: ■ **PPP** ■ **Frame Relay** ■ **Cisco HDLC**
Enable CHAP	Enables or disables CHAP authentication on an E3 interface with PPP encapsulation only.	■ To enable CHAP, select the check box. ■ To disable CHAP, clear the check box.
CHAP Local Identity (available if CHAP is enabled)		
Use System Host Name	Specifies that the E3 interface uses the device's system hostname in CHAP challenge and response packets.	■ To enable, select the check box (the default). ■ To disable, clear the check box.
Local Name	If Use System Host Name is disabled, specifies the local name for CHAP to use.	Type a local name for this E3 interface.
CHAP Peer Identity	Identifies the client or peer with which the device communicates on this E3 interface.	Type the CHAP client name.

Table 24: E3 Quick Configuration Summary *(continued)*

Field	Function	Your Action
CHAP Secret	Specifies the secret password for CHAP authentication, known to both sides of the connection.	Type a password that is known to the other side of the connection. Use a combination of letters and numbers that is difficult for others to guess.
E3 Options		
Bert Algorithm	Specifies the bit error rate test (BERT) algorithm to use during a BERT. BERT is supported only when transmission is unframed. (See the Unframed option.)	From the Bert Algorithm list, select the algorithm to use: ■ **all-ones-repeating** ■ **alternating-ones-zeros** ■ **all-zeros-repeating** ■ **pseudo-2e11–o152** ■ **pseudo-2e15–o151** ■ **pseudo-2e20–o151** ■ **pseudo-2e20–o153** ■ **pseudo-2e23–o151** ■ **pseudo-2e29** ■ **pseudo-2e31** ■ **pseudo-2e9–o153** The default is pseudo-2e15-o151.
Bert Error Rate	Specifies the exponent *n* in the bit error rate 10^{-n}.	Type a value between 3 and 7, or 0. For example, a value of 6 specifies that 1 bit out of 1,000,000 is transmitted in error. The default is 0 (no bits are transmitted in error).
Bert Period	Specifies the length of time—in seconds—of the BERT.	Type a value between 1 and 240. The default is 10.
Compatibility Mode	Defines the transmission mode and subrating to use on the E3 interface. The mode must be set to the type of channel service unit (CSU) connected to the interface. The subrating specified must be the same subrating configured on the CSU. CSU compatibility mode and subrating are supported only when transmission is unframed. (See the Unframed option.)	Select one of the following check boxes: ■ **Off**—CSU compatibility is disabled. ■ **Digital-Link**—Compatible with a Digital Link CSU. ■ **Kentrox**—Compatible with a Kentrox CSU. If you select **Digital-Link**, you can optionally specify a subrate by selecting a value from the Subrate list. If you select **Kentrox**, you can optionally specify a subrate by typing a value from 1 through 48 in the Subrate box. If you do not specify a subrate, the full E3 rate is used.
Frame Checksum	Specifies the number of bits in the frame checksum. A 32-bit checksum provides more reliable packet verification, but is not supported by some older equipment.	From the Frame Checksum list, select **16** or **32**. The default value is 16.

Table 24: E3 Quick Configuration Summary *(continued)*

Field	Function	Your Action
Idle Cycle Flag	Specifies the value to transmit during idle cycles.	From the Idle Cycle Flag list, select one of the following: ■ **flags**—Transmits the value 0x7E during idle cycles. This is the default. ■ **ones**—Transmits the value 0xFF during idle cycles.
Loopback	Configures the E3 interface as a loopback interface for testing purposes. When E3 is configured as a local loopback interface, the device transmits test traffic simultaneously to the CSU and to the receiver at the E3 interface. When E3 is configured as a remote loopback interface, test traffic transmitted by the CSU is simultaneously received at the E3 interface and transmitted back to the CSU.	From the Loopback list, select one of the following: ■ **local**—Traffic loops from the transmitter to the receiver at the E3 interface during tests. ■ **remote**—Traffic loops from the receiver to the transmitter at the E3 interface during tests.
Payload Scrambler	Specifies whether the payload of the packet is to be scrambled, or randomized, when transmitted. Scrambling eliminates nonvariable bit patterns in the transmission, which can generate link-layer errors across an E3 link. The payload scrambler is supported only when CSU compatibility is enabled and transmission is framed. (See the Compatibility Mode and Unframed options).	Select one of the following check boxes: ■ **Yes**—Transmission is scrambled. ■ **No**—Transmission is not scrambled.
Start End Flag	Specifies whether the end and start flags are separated.	From the Start End Flag list, select one of the following: ■ **filler**—Flags are separated by idle cycles. ■ **shared**—Flags overlap (no separation).
Unframed	Specifies whether the transmission is framed (G.751 framing) or unframed.	Select one of the following check boxes: ■ **Yes**—Unframed transmission. ■ **No**—Framed transmission.

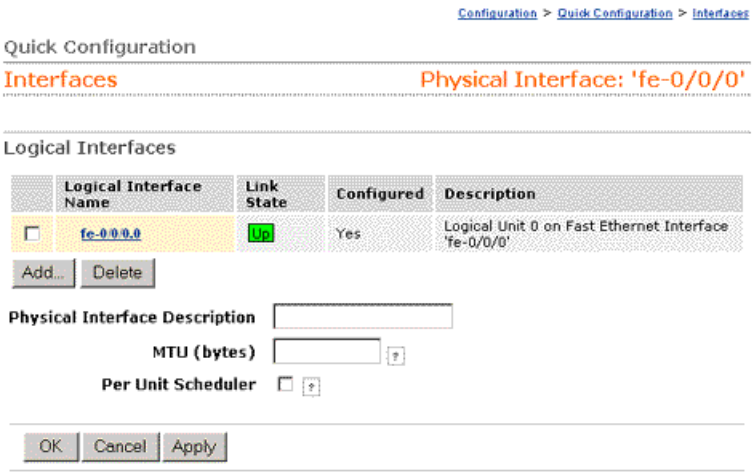
Configuring a Fast Ethernet Interface with Quick Configuration

To configure properties on a Fast Ethernet interface:

1. From the Quick Configuration page, as shown in Figure 12 on page 74, select the Fast Ethernet interface you want to configure.

   The properties you can configure on a Fast Ethernet interface are displayed, as shown in Figure 15 on page 83. (see "Network Interface Naming" on page 16.)

Figure 15: Fast Ethernet Interfaces Quick Configuration Page



2. Enter information into the Quick Configuration page, as described in Table 25 on page 83.
3. Click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration page, click **Apply**.
   - To apply the configuration and return to the main configuration page, click **OK**.
   - To cancel your entries and return to the main page, click **Cancel**.
4. To verify that the Fast Ethernet interface is configured correctly, see "Verifying Interface Configuration" on page 106.

Table 25: Fast Ethernet Quick Configuration Summary

Field	Function	Your Action
Logical Interfaces		
Add logical interfaces	Defines one or more logical units that you connect to this physical Fast Ethernet interface. You must define at least one logical unit for a Fast Ethernet interface. You can define multiple units if the encapsulation type is Frame Relay.	Click **Add**.

Table 25: Fast Ethernet Quick Configuration Summary *(continued)*

Field	Function	Your Action
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to more clearly identify it in monitoring displays.
IPv4 Addresses and Prefixes	Specifies one or more IPv4 addresses for the interface.	1. Type one or more IPv4 addresses and prefixes. For example: 10.10.10.10/24 2. Click **Add**. 3. Click **OK**.
ARP Address	Enables the device to create a static Address Resolution Protocol (ARP) entry for this interface by specifying the IP address of a node to associate with its media access control (MAC) address. The IP address must be in the same subnet as the IPv4 address or prefix of the interface you are configuring. Static ARP entries associate the IP addresses and MAC addresses of nodes on the same subnet, enabling a device to respond to ARP requests having destination addresses that are not local to the incoming interface.	Type an IPv4 address that you want to associate with the MAC address—for example, 10.10.10.1.
MAC Address	Specifies the hardware media access control (MAC) address associated with the ARP address. The MAC address uniquely identifies the system and is expressed in the following format: mm:mm:mm:ss:ss:ss. The first three octets denote the hardware manufacturer ID, and the last three are serial numbers identifying the device.	Type the MAC address to be mapped to the ARP entry—for example, 00:12:1E:A9:8A:80.
Publish	Enables the device to reply to ARP requests for the specified address. For more information, see “Configuring Static ARP Entries on Ethernet Interfaces” on page 104.	■ To enable publishing, select the check box. ■ To disable publishing, clear the check box.
Physical Interface Description	(Optional) Adds supplementary information about the physical Fast Ethernet interface.	Type a text description of the Fast Ethernet interface to more clearly identify it in monitoring displays.

Table 25: Fast Ethernet Quick Configuration Summary *(continued)*

Field	Function	Your Action
MTU (bytes)	Specifies the maximum transmission unit size for the Fast Ethernet interface.	Type a value between 256 bytes and one of the following values: ■ For built-in Fast Ethernet interfaces and Dual-Port Fast Ethernet PIM interfaces, 9192 bytes ■ For 4-Port Fast Ethernet ePIM interfaces, 1514 bytes The default MTU for Fast Ethernet interfaces is 1514.
Per unit scheduler	Enables scheduling on logical interfaces. Allows you to configure multiple output queues on a logical interface and associate an output scheduler and shaping rate with the queues.	■ To enable scheduling, select the check box. ■ To disable scheduling, clear the check box.

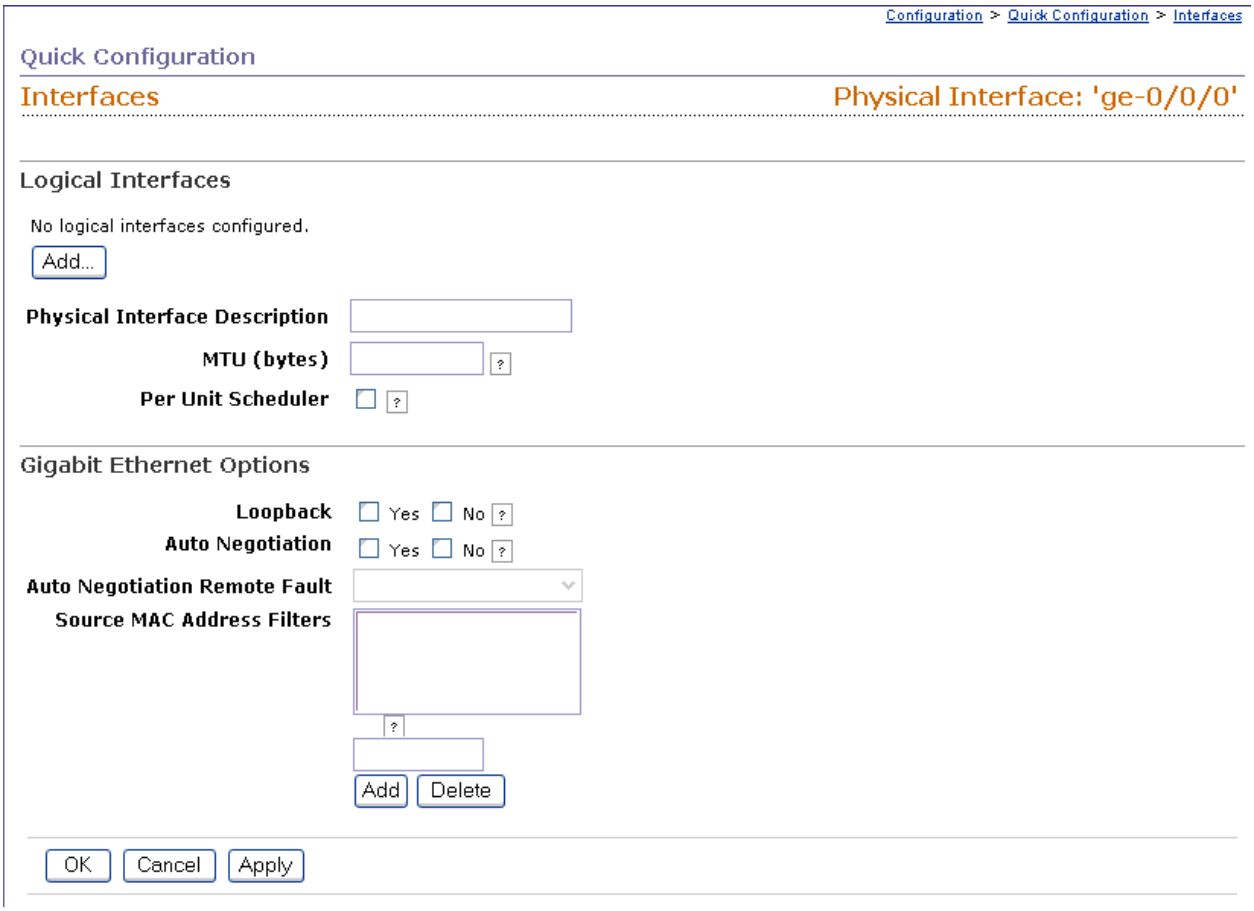
NOTE: You can also manually set the speed and link mode for a Fast Ethernet interface using the CLI commands set interfaces fe-*pim*/0/*port* speed 10m | 100m and set interfaces fe-*pim*/0/*port* link-mode half-duplex | full-duplex.

Configuring Gigabit Ethernet Interfaces—Quick Configuration

You can use J-Web Quick Configuration to quickly configure a Gigabit Ethernet interface.

1. Select **Configuration > Quick Configuration > Interfaces**. The properties you can configure on a Gigabit Ethernet interface appear as shown in Figure 16 on page 86.

Figure 16: Gigabit Ethernet Interface Quick Configuration



2. Fill in the information as described in Table 26 on page 87.
3. Click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration page, click **Apply**.
   - To apply the configuration and return to the main Configuration page, click **OK**.
   - To cancel your entries and return to the main page, click **Cancel**.
4. To verify that the Gigabit Ethernet interface is configured correctly, see "Verifying Interface Configuration" on page 106.

Table 26: Gigabit Ethernet Quick Configuration Page Summary

Field	Function	Action
Logical Interfaces		
Add logical interfaces	Defines one or more logical units that you connect to this physical Gigabit Ethernet interface. You must define at least one logical unit for a Gigabit Ethernet interface.	Click **Add**.
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to more clearly identify it in monitoring displays.
IPv4 Addresses and Prefixes	Specifies one or more IPv4 addresses for the interface.	1. Type one or more IPv4 addresses and prefixes. For example: 10.10.10.10/24 2. Click **Add**. 3. Click **OK**. To delete an IP address and prefix, select them in the Source Addresses and Prefixes box, then click **Delete**.
ARP Address	Enables the device to create a static Address Resolution Protocol (ARP) entry for this interface by specifying the IP address of a node to associate with its media access control (MAC) address. The IP address must be in the same subnet as the IPv4 address or prefix of the interface you are configuring. Static ARP entries associate the IP addresses and MAC addresses of nodes on the same subnet, enabling a device to respond to ARP requests having destination addresses that are not local to the incoming interface.	Type an IPv4 address that you want to associate with the MAC address—for example, 10.10.10.1.
MAC Address	Specifies the hardware media access control (MAC) address associated with the ARP address. The MAC address uniquely identifies the system and is expressed in the following format: mm:mm:mm:ss:ss:ss. The first three octets denote the hardware manufacturer ID, and the last three are serial numbers identifying the device.	Type the MAC address to be mapped to the ARP entry—for example, 00:12:1E:A9:8A:80.
Publish	Enables the device to reply to ARP requests for the specified address. For more information, see “Configuring Static ARP Entries on Ethernet Interfaces” on page 104.	■ To enable publishing, select the check box. ■ To disable publishing, clear the check box.
Physical Interface Description	(Optional) Adds supplementary information about the physical Gigabit Ethernet interface.	Type a text description of the Gigabit Ethernet interface to more clearly identify it in monitoring displays.

Table 26: Gigabit Ethernet Quick Configuration Page Summary *(continued)*

Field	Function	Action
MTU (bytes)	Specifies the maximum transmission unit size for the Gigabit Ethernet interface.	Type a value between 256 and 9014 bytes. The default MTU for Gigabit Ethernet interfaces is 1514.
Per unit scheduler	Enables scheduling on logical interfaces. Allows you to configure multiple output queues on a logical interface and associate an output scheduler and shaping rate with the queues.	■ To enable scheduling, select the check box. ■ To disable scheduling, clear the check box.
Gigabit Ethernet Options		
Loopback	Enables or disables the loopback option.	Select **Yes** to enable the loopback diagnostic option, or select **No** to disable the loopback option. By default, loopback is disabled.
Auto Negotiation	Enables or disables autonegotiation. By default, Gigabit Ethernet interfaces autonegotiate the link mode and speed settings. If you disable autonegotiation and do not manually configure link mode and speed, the link is negotiated at 1000 Mbps, full duplex. When you configure both the link mode and the speed, the link negotiates with the manually configured settings whether autonegotiation is enabled or disabled.	Select **Yes** to enable autonegotiation, or select **No** to disable it. By default, autonegotiation is enabled.
Auto Negotiation Remote Fault	Indicates the autonegotiation remote fault value.	Select the autonegotation remote fault value from the list of options given. This field is enabled only if autonegotiation is enabled.
Source MAC Address Filters	Displays the list of media access control (MAC) addresses from which you want to receive packets on this interface.	To add MAC addresses, type them in the boxes above the **Add** button, then click **Add**. To delete a MAC address, select it in the Source Addresses box, then click **Delete**.

NOTE: You can also manually set the speed and link mode for built-in and copper PIM Gigabit Ethernet interfaces on J4350 and J6350 devices using the CLI commands set interfaces ge-*pim*/0/*port* speed 10m | 100m | 1000m and set interfaces ge-*pim*/0/*port* link-mode half-duplex | full-duplex. (You cannot manually configure speed and link mode on SFP Gigabit Ethernet PIMs.) You must configure both link mode and speed—if you configure only one or the other, the system ignores the configuration and generates a system log message.

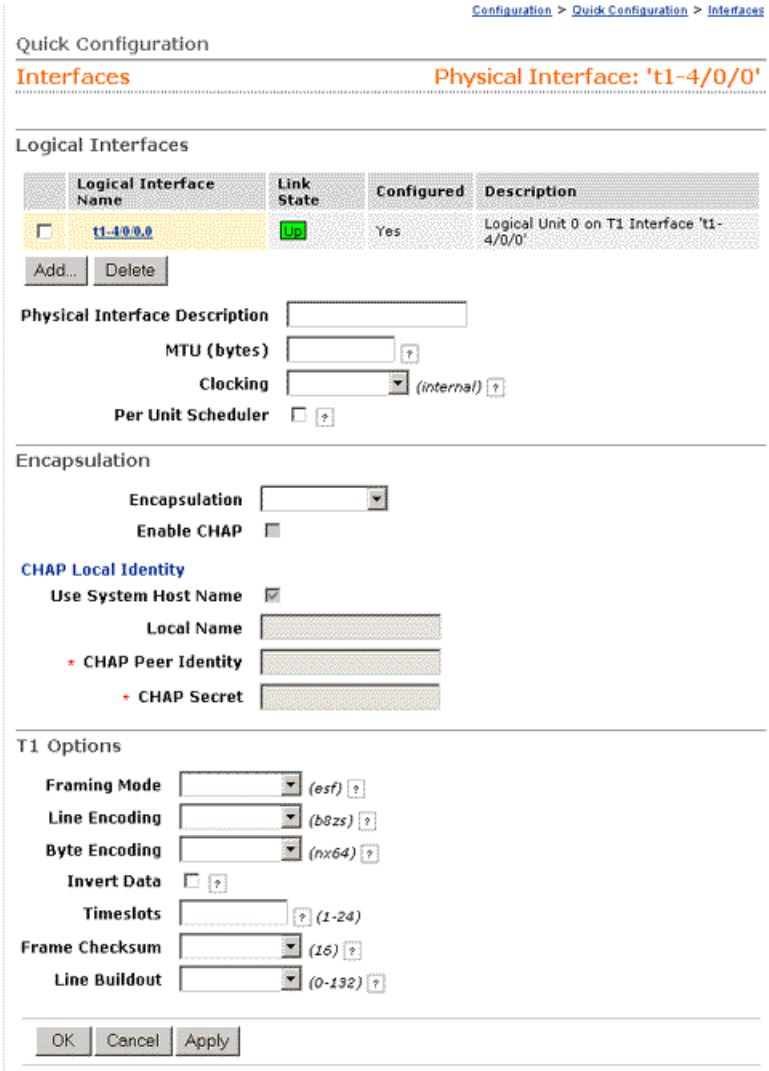
Configuring T1 Interfaces with Quick Configuration

To configure properties on a T1 interface:

1. From the Quick Configuration page, as shown in Figure 12 on page 74, select the T1 interface you want to configure.

   The properties you can configure on a T1 interface are displayed, as shown in Figure 17 on page 89. (see "Network Interface Naming" on page 16.)

Figure 17: T1 Interfaces Quick Configuration Page



2. Enter information into the Quick Configuration page, as described in Table 27 on page 90.
3. Click one of the following buttons:

- To apply the configuration and stay on the Quick Configuration page, click **Apply**.
- To apply the configuration and return to the main configuration page, click **OK**.
- To cancel your entries and return to the main page, click **Cancel**.

4. To verify that the T1 interface is configured correctly, see "Verifying Interface Configuration" on page 106.

Table 27: T1 Quick Configuration Summary

Field	Function	Your Action
Logical Interfaces		
Add logical interfaces	Defines one or more logical units that you connect to this physical T1 interface. You must define at least one logical unit for a T1 interface. You can define multiple units if the encapsulation type is Frame Relay.	Click **Add**.
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to more clearly identify it in monitoring displays.
IPv4 Addresses and Prefixes	Specifies one or more IPv4 addresses for the interface.	1. Type one or more IPv4 addresses and prefixes. For example: 10.10.10.10/24 2. Click **Add**. 3. Click **OK**.
Physical Interface Description	(Optional) Adds supplementary information about the physical T1 interface.	Type a text description of the T1 interface to more clearly identify it in monitoring displays.
MTU (bytes)	Specifies the maximum transmission unit size for the T1 interface.	Type a value between 256 and 9192 bytes. The default MTU for T1 interfaces is 1504.
Clocking	Specifies the transmit clock source for the T1 line.	From the list, select one of the following: ■ **internal**—Device's own system clock (the default) ■ **external**—Clock received from the T1 interface
Per unit scheduler	Enables scheduling on logical interfaces. Allows you to configure multiple output queues on a logical interface and associate an output scheduler and shaping rate with the queues.	■ To enable scheduling, select the check box. ■ To disable scheduling, clear the check box.
Encapsulation		

Table 27: T1 Quick Configuration Summary *(continued)*

Field	Function	Your Action
Encapsulation	Specifies the encapsulation type for traffic on the interface.	From the list, select the encapsulation for this T1 interface: ■ **PPP** ■ **Frame Relay** ■ **Cisco HDLC**
Enable CHAP	Enables or disables CHAP authentication on a T1 interface with PPP encapsulation only.	■ To enable CHAP, select the check box. ■ To disable CHAP, clear the check box.
CHAP Local Identity (available if CHAP is enabled)		
Use System Host Name	Specifies that the T1 interface uses the device's system hostname in CHAP challenge and response packets.	■ To enable, select the check box (the default). ■ To disable, clear the check box.
Local Name	If Use System Host Name is disabled, specifies the local name for CHAP to use.	Type a local name for this T1 interface.
CHAP Peer Identity	Identifies the client or peer with which the device communicates on this T1 interface.	Type the CHAP client name.
CHAP Secret	Specifies the secret password for CHAP authentication, known to both sides of the connection.	Type a password that is known to the other side of the connection. Use a combination of letters and numbers that is difficult for others to guess.
T1 Options		
Framing Mode	Specifies the framing mode for the T1 line.	From the list, select one of the following: ■ **esf**—Extended superframe (the default) ■ **sf**—Superframe
Line Encoding	Specifies the line encoding method.	From the list, select one of the following: ■ **ami**—Alternate mark inversion ■ **b8zs**—Binary 8 zero substitution (the default)
Byte Encoding	Specifies the byte encoding method.	From the list, select one of the following: ■ **nx56**—7 bits per byte ■ **nx64**—8 bits per byte (the default)
Invert Data	Enables or disables data inversion. Data inversion is normally used only in alternate mark inversion (AMI) mode.	■ To enable, select the check box. ■ To disable, clear the check box.
Timeslots	Specifies the number of time slots allocated to a fractional T1 interface. By default, a T1 interface uses all the time slots.	Type numeric values from 1 through 24. You can use any combination of time slots. To configure ranges, use hyphens. To configure discontinuous slots, use commas. For example: 1–5,10,24

Table 27: T1 Quick Configuration Summary *(continued)*

Field	Function	Your Action
Frame Checksum	Specifies the number of bits in the frame checksum. A 32–bit checksum provides more reliable packet verification, but is not supported by some older equipment.	Select **16** or **32**. The default value is **16**.
Line Buildout	Specifies the T1 line buildout in feet for cables 655 feet (200 m) or shorter, or in decibels for longer cables. Line buildout compensates for the loss in decibels based on the distance from the device to the first repeater in the circuit.	From the list, select one of the following line buildouts: ■ **0–132** (0 m–40 m) (the default) ■ **133–265** (40 m–81 m) ■ **266–398** (81 m–121 m) ■ **399–531** (121 m–162 m) ■ **532–655** (162 m–200 m) ■ **long-0db** ■ **long-7.5db** ■ **long-15db** ■ **long-22.5db**

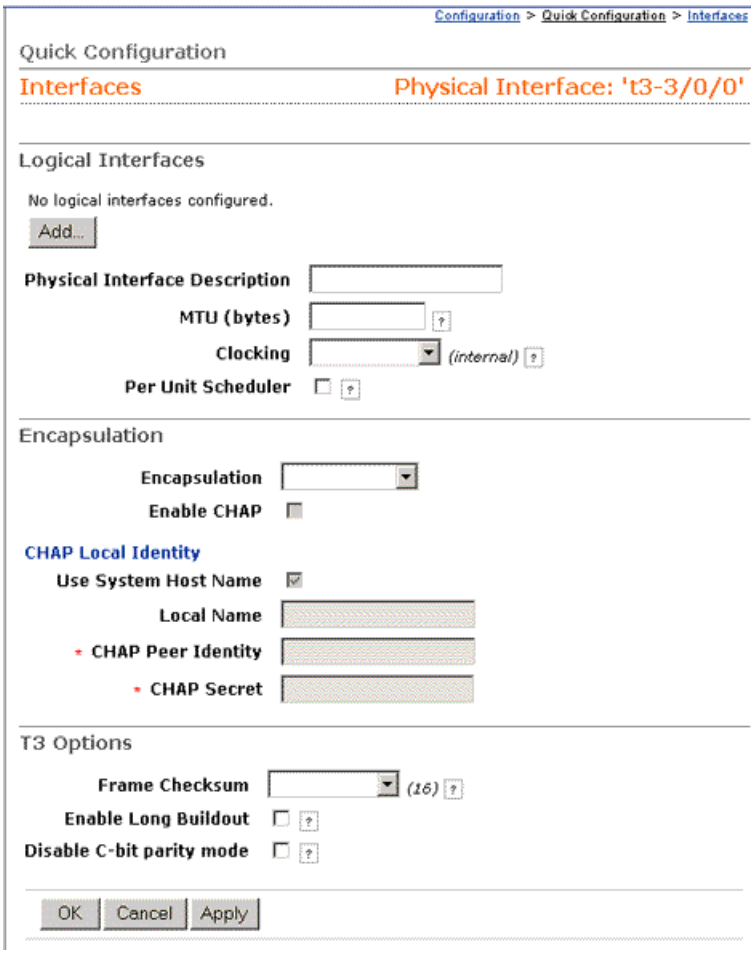
Configuring T3 Interfaces with Quick Configuration

To configure properties on a T3 (DS3) interface:

1. From the Quick Configuration page, as shown in Figure 12 on page 74, select the T3 interface you want to configure.

   The properties you can configure on a T3 interface are displayed, as shown in Figure 18 on page 93. (see “Network Interface Naming” on page 16.)

Figure 18: T3 Interfaces Quick Configuration Page



2. Enter information into the Quick Configuration page, as described in Table 28 on page 94.
3. Click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration page, click **Apply**.
   - To apply the configuration and return to the main configuration page, click **OK**.
   - To cancel your entries and return to the main page, click **Cancel**.
4. To verify that the T3 interface is configured correctly, see “Verifying Interface Configuration” on page 106.

Table 28: T3 Quick Configuration Summary

Field	Function	Your Action
Logical Interfaces		
Add logical interfaces	Defines one or more logical units that you connect to this physical T3 interface. You must define at least one logical unit for a T3 interface. You can define multiple units if the encapsulation type is Frame Relay.	Click **Add**.
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to more clearly identify it in monitoring displays.
IPv4 Addresses and Prefixes	Specifies one or more IPv4 addresses for the interface.	1. Type one or more IPv4 addresses and prefixes. For example: 10.10.10.10/24 2. Click **Add**. 3. Click **OK**.
Physical Interface Description	(Optional) Adds supplementary information about the physical T3 interface.	Type a text description of the T3 interface to more clearly identify it in monitoring displays.
MTU (bytes)	Specifies the maximum transmission unit size for the T3 interface.	Type a value between 256 and 9192 bytes. The default MTU for T3 interfaces is 4474.
Clocking	Specifies the transmit clock source for the T3 line.	From the list, select one of the following: ■ **internal**—Device's own system clock (the default) ■ **external**—Clock received from the T3 interface
Per unit scheduler	Enables scheduling on logical interfaces. Allows you to configure multiple output queues on a logical interface and associate an output scheduler and shaping rate with the queues.	■ To enable scheduling, select the check box. ■ To disable scheduling, clear the check box.
Encapsulation		
Encapsulation	Specifies the encapsulation type for traffic on the interface.	From the list, select the encapsulation for this T3 interface: ■ **PPP** ■ **Frame Relay** ■ **Cisco HDLC**
Enable CHAP	Enables or disables CHAP authentication on a T3 interface with PPP encapsulation only.	■ To enable CHAP, select the check box. ■ To disable CHAP, clear the check box.
CHAP Local Identity (available if CHAP is enabled)		

Table 28: T3 Quick Configuration Summary *(continued)*

Field	Function	Your Action
Use System Host Name	Specifies that the T3 interface uses the device's system hostname in CHAP challenge and response packets.	■ To enable, select the check box (the default). ■ To disable, clear the check box.
Local Name	If Use System Host Name is disabled, specifies the local name for CHAP to use.	Type a local name for this T3 interface.
CHAP Peer Identity	Identifies the client or peer with which the device communicates on this T3 interface.	Type the CHAP client name.
CHAP Secret	Specifies the secret password for CHAP authentication, known to both sides of the connection.	Type a password that is known to the other side of the connection. Use a combination of letters and numbers that is difficult for others to guess.
T3 Options		
Frame Checksum	Specifies the number of bits in the frame checksum. A 32-bit checksum provides more reliable packet verification, but is not supported by some older equipment.	Select **16** or **32**. The default value is 16.
Enable Long Buildout	Specifies a short or long cable length for copper-cable-based T3 interfaces. A long cable is longer than 225 feet (68.6m).	■ To enable long buildout, select the check box. ■ To disable long buildout, clear the check box.
Disable C-Bit Parity Mode	Enables or disables C-bit parity mode, which controls the type of framing that is present on the transmitted T3 signal.	■ To disable, select the check box. ■ To enable, clear the check box.

Configuring Serial Interfaces with Quick Configuration

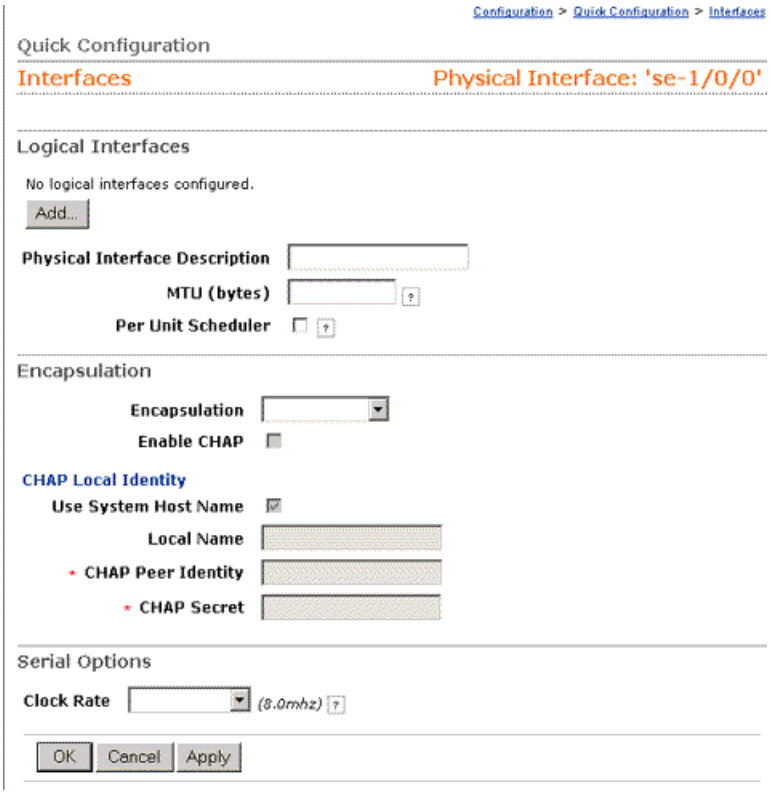
A serial interface uses a serial line protocol—such as EIA-530, X.21, RS-449/422, RS-232, or V.35—to control the transmission of signals across the interface. You do not need to explicitly configure the serial line protocol, because it is automatically detected by the Juniper Networks device based on the cable plugged into the serial interface.

To configure properties on a serial interface:

1. From the Quick Configuration page, as shown in Figure 12 on page 74, select the serial interface you want to configure.

   The properties you can configure on a serial interface are displayed, as shown in Figure 19 on page 96. (see "Network Interface Naming" on page 16.)

Figure 19: Serial Interfaces Quick Configuration Page



2. Enter information into the Quick Configuration page, as described in Table 29 on page 97.
3. Click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration page, click **Apply**.
   - To apply the configuration and return to the main configuration page, click **OK**.
   - To cancel your entries and return to the main page, click **Cancel**.
4. To verify that the serial interface is configured correctly, see "Verifying Interface Configuration" on page 106.

Table 29: Serial Quick Configuration Summary

Field	Function	Your Action
Logical Interfaces		
Add logical interfaces	Defines one or more logical units that you connect to this physical serial interface. You must define at least one logical unit for a serial interface. You can define multiple units if the encapsulation type is Frame Relay.	Click **Add**.
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to more clearly identify it in monitoring displays.
IPv4 Addresses and Prefixes	Specifies one or more IPv4 addresses for the interface.	1. Type one or more IPv4 addresses and prefixes. For example: 10.10.10.10/24 2. Click **Add**. 3. Click **OK**.
Physical Interface Description	(Optional) Adds supplementary information about the physical serial interface.	Type a text description of the serial interface to more clearly identify it in monitoring displays.
MTU (bytes)	Specifies the maximum transmission unit size for a serial interface.	Type a value between 256 and 9192 bytes. The default MTU for serial interfaces is 1504.
Per unit scheduler	Enables scheduling on logical interfaces. Allows you to configure multiple output queues on a logical interface and associate an output scheduler and shaping rate with the queues.	■ To enable scheduling, select the check box. ■ To disable scheduling, clear the check box.
Encapsulation		
Encapsulation	Specifies the encapsulation type for traffic on the interface.	From the list, select the encapsulation for this serial interface: ■ **PPP** ■ **Frame Relay** ■ **Cisco HDLC**
Enable CHAP	Enables or disables CHAP authentication on a serial interface with PPP encapsulation only.	■ To enable CHAP, select the check box. ■ To disable CHAP, clear the check box.
CHAP Local Identity (available if CHAP is enabled)		
Use System Host Name	Specifies that the serial interface use the device's system hostname in CHAP challenge and response packets.	■ To enable, select the check box (the default). ■ To disable, clear the check box.
Local Name	If Use System Host Name is disabled, specifies the local name for CHAP to use.	Type a local name for this serial interface.

Table 29: Serial Quick Configuration Summary *(continued)*

Field	Function	Your Action
CHAP Peer Identity	Identifies the client or peer with which the device communicates on this serial interface.	Type the CHAP client name.
CHAP Secret	Specifies the secret password for CHAP authentication, known to both sides of the connection.	Type a password that is known to the other side of the connection. Use a combination of letters and numbers that is difficult for others to guess.
Serial Options		
Clocking Mode	Specifies the clock source to determine the timing on serial interfaces. If you use an externally timed clocking mode—**dce** or **loop**—long cables might introduce a phase shift of DTE-transmitted clock and data. At high speeds, this phase shift might cause errors. Inverting the transmit clock corrects the phase shift, thereby reducing error rates. By default, the transmit clock is not inverted. To invert the transmit clock, do either of the following: ■ In the J-Web configuration editor, set the Transmit clock value to invert on the **Interfaces > *interface-name* > Serial options** page. ■ In the CLI configuration editor, include the transmit-clock invert statement at the [edit interfaces se-*pim*/0/*port* serial-options] hierarchy level.	From the list, select one of the following timing sources: ■ **dce**—Uses a transmit clock generated by the data circuit-terminating equipment (DCE) for the device's DTE. ■ **internal**—Uses the device's internal clock. ■ **loop**—Uses the DCE's or DTE's receive clock (the default). For X.21 serial interfaces, you must use the **loop** clocking mode. When the device is functioning as DTE, you must use the **dce** clocking mode for all interfaces except X.21 serial interfaces. When the device is functioning as DCE, we recommend using the **internal** clocking mode for all interfaces.

Table 29: Serial Quick Configuration Summary *(continued)*

Field	Function	Your Action
Clock Rate **NOTE:** RS-232 serial interfaces cannot function error-free with a clock rate greater than 200 KHz.	Specifies the line speed in kilohertz or megahertz for serial interfaces that use the DTE clocking mode.	From the list, select one of the following clock rates: ■ 1.2 KHz ■ 2.4 KHz ■ 9.6 KHz ■ 19.2 KHz ■ 38.4 KHz ■ 56.0 KHz ■ 64.0 KHz ■ 72.0 KHz ■ 125.0 KHz ■ 148.0 KHz ■ 250.0 KHz ■ 500.0 KHz ■ 800.0 KHz ■ 1.0 MHz ■ 1.3 MHz ■ 2.0 MHz ■ 4.0 MHz ■ 8.0 MHz

Configuring Redundant Ethernet Interfaces—Quick Configuration

NOTE: For SRX 210 devices, you can configure a maximum of eight redundant Ethernet interfaces.

You can use J-Web Quick Configuration to quickly configure redundant Ethernet (reth) interfaces. A redundant Ethernet interface is a pseudo interface that manages two “child” physical interfaces, one on each node of the cluster. Configuration parameters set for a redundant Ethernet interface are inherited by its child interfaces. A redundant Ethernet interface allows the chassis cluster to share one IP address across two links. When a redundancy group that the redundant Ethernet interface belongs to fails over, its redundant Ethernet interfaces fail over with it and their interfaces on the new node become active.

NOTE: Before configuring redundant Ethernet interfaces, you must specify the reth-count so that reth interfaces will show in the configuration or J-Web interfaces screen. For example, to specify that there will be five redundant Ethernet interfaces, enter:

```
{primary:node1}
user@host# set chassis cluster reth-count 5
```

Figure 20 on page 100 shows the Interface page.

Figure 20: Interface Page

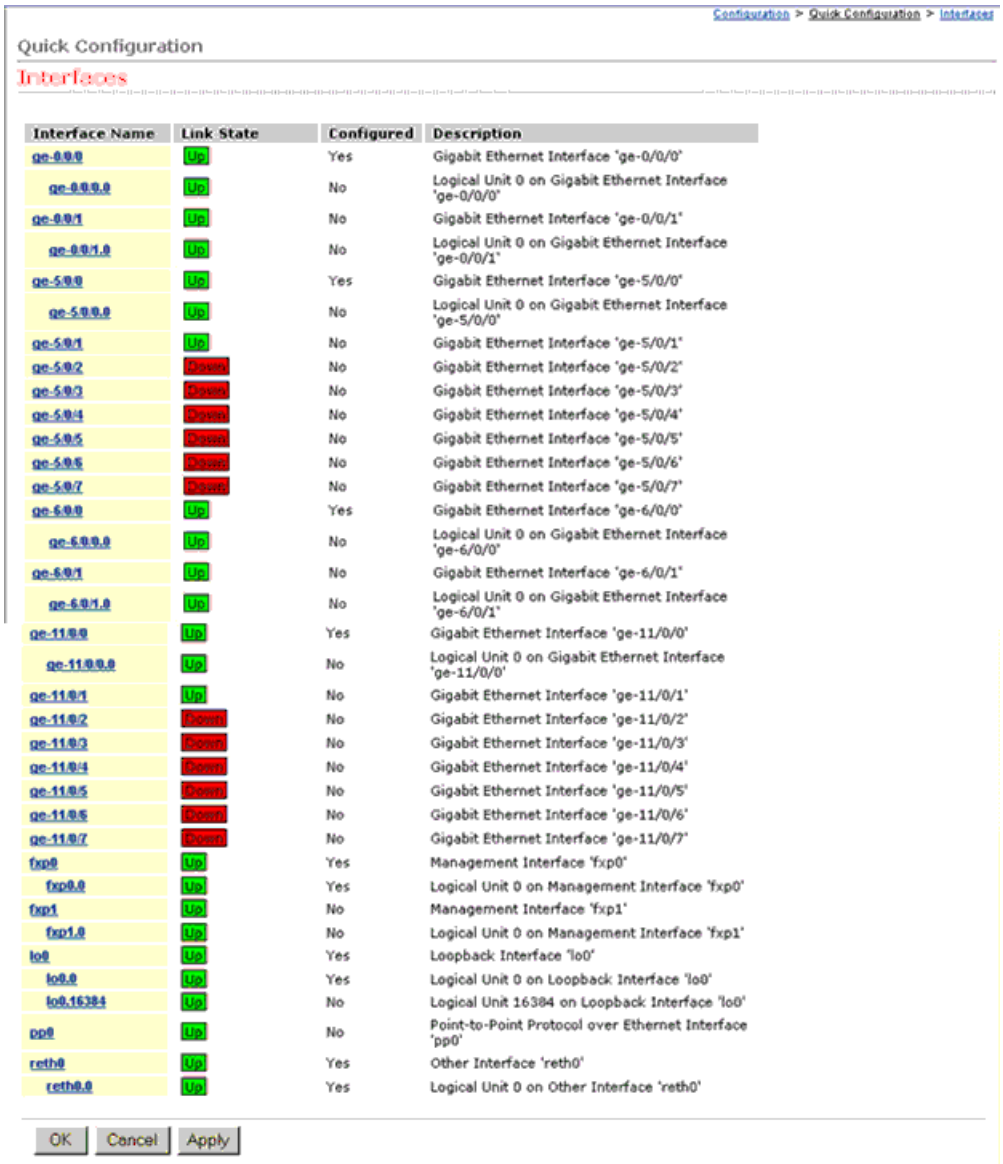
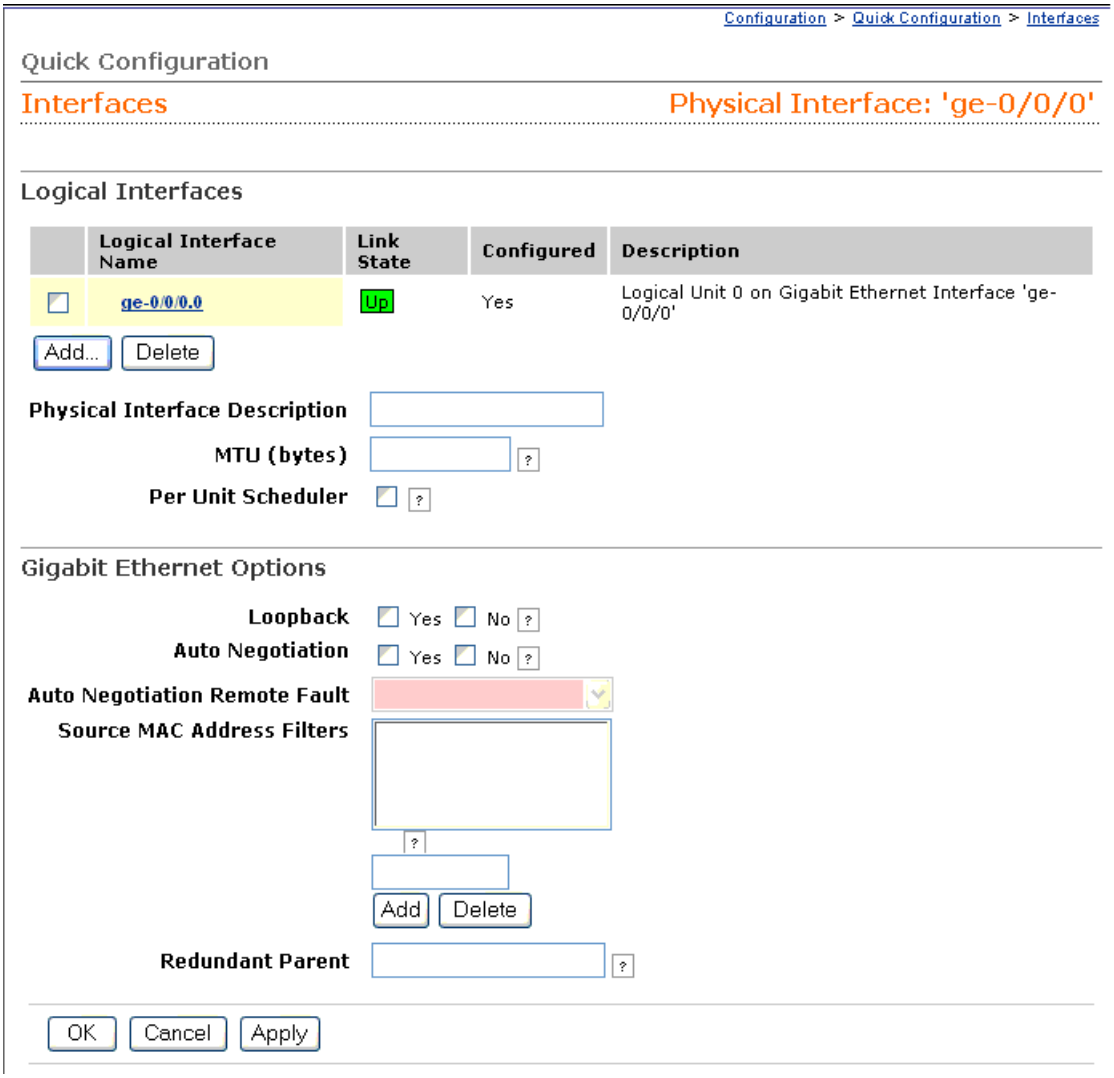


Figure 21 on page 101 shows the Redundant Ethernet Interface Configuration page.

Figure 21: Redundant Ethernet Interface Configuration page



To configure redundant Ethernet interfaces with J-Web Quick Configuration:

1. Select Configuration > Quick Configuration > Interfaces.
2. Click an interface name by which to group physical Ethernet interfaces for redundancy. See Figure 21 on page 101.
3. To add an interface to a redundant Ethernet interface, click Add.
4. Fill in the parameter settings for the logical interfaces as described in Table 30 on page 102. For details, see the *JUNOS Software Interfaces and Routing Configuration Guide*.
5. Fill in the information for Redundant Parent to specify the redundant parent Ethernet interface of the child physical interface.

- To apply the configuration and stay on the Quick Configuration page, click Apply.
- To apply the configuration and return to the main Configuration page, click OK.
- To cancel your entries and return to the main page, click Cancel.

Table 30: Redundant Ethernet Interface Options

Field	Function	Action
Logical Interfaces		
Add Logical Interfaces	Defines one or more logical units that you connect to this physical redundant Ethernet interface. You must define at least one logical unit for a redundant Ethernet interface.	Click Add. To delete a logical interface, select the check box corresponding to the interface you want to delete and click Delete.
High Availability		
Redundancy Number	Specifies the number of the redundancy group to which the redundant interface belongs. Failover properties of the interface are inherited from the redundancy group.	Select a number from 0 through 225.
Loop Back	Enables or disables the loopback option.	By default, the loopback is disabled. Select Yes to enable loopback mode.
Flow Control	Enables flow control on the Ethernet interface.	Select Yes.
Sources Filtering	Enables the filtering of media access control (MAC) source addresses to block all incoming packets to that interface.	By default, the source address filtering is disabled. Select Yes.
Redundant Parent	Specifies the name of the redundant Ethernet interface that a physical interface is associated with to form a redundant Ethernet interface pair.	Specify a redundant Ethernet interface name.

Configuring Network Interfaces with a Configuration Editor

To enable the interfaces installed on your device to work properly, you must configure their properties. You can perform basic interface configuration using the J-Web Quick Configuration pages, as described in “Configuring Interfaces—Quick Configuration” on page 74. You can perform the same configuration tasks using the J-Web or CLI configuration editor. In addition, you can configure a wider variety of options that are encountered less frequently.

You can perform the following tasks to configure interfaces:

- Adding a Network Interface with a Configuration Editor on page 103
- Configuring Static ARP Entries on Ethernet Interfaces on page 104
- Deleting a Network Interface with a Configuration Editor on page 105

For information about using the J-Web and CLI configuration editors, see the *J-Web Interface User Guide* and the *JUNOS CLI User Guide*.

Adding a Network Interface with a Configuration Editor

After you install a PIM, connect the interface cables to the ports, and power on the device, you must complete initial configuration of each network interface, as described in the following procedure:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 31 on page 103.
3. When you are finished configuring the device, commit the configuration.
4. To verify that the network interface is configured correctly, see "Verifying Interface Configuration" on page 106.

Table 31: Adding an Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces *interface-name* For information about interface names, see "Network Interface Naming" on page 16.
Create the new interface.	1. Next to Interface, click **Add new entry**. 2. Enter the name of the new interface in the Interface name box. Make sure the name conforms to the interface naming rules. For more information, see "Network Interface Naming" on page 16. 3. Click **OK**.	
Create the basic configuration for the new interface.	1. Under Interface Name in the table, click the name of the new interface. 2. Enter values in the other fields on this page if warranted. All these entries are optional, but you need to set values for Clocking and Encapsulation in particular if the default values are not suitable.	Enter values for physical interface properties as needed. Examples include changes to the default values for physical encapsulation or MTU. For example: set *interface-name* encapsulation ppp

Table 31: Adding an Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Add values for interface-specific options. Most interface types have optional parameters that are specific to the interface type.	1. Under Nested configuration, click **Configure** for the appropriate interface type. 2. In the interface-specific page that appears, enter the values you need to supply or change the default values. 3. When you are finished, click **OK** to confirm your changes or **Cancel** to cancel them and return to the previous page.	1. From the [edit interfaces *interface-name*] hierarchy level, type edit *interface*-options 2. Enter the statement for each interface-specific property for which you need to change the default value.
Add logical interfaces.	1. In the main Interface page for this interface, next to Unit, click **Add new entry**. 2. On the Unit page for logical interfaces that appears, type a number from 0 through 16384 in the Interface unit number box. 3. Enter values in other fields as required for your network. 4. To configure protocol family values if needed, under Family, click **Configure** next to the appropriate protocol. 5. To access additional subordinate hierarchies under Nested configuration, click **Configure** next to any parameter you want to configure. 6. When you are finished, click **OK**.	1. From the [edit interfaces *interface-name*] hierarchy level, type set unit*logical-unit-number* Replace *logical-unit-number* with a value from 0 through 16384. 2. Enter additional values for properties you need to configure on the logical interface, such as logical encapsulation or protocol family.

Configuring Static ARP Entries on Ethernet Interfaces

By default, the device responds to an Address Resolution Protocol (ARP) request only if the destination address of the ARP request is on the local network of the incoming interface. For Fast Ethernet or Gigabit Ethernet interfaces, you can configure static ARP entries that associate the IP addresses of nodes on the same Ethernet subnet with their media access control (MAC) addresses. These static ARP entries enable the device to respond to ARP requests even if the destination address of the ARP request is not local to the incoming Ethernet interface.

In this example, you configure a static ARP entry on Gigabit Ethernet interface ge-0/0/3 of the device consisting of the IP address and corresponding MAC address of a node on the same Ethernet subnet. The ge-0/0/3 interface has the IP address 10.1.1.1/24. The node has the IP address 10.1.1.3 and the MAC address 00:ff:85:7f:78:03. If the node on your network is another device running the JUNOS software, you can enter the show interfaces *interface-name* command to learn the IP and MAC (hardware) address of the node.

For more information about configuring static ARP entries, see the *JUNOS Network Interfaces Configuration Guide*.

To configure a static ARP entry on the ge-0/0/3 interface:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 32 on page 105.
3. If you are finished configuring the device, commit the configuration.
4. To verify the configuration, see "Verifying Interface Configuration" on page 106.

Table 32: Configuring Static ARP Entries

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces ge-0/0/3
Select the Gigabit Ethernet interface ge-0/0/3.	In the Interface name column, click ge-0/0/3.	
Configure a static ARP entry on logical unit 0 with the source address 10.1.1.1/24 on the ge-0/0/3 interface. Set the IP address of the subnet node to 10.1.1.3 and the corresponding MAC address to 00:ff:85:7f:78:03. To have the device reply to ARP requests from the node, use the publish option.	1. Under Unit, next to 0, click **Edit**. 2. Under Family, next to Inet, click **Edit**. 3. Under Address, next to 10.1.1.1/24, click **Edit**. 4. Next to Arp, click **Add new entry**. 5. In the Address box, type the IP address of the node—10.1.1.3. 6. Select the **Publish** check box. 7. From the Mac address type list, select **Mac**. 8. In the Mac box, type the MAC address 00:ff:85:7f:78:03 of node. 9. Click **OK** until you return to the Interfaces page.	1. Enter edit unit 0 2. Enter edit family inet address 10.1.1.1/24 3. Enter set arp 10.1.1.3 mac 00:ff:85:7f:78:03 publish

Deleting a Network Interface with a Configuration Editor

To delete an interface:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 33 on page 106.

NOTE: Performing this action removes the interface from the software configuration and disables it. Network interfaces remain physically present, and their identifiers continue to appear on the J-Web Monitor and Quick Configuration pages.

Table 33: Deleting an Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces
Select the interface you want to delete.	In the Interface table, under Interface name, select the name of the interface you want to delete. For information about interface names, see "Network Interface Naming" on page 16.	Enter delete *interface-name*
Execute the selection.	1. Click **Discard**. 2. In the page that appears, select the appropriate option button. If you have not made any previous changes, the only selection available is **Delete Configuration Below This Point**.	Commit the configuration change: commit

Verifying Interface Configuration

To verify an interface configuration, perform these tasks:

- Verifying the Link State of All Interfaces on page 106
- Verifying Interface Properties on page 107

Verifying the Link State of All Interfaces

Purpose By using the ping tool on each peer address in the network, verify that all interfaces on the device are operational.

Action For each interface on the device:

1. In the J-Web interface, select **Diagnose > Ping Host**.
2. In the Remote Host box, type the address of the interface for which you want to verify the link state.
3. Click **Start**. Output appears on a separate page.

Sample Output

```
PING 10.10.10.10 : 56 data bytes
64 bytes from 10.10.10.10: icmp_seq=0 ttl=255 time=0.382 ms
64 bytes from 10.10.10.10: icmp_seq=1 ttl=255 time=0.266 ms
```

Meaning If the interface is operational, it generates an ICMP response. If this response is received, the round-trip time in milliseconds is listed in the time field. For more information about the output, see the *JUNOS Software Administration Guide*.

Related Topics For more information about using the J-Web interface to ping a host, see the *JUNOS Software Administration Guide*.

For information about the ping command, see the *JUNOS Software Administration Guide* or the *JUNOS System Basics and Services Command Reference*.

Verifying Interface Properties

Purpose Verify that the interface properties are correct.

Action From the CLI, enter the show interfaces detail command.

Sample Output

```
user@host> show interfaces detail
Physical interface: ge-1/0/0, Enabled, Physical link is Up
  Interface index: 134, SNMP ifIndex: 27, Generation: 17
  Link-level type: Ethernet, MTU: 1514, Speed: 100mbps, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps 16384
  Link flags     : None
  CoS queues     : 4 supported
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:90:69:87:44:9d, Hardware address: 00:90:69:87:44:9d
  Last flapped   : 2004-08-25 15:42:30 PDT (4w5d 22:49 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                    0                    0 bps
   Output bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
   Output packets:                    0                    0 pps
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                    0                    0                    0

    1 expedited-fo                   0                    0                    0

    2 assured-forw                   0                    0                    0

    3 network-cont                   0                    0                    0

  Active alarms  : None
  Active defects : None
```

Meaning The output shows a summary of interface information. Verify the following information:

- The physical interface is Enabled. If the interface is shown as Disabled, do either of the following:
  - In the CLI configuration editor, delete the disable statement at the [edit interfaces *interface-name*] level of the configuration hierarchy.

  - In the J-Web configuration editor, clear the **Disable** check box on the **Interfaces > *interface-name*** page.

- The physical link is Up. A link state of Down indicates a problem with the interface module, interface port, or physical connection (link-layer errors).
- The Last Flapped time is an expected value. The Last Flapped time indicates the last time the physical interface became unavailable and then available again. Unexpected flapping indicates likely link-layer errors.
- The traffic statistics reflect expected input and output rates. Verify that the number of inbound and outbound bytes and packets matches expected throughput for the physical interface. To clear the statistics and see only new changes, use the clear interfaces statistics *interface-name* command.

Related Topics For a complete description of show interfaces detail output, see the *JUNOS Interfaces Command Reference*.

Chapter 5

Configuring Channelized T1/E1/ISDN PRI Interfaces

The J-series device supports the software-configurable interfaces on the Dual-Port Channelized T1/E1/ISDN PRI PIM. Each interface can be partitioned into T1 or E1 DS0 channels, or into a combination of T1 or E1 and ISDN Primary Rate Interface (PRI) B-channels and a D-channel.

NOTE: You cannot configure channelized T1/E1/ISDN/PRI interfaces through a J-Web Quick Configuration page. You must use the J-Web or CLI configuration editor. Even after configuration, channelized interfaces do not appear on the Quick Configuration Interfaces page.

This chapter includes the following topics. For more information about interfaces, see "Interfaces Overview" on page 11 and the *JUNOS Network Interfaces Configuration Guide*. For ISDN information, see "Configuring ISDN" on page 177.

Channelized T1/E1/ISDN PRI Terms

Before configuring channelized T1/E1/ISDN PRI interfaces on a J-series device, become familiar with the terms defined in Table 34 on page 109.

Table 34: Channelized T1/E1/ISDN PRI Terms

Term	Definition
channel group	Combination of DS0 or ISDN PRI B-channels interfaces partitioned from a channelized interface into a single logical bundle.

Table 34: Channelized T1/E1/ISDN PRI Terms *(continued)*

Term	Definition
channelized E1	2.048-Mbps interface that can be configured as a single clear-channel E1 interface or channelized into as many as 31 discrete DS0 interfaces, or up to 30 ISDN PRI B-channels and 1 D-channel. On J-series channelized T1/E1/ISDN PRI interfaces, time slots are numbered from 1 through 31, and time slot 1 is reserved for framing. When the interface is configured for ISDN PRI service, time slot 16 is reserved for the D-channel.
channelized interface	Interface that is a subdivision of a larger interface, minimizing the number of Physical Interface Modules (PIMs) that an installation requires. On a channelized PIM, each port can be configured as a single T1 or E1 clear channel or partitioned into multiple discrete DS0 interfaces or ISDN PRI channels.
channelized T1	1.544-Mbps interface that can be configured as a single clear-channel T1 interface or channelized into as many as 24 discrete DS0 interfaces, or up to 23 ISDN PRI B-channels and 1 D-channel. When the interface is configured for ISDN PRI service, time slot 24 is reserved for the D-channel.
E1 interface	Physical WAN interface for transmitting signals in European digital transmission (E1) format. The E1 signal format transmits information at a rate of 2.048 Mbps and can carry 32 channels of 64 Kbps each.
Primary Rate Interface (PRI)	ISDN service intended for higher-bandwidth applications than ISDN BRI. ISDN PRI consists of a single D-channel for control and signaling, plus a number of 64-Kbps B-channels—either 23 B-channels on a T1 line or 30 B-channels on an E1 line—to carry network traffic.
T1 interface	Physical WAN interface for transmitting digital signals in the T-carrier system used in the United States, Japan, and Canada. The T1 signal format carries 24 pulse code modulation (PCM) signals using time-division multiplexing (TDM) at an overall rate of 1.544 Mbps.

Channelized T1/E1/ISDN PRI Overview

You can configure a channelized T1/E1/ISDN PRI interface for T1 or E1 or ISDN PRI service.

On a channelized T1/E1/ISDN PRI PIM configured for channelized operation, you can use the "drop-and-insert" feature to integrate voice and data on a single T1 or E1 link, and save the cost of two lines.

This overview contains the following topics:

Channelized T1/E1/ISDN PRI Interfaces

Each port on a channelized T1/E1/ISDN PRI PIM is software configurable for T1, E1, or ISDN PRI service. Each channelized T1 or E1 interface can be configured as a single clear channel, or for fractional (*N*xDS0) or channelized operation, where *N* is channels 1 to 31 for an E1 interface and channels 1 to 24 for a T1 interface.

Each channelized interface can be configured as ISDN PRI B-channels and one D-channel or as a combination of T1 or E1 DS0 channels and ISDN PRI channels.

J-series ISDN PRI interfaces support the following switch types:

- ATT5E—AT&T 5ESS
- ETSI—NET3 for the United Kingdom and Europe
- NI2—National ISDN-2
- NTDMS100—Northern Telecom DMS-100
- NTT—NTT Group switch for Japan

For more information, see "ISDN PRI Transmission on Channelized Interfaces" on page 111.

Channelized T1/E1/ISDN PRI interfaces are configured through a configuration editor only.

A channelized T1/E1/ISDN PRI interface supports CoS configuration. For information about CoS features, see "Class-of-Service Overview" on page 553 and "Configuring Class of Service" on page 579.

Drop and Insert

On channelized T1/E1 interfaces configured for channelized operation, you can insert channels (time slots) from one port (for example, channels carrying voice) directly into the other port on the PIM, to replace channels coming through the Routing Engine. This feature, known as drop and insert, allows you to integrate voice and data on a single T1 or E1 link by removing the DS0 time slots of one T1 or E1 port and replacing them by inserting the time slots of another T1 or E1 port. You need not use the same time slots on both interfaces, but the time slots count must be the same.

The channels that are not configured for the drop-and-insert feature are used for normal traffic.

ISDN PRI Transmission on Channelized Interfaces

The Dual-Port Channelized T1/E1/ISDN PRI PIM provides support for ISDN PRI services such as dial-in at the central office, callback from the central office, and primary or backup network connections from branch offices. For more information about the services, see "Configuring ISDN" on page 177.

You can configure up to 23 time slots in a channelized T1 PRI interface and up to 30 time slots in a channelized E1 PRI interface as B-channels. The 24th time slot in a T1 interface and the 16th time slot in an E1 interface are configured as the D-channel interface for signaling purposes. Each B-channel supports 64 Kbps of traffic. The unconfigured time slots can be used as regular DS0 interfaces on top of the T1 or E1 physical layer.

You can install channelized T1/EI/ISDN PRI PIMs and ISDN BRI PIMs and configure both ISDN PRI and ISDN BRI service on the same J-series device.

Before You Begin

Before you configure network interfaces, you need to perform the following tasks:

- Install J-series device hardware. For more information, see the *J-series Services Routers Hardware Guide*.
- Establish basic connectivity. For more information, see the Getting Started Guide for your device.
- If you do not already have a basic understanding of physical and logical interfaces and Juniper Networks interface conventions, read "Interfaces Overview" on page 11.

Although it is not a requirement, you might also want to plan how you are going to use the various network interfaces before you start configuring them.

Configuring Channelized T1/E1/ISDN PRI interfaces with a Configuration Editor

Each port on a channelized T1/E1/ISDN PRI PIM is software configurable as a T1 or E1 clear channel. You can partition each port into up to 24 DS0 channels on a T1 interface or up to 31 DS0 channels on an E1 interface, and can insert channels from one port into another with the drop-and-insert feature.

Channelized T1/E1/ISDN PRI ports can also be partitioned into channels for ISDN PRI service.

Channelized T1/E1/ISDN PRI interfaces are configured through a configuration editor only.

This section includes the following topics:

Configuring Channelized T1/E1/ISDN PRI Interface as a Clear Channel

To configure or edit a channelized T1/E1/ISDN PRI interface as a clear channel:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 35 on page 113.
3. If you are finished configuring the J-series device, commit the configuration.
4. To verify the configuration, see "Verifying Interface Configuration" on page 106.

Table 35: Configuring a Channelized T1/E1/ISDN PRI interface as a Clear Channel

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy. For information about interface names, see "Network Interface Naming" on page 16.	1. In the J-Web interface, select **Configuration > Edit Configuration > View and Edit**. 2. Next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter one of the following: edit interfaces ct1-3/0/0 edit interfaces ce1-3/0/0
Create the new interface—for example, ct1-3/0/0 or ce1-3/0/0.	1. Next to Interfaces, click **Add new entry**. 2. In the Interface name box, type one of the following interface names: ■ ct1-3/0/0 ■ ce1-3/0/0 3. Click **OK**.	
Configure interface options: ■ Specify a transmit clock source—for example, internal. Internal clocking uses the device's own system clock (the default). External clocking uses a signal received from the T1 or E1 interface. ■ Describe the physical interface. ■ To delay the advertisement of interface transitions from up to down or down to up, set the link hold down time or link hold up time, or both. Set a value in milliseconds from 0 (the default) through 65534—for example, 500. ■ To use the channelizable interface as a single clear channel, specify no partition. ■ To use subunit queuing on Frame Relay or virtual LAN (VLAN) IQ interfaces, enable the per-unit scheduler.	1. In the Interface table, under Interface name, click the interface you are configuring:. ■ **ct1-3/0/0** ■ **ce1-3/0/0** 2. From the Clocking list, select **internal**. 3. In the Description box, type one of the following descriptions: ■ clear t1 interface ■ clear e1 interface 4. Under Hold time: Next to Down, type 500. Next to Up, type 500. 5. Under No partition, from the Interface type list, select the type of interface: ■ **t1** ■ **e1** 6. From the Scheduler type list, select **Per unit scheduler**.	1. Enter set clocking internal 2. Add a description: ■ For T1 interfaces, enter set description clear t1 interface. ■ For E1 interfaces, enter set description clear e1 interface. 3. Enter set hold-time down 500 up 500 4. Specify a clear channel: ■ For T1 interfaces, enter set no–partition interface-type t1. ■ For E1 interfaces, enter set no–partition interface-type e1. 5. Enter set per-unit-scheduler

Table 35: Configuring a Channelized T1/E1/ISDN PRI interface as a Clear Channel *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor

Configure T1 or E1 options:
■ Bit error rate test (BERT) algorithm—for example, all ones repeating.
■ BERT error rate, a value from 0 through 7—for example, 5.
■ BERT period, in seconds, a value from 1 through 240—for example, 5.
■ (T1 interfaces only) Line buildout, in feet for cables 655 ft (200 m) or shorter—for example, 0-132—or in decibels for longer cables.
■ Framing mode:
  ■ For T1 interfaces, either superframe or extended superframe (ESF)—for example, ESF
  ■ For E1 interfaces, G704, G704 without cyclic redundancy check 4 (CRC4), or G703 unframed—for example, G704.
■ (T1 interfaces only) Line encoding method—for example, alternate mark inversion (AMI).
■ Loopback mode—for example, local.
	1. Next to T1 options or E1 options, click **Configure**.
	2. From the Bert algorithm list, select **all-ones-repeating**.
	3. In the Bert error rate box, type 5.
	4. In the Bert period box, type 5.
	5. For T1 interfaces only, from the Buildout list, select **0-132**.
	6. From the Framing list:
	  ■ For T1 interfaces, select **esf**.
	  ■ For E1 interfaces, select **g704**.
	7. For T1 interfaces only, from the Line encoding list, select **ami**
	8. From the Loopback list, select **local**.
	9. Click **OK**
		1. Enter
		set bert-algorithm all-ones-repeating
		2. Enter
		set bert-error-rate 5
		3. Enter
		set bert-period 5
		4. For T1 interfaces only, enter
		set buildout 0-132
		5. Set the framing mode:
		  ■ For T1 interfaces, enter set framing esf.
		  ■ For E1 interfaces, enter set framing g704.
		6. For T1 interfaces only, enter
		set line encoding ami
		7. Enter
		set loopback local

Configure trace options.	1. Next to Traceoptions, select the check box and click **Configure**. 2. Next to Flag, click **Add new entry**. 3. From the Flag name list, select **all**. 4. Click **OK** until you return to the Interface page.	Enter set traceoptions flag all
Configure advanced options. For example, apply configuration settings from one or more groups except the test group.	1. Next to Advanced, click the expand (+) icon. 2. Next to Apply groups except, click **Add new entry**. 3. In the Value box, type test. 4. Click **OK**.	Enter set interfaces apply-groups test

Configuring Channelized T1/E1/ISDN PRI Interface to Drop and Insert Time Slots

On channelized T1/E1/ISDN PRI interfaces configured for channelized operation, you can insert channels (time slots) from one port (for example, channels carrying voice) directly into the other port on the PIM, to replace channels coming through the Routing Engine. Although you need not use the same time slots on both interfaces, the time slots count must be the same. The channels that are not configured for the drop-and-insert feature are used for normal traffic.

You must ensure that the signaling channels (port 16 for an E1 interface and port 24 for a T1 interface) are also part of the channels that are being switched through the drop-and-insert functionality. The JUNOS software does not support switching of voice and data between ports by default.

Both ports involved in the drop-and-insert configuration must use the same clock source—either the device's internal clock or an external clock. The following clock source settings are valid:

- When port 0 is set to use the internal clock, port 1 must also be set to use it, and vice versa.
- When port 0 is set to use its external clock, port 1 must be set to run on the same clock—the external clock for port 0.
- When port 1 is set to use its external clock, port 0 must be set to run on the same clock—the external clock for port 1.

For more details about valid clock combinations, see “Frequently Asked Questions About Channelized T1/E1/ISDN PRI Interfaces” on page 122.

To configure or edit the drop-and-insert feature on a channelized T1/E1/ISDN PRI interface:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 36 on page 116.
3. If you are finished configuring the device, commit the configuration.
4. To verify the configuration, see “Verifying Interface Configuration” on page 106.

Table 36: Configuring a Channelized T1/E1/ISDN PRI Interface to Drop and Insert Time Slots

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy. For information about interface names, see "Network Interface Naming" on page 16.	1. In the J-Web interface, select **Configuration > Edit Configuration > View and Edit**. 2. Next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces ct1-3/0/0
Create a new interface—for example, ct1-3/0/0.	1. Next to Interfaces, click **Add new entry**. 2. In the Interface name box, type ct1–3/0/0. 3. Click **OK**.	
Configure the clock source and partition on ct1-3/0/0. **NOTE:** While configuring the drop-and-insert feature, you must ensure that both ports on the channelized T1/E1 PIM run on the same clock. For more details about valid clock combinations, see "Frequently Asked Questions About Channelized T1/E1/ISDN PRI Interfaces" on page 122.	1. In the Interface name column, click **ct1–3/0/0**. 2. On the Interfaces page, next to Clocking, select the check box and click **Configure**. 3. From the Clocking choices list, select **external**. 4. Click **OK**. 5. On the Interfaces page, next to Partition, click **Add new entry**. 6. On the Interface Partition page, type 1 in the Partition number box. 7. From the Interface type list, Select **ds**. 8. In the Timeslots box, type 1-10. 9. Click **OK** twice.	From the [edit] hierarchy level, enter set interfaces ct1-3/0/0 clocking external set interfaces ct1-3/0/0 partition 1 timeslots 1-10 set interfaces ct1-3/0/0 partition 1 interface-type ds
Create a new interface—for example, ct1-3/0/1.	1. On the Interfaces Configuration page, next to Interface, click **Add new entry**. 2. In the Interface name box, type ct1-3/0/1. 3. Click **OK**.	From the [edit] hierarchy level, enter edit interfaces ct1-3/0/1

Table 36: Configuring a Channelized T1/E1/ISDN PRI Interface to Drop and Insert Time Slots *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure the clock source and partition on ct1-3/0/1. **NOTE:** While configuring the drop-and-insert feature, you must ensure that both ports on the channelized T1/E1 PIM run on the same clock. For more details about valid clock combinations, see "Frequently Asked Questions About Channelized T1/E1/ISDN PRI Interfaces" on page 122.	1. On the Interfaces Configuration page, click **ct1-3/0/1** in the Interface name column. 2. Next to Clocking, select the **Yes** check box, and click **Configure**. 3. From the Clocking choices list, select **external**. 4. Next to External, click **Configure**. 5. In the Interface box, type ct1-3/0/0. 6. Click **OK** twice. 7. On the Interfaces page, next to Partition, click **Add new entry**. 8. On the Interface Partition page, type 1 in the Partition number box. 9. From the Interface type list, Select **ds**. 10. In the Timeslots box, type 1-10. 11. Click **OK** twice	From the [edit] hierarchy level, enter set interfaces ct1-3/0/1 clocking external interface ct1-3/0/0 set interfaces ct1-3/0/1 partition 1 timeslots 1-10 set interfaces ct1-3/0/1 partition 1 interface-type ds
Create new interfaces—for example, ds-3/0/0:1, ds-3/0/1:1 and configure drop-and-insert feature. **NOTE:** Both interfaces configured for the drop-and-insert feature must exist on the same PIM. For example, you can configure ds-3/0/0:1 as the data input interface for ds-3/0/1:1, but not for ds-4/0/0:1.	1. On the Interfaces Configuration page, next to Interface, click **Add new entry**. 2. In the Interface name box, type ds–3/0/0:1. 3. Click **OK**. 4. On the Interfaces Configuration page, click **ds–3/0/0:1** in the Interface name column. 5. Next to Data input, click **Configure**. 6. From the Input choice list, select **interface**. 7. In the Interface box, type ds-3/0/1:1. 8. Click **OK**.	From the [edit] hierarchy level, enter edit interfaces ds-3/0/0:1 Enter set interfaces ds-3/0/0:1 data-input interface ds-3/0/1:1

Configuring Channelized T1/E1/ISDN PRI Interfaces for ISDN PRI Operation

On a J-series device with Dual-Port Channelized T1/E1/ISDN PRI PIMs, you can configure each port for either T1, E1, or ISDN PRI service, or for a combination of ISDN PRI and either channelized T1 or E1 service. For a channelized T1 interface with ISDN PRI service, you can configure 23 B-channels and for a channelized E1 interface with ISDN PRI service, you can configure 30 B-channels.

You must also explicitly configure a D-channel: time slot 24 on a channelized T1 interface and time slot 16 on a channelized E1 interface. In addition, you select a switch type and trace options.

Setting up the J-series device for ISDN PRI operation is a multipart process. First, you add ISDN PRI service on a channelized interface as shown here. Second, you follow the instructions in "Configuring Dialer Interfaces (Required)" on page 192 to configure a dialer interface. You can then configure ISDN services such as dial-in, callback, and backup. For details, see "Configuring ISDN" on page 177.

To configure an ISDN PRI network service on a channelized T1 or E1 interface for the J-series device:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 37 on page 118.
3. Go on to "Configuring Dialer Interfaces (Required)" on page 192.

Table 37: Adding an ISDN PRI Service to a Channelized T1/E1/ISDN PRI Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces ct1-2/0/0
Create a new interface—for example, ct1-2/0/0. For information about interface names, see "Network Interface Naming" on page 16.	1. Next to Interfaces, click **Add new entry**. 2. In the Interface name box, type ct1–2/0/0. 3. Click **OK**.	

Table 37: Adding an ISDN PRI Service to a Channelized T1/E1/ISDN PRI Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure the partition and interface type. For example, partition the interface into time slots 1 through 23 for B-channels and time slot 24 for the D-channel. For a channelized T1 interface, you can configure 1 through 23 as B-channels and the 24th channel as the signaling channel (D-channel). For a channelized E1 interface, you can configure 1 through 15 and 17 through 31 as B-channels and the 16th channel as the signaling channel (D-channel).	1. In the Interface name column, click **ct1–2/0/0**. 2. On the Interfaces page, next to Partition, click **Add new entry**. 3. In the Partition number box, type **1–23**. 4. In the Timeslots box, type **1-23**. 5. From the Interface type list, Select **bc**. 6. Click **OK**. 7. On the Interfaces page, next to Partition, click **Add new entry**. 8. In the Partition number box, type **24**. 9. In the Timeslots box, type **24**. 10. From the Interface type list, Select **dc**. 11. Click **OK**.	From the [edit] hierarchy level, enter set interfaces ct1-2/0/0 partition 1-23 timeslots 1-23 set interfaces ct1-2/0/0 partition 1-23 interface-type bc set interfaces ct1-2/0/0 partition 24 timeslots 24 set interfaces ct1-2/0/0 partition 24 interface-type dc
Configure a trace options flag.	1. Next to Traceoptions, select the check box and click **Configure**. 2. Next to Flag, click **Add new entry**. 3. From the Flag name list, select **q921**. 4. Click **OK** until you return to the Interface page.	From the [edit] hierarchy level, enter set interfaces ct1-2/0/0 traceoptions flag q921
Configure B-channel allocation order for allocating a free B-channel for dial-out calls. You can allocate from the lowest-numbered or highest-numbered time slot. The default value is descending.	1. On the Interfaces page, next to Isdn options, click **Configure**. 2. From the Bchannel allocation list, select **ascending**. 3. Click **OK**.	To set the ISDN options, from the [edit] hierarchy level, enter set interfaces ct1-2/0/0 isdn-options bchannel-allocation ascending
Select the type of ISDN switch—for example, NI2. The following switches are compatible with J-series devices: ■ ATT5E—AT&T 5ESS ■ ETSI—NET3 for the UK and Europe ■ NI2—National ISDN-2 ■ NTDMS-100—Northern Telecom DMS-100 ■ NTT—NTT Group switch for Japan	From the Switch type list, select **ni2**.	From the [edit] hierarchy level, enter set interfaces ct1-2/0/0 isdn-options switch-type ni2

Table 37: Adding an ISDN PRI Service to a Channelized T1/E1/ISDN PRI Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure Q.931 timers. Q.931 is a Layer 3 protocol for the setup and termination of connections. The default value for each timer is 10 seconds, but can be configured between 1 and 65536 seconds—for example, 15.	1. In the T310 box, type 15. 2. Click **OK**.	From the [edit] hierarchy level, enter set isdn-options t310 15
Configure dialer options. ■ Name the dialer pool—for example, ISDN-dialer-group. ■ Set the dialer pool priority—for example, 1. Dialer pool priority has a range from 1 to 255, with 1 designating lowest-priority interfaces and 255 designating the highest-priority interfaces.	1. On the Interfaces page, next to Dialer options, select **Yes** and then click **configure**. 2. Next to Pool, click **Add new entry**. 3. In the Pool identifier box, type isdn-dialer-group. 4. In the Priority box, type 1. 5. Click **OK**.	From the [edit interfaces ct1-2/0/0] hierarchy level, enter set dialer-options pool isdn-dialer-group priority 1

To configure a dialer interface, see "Configuring Dialer Interfaces (Required)" on page 192.

Verifying Channelized T1/E1/ISDN PRI Interfaces

To verify an interface configuration, perform these tasks:

- Verifying Channelized Interfaces on page 120
- Verifying Clear-Channel Interfaces on page 121
- Verifying ISDN PRI Configuration on Channelized T1/E1/ISDN PRI Interfaces on page 122

Verifying Channelized Interfaces

Purpose Verify that your configurations for the channelized interfaces are correct.

Action From the CLI, enter the show interfaces ct1-3/0/1 command.

Sample Output

```
user@host> show interfaces ct1-3/0/1

Physical interface: ct1-3/0/1, Enabled, Physical link is Up
  Interface index: 151, SNMP ifIndex: 28
  Link-level type: Controller, Clocking: Internal, Speed: E1, Loopback: None,
  Framing: G704, Parent: None
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Last flapped   : 2006-10-05 21:11:48 PDT (06:45:04 ago)
  DS1   alarms   : None
  DS1   defects  : None
    Line encoding: HDB3
```

Meaning The output shows a summary of information about the physical parent interface—a channelized T1 interface in this example.

Related Topics For a complete description of show interfaces output, see the *JUNOS Interfaces Command Reference*.

Verifying Clear-Channel Interfaces

Purpose Verify that your configurations for the clear-channel interfaces are correct.

Action From the CLI, enter the show interfaces e1-3/0/1 command.

Sample Output

```
user@host> show interfaces e1-3/0/1

Physical interface: e1-3/0/1, Enabled, Physical link is Up
  Interface index: 212, SNMP ifIndex: 237
  Link-level type: PPP, MTU: 1504, Speed: E1, Loopback: None, FCS: 16,
  Parent: ce1-3/0/1 Interface index 151
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps Internal: 0x4000
  Link flags     : Keepalives
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 1066 (00:00:02 ago), Output: 1066 (00:00:02 ago)
  LCP state: Opened
  NCP state: inet: Opened, inet6: Opened, iso: Not-configured, mpls:
  Not-configured
  CHAP state: Closed
  CoS queues     : 8 supported, 8 maximum usable queues
  Last flapped   : 2006-10-06 01:01:36 PDT (02:57:27 ago)
  Input rate     : 88 bps (0 pps)
  Output rate    : 58144 bps (157 pps)
  DS1   alarms   : None
  DS1   defects  : None

  Logical interface e1-3/0/1.0 (Index 66) (SNMP ifIndex 238)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: PPP
    Bandwidth: 1984kbps
    Protocol inet, MTU: 1500
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 47.47.47.0/30, Local: 47.47.47.2, Broadcast: 47.47.47.3
    Protocol inet6, MTU: 1500
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 8b8b:8b01::/64, Local: 8b8b:8b01::2
      Addresses, Flags: Is-Preferred
        Destination: fe80::/64, Local: fe80::205:85ff:fec5:d3d0
```

Meaning The output shows a summary of interface information. Although the parent interface is ce1-3/0/1, the physical and logical clear-channel interfaces are named e1-3/0/1 and e1-3/0/1.0.

Related Topics For a complete description of show interfaces output, see the *JUNOS Interfaces Command Reference*.

Verifying ISDN PRI Configuration on Channelized T1/E1/ISDN PRI Interfaces

Purpose Verify that your configuration of ISDN PRI service on a channelized interface is correct.

Action From the J-Web interface, select **Configuration > View and Edit > View Configuration Text**. Alternatively, from configuration mode in the CLI, enter the show interfaces ct1-2/0/0 command.

Sample Output

```
user@host# show interfaces ct1-2/0/0

traceoptions {
    flag q921;
    file {
        isdnback;
    }
}
clocking external;
isdn-options {
    switch-type ni2;
}
dialer-options {
    isdn-dialer-group priority 1;
    }
partition 24 timeslots 24 interface-type dc;
partition 1-23 timeslots 1-23 interface-type bc;

[edit]
```

Meaning Verify that the output shows your intended ISDN PRI interface configuration.

Related Topics For more information about the format of a configuration file, see the *J-Web Interface User Guide* or the *JUNOS CLI User Guide*.

To additionally verify ISDN PRI configuration, see Verifying the ISDN Configuration on page 211.

Frequently Asked Questions About Channelized T1/E1/ISDN PRI Interfaces

Use answers to the following question to solve configuration problems on a channelized T1/E1/ISDN PRI interface:

What Clock Combinations Are Possible for Channelized T1/E1/ISDN PRI Drop and Insert?

When you configure the drop-and-insert feature on a channelized T1/E1/ISDN PRI PIM, you must ensure that both ports run on the same clock. The following clock combinations are valid:

- When port 0 is configured to use the *internal* clock, port 1 must also be configured to use the *internal* clock.
- When port 0 is configured to use the *external* clock, port 1 must be configured to run on the same clock, the *external clock for port 0*.

- When port 1 is configured to use the *external* clock, port 0 must be configured to run on the same clock, the *external clock for port 1*.

J-series devices connected to one another must have complementary clock sources configured. Consider a scenario where Device R1 is connected to Devices R2 and R3. Port 0 on the channelized T1/E1/ISDN PRI PIM of R1 is connected to R2, and port 1 is connected to R3. The drop-and-insert feature is configured on R1 to insert input coming from R2 on port 0 into port 1 for transmission to R3.

Devices R1, R2, and R3 can be configured in three ways, according to whether the drop-and-insert clock source on R1 is the external clock for port 0, the external clock for port 1, or the device's internal clock.

To configure the drop-and-insert interfaces on Device R1 to use the external clock for port 0:

1. On Device R2, configure:

```
user@hostR2# set interfaces ct1-6/0/0 partition 1 timeslots 1-10
user@hostR2# set interfaces ct1-6/0/0 partition 1 interface-type ds
user@hostR2# set interfaces ds-6/0/0:1 unit 0 family inet address 10.46.46.1/30
```

2. On Device R3, configure:

```
user@hostR3# set interfaces ct1-3/0/0 clocking external
user@hostR3# set interfaces ct1-3/0/0 partition 1 timeslots 1-10
user@hostR3# set interfaces ct1-3/0/0 partition 1 interface-type ds
user@hostR3# set interfaces ds-3/0/0:1 unit 0 family inet address 10.46.46.2/30
```

3. On Device R1, configure:

```
user@hostR1# set interfaces ct1-3/0/0 clocking external
user@hostR1# set interfaces ct1-3/0/0 partition 1 timeslots 1-10
user@hostR1# set interfaces ct1-3/0/0 partition 1 interface-type ds
user@hostR1# set interfaces ds-3/0/0:1 data-input interface ds-3/0/1:1
user@hostR1# set interfaces ct1-3/0/1 clocking external interface ct1-3/0/0
user@hostR1# set interfaces ct1-3/0/1 partition 1 timeslots 1-10
user@hostR1# set interfaces ct1-3/0/1 partition 1 interface-type ds
```

To configure the drop-and-insert interfaces on Device R1 to use the external clock for port 1:

1. On Device R2, configure:

```
user@hostR2# set interfaces ct1-6/0/0 clocking external
user@hostR2# set interfaces ct1-6/0/0 partition 1 timeslots 1-10
user@hostR2# set interfaces ct1-6/0/0 partition 1 interface-type ds
user@hostR2# set interfaces ds-6/0/0:1 unit 0 family inet address 10.46.46.1/30
```

2. On Device R3, configure:

```
user@hostR3# set interfaces ct1-3/0/0 partition 1 timeslots 1-10
user@hostR3# set interfaces ct1-3/0/0 partition 1 interface-type ds
user@hostR3# set interfaces ds-3/0/0:1 unit 0 family inet address 10.46.46.2/30
```

3. On Device R1, configure:

```
user@hostR1# set interfaces ct1-3/0/0 clocking external interface ct1-3/0/1
user@hostR1# set interfaces ct1-3/0/0 partition 1 timeslots 1-10
```

```
user@hostR1# set interfaces ct1-3/0/0 partition 1 interface-type ds
user@hostR1# set interfaces ds-3/0/0:1 data-input interface ds-3/0/1:1
user@hostR1# set interfaces ct1-3/0/1 clocking external
user@hostR1# set interfaces ct1-3/0/1 partition 1 timeslots 1-10
user@hostR1# set interfaces ct1-3/0/1 partition 1 interface-type ds
```

To configure the drop-and-insert interfaces on Device R1 to use the device's internal clock:

1. On Device R2, configure:

```
user@hostR2# set interfaces ct1-6/0/0 clocking external
user@hostR2# set interfaces ct1-6/0/0 partition 1 timeslots 1-10
user@hostR2# set interfaces ct1-6/0/0 partition 1 interface-type ds
user@hostR2# set interfaces ds-6/0/0:1 unit 0 family inet address 10.46.46.1/30
```

2. On Device R3, configure:

```
user@hostR3# set interfaces ct1-3/0/0 clocking external
user@hostR3# set interfaces ct1-3/0/0 partition 1 timeslots 1-10
user@hostR3# set interfaces ct1-3/0/0 partition 1 interface-type ds
user@hostR3# set interfaces ds-3/0/0:1 unit 0 family inet address 10.46.46.2/30
```

3. On Device R1, configure:

```
user@hostR1# set interfaces ct1-3/0/0 clocking internal
user@hostR1# set interfaces ct1-3/0/0 partition 1 timeslots 1-10
user@hostR1# set interfaces ct1-3/0/0 partition 1 interface-type ds
user@hostR1# set interfaces ds-3/0/0:1 data-input interface ds-3/0/1:1
user@hostR1# set interfaces ct1-3/0/1 clocking internal
user@hostR1# set interfaces ct1-3/0/1 partition 1 timeslots 1-10
user@hostR1# set interfaces ct1-3/0/1 partition 1 interface-type ds
```

Chapter 6

Configuring Digital Subscriber Line Interfaces

The J-series device supports DSL features including ATM-over-ADSL and ATM-over-SHDSL interfaces.

You can use either J-Web Quick Configuration or a configuration editor to configure ATM-over-ADSL or ATM-over-SHDSL interfaces.

NOTE: Payload loopback functionality is not supported on ATM-over-SHDSL interfaces.

This chapter contains the following topics.

DSL Terms

Before configuring DSL on a J-series device, become familiar with the terms defined in Table 38 on page 125.

Table 38: DSL Terms

Term	Definition
asymmetric digital subscriber line (ADSL) interface	Physical WAN interface for connecting a J-series device to a digital subscriber line access multiplexer (DSLAM). An ADSL interface allocates line bandwidth asymmetrically with downstream (provider-to-customer) data rates of up to 8 Mbps for ADSL, 12 Mbps for ADSL2, and 25 Mbps for ADSL2 + , and upstream (customer-to-provider) rates of up to 800 Kbps for ADSL and 1 Mbps for ADSL2 and ADSL2 + , depending on the implementation.
ADSL2 interface	An ADSL interface that supports ITU-T Standards G.992.3 and G.992.4 and allocates downstream (provider-to-customer) data rates of up to 12 Mbps and upstream (customer-to-provider) rates of up to 1 Mbps.

Table 38: DSL Terms *(continued)*

Term	Definition
ADSL2 + interface	An ADSL interface that supports ITU-T Standard G.992.5 and allocates downstream (provider-to-customer) data rates of up to 25 Mbps and upstream (customer-to-provider) rates of up to 1 Mbps.
Annex A	ITU-T Standard G.992.1 that defines how ADSL works over plain old telephone service (POTS) lines.
Annex B	ITU-T Standard G.992.1 that defines how ADSL works over Integrated Services Digital Network (ISDN) lines.
ITU-T G.991.2	International Telecommunication Union standard describing a data transmission method for symmetric high-speed digital subscriber line (SHDSL) as a means for data transport in telecommunications access networks. The standard also describes the functionality required for interoperability of equipment from various manufacturers.
ITU-T G.992.1	International Telecommunication Union standard that requires the downstream (provider-to-customer) data transmission to consist of full-duplex low-speed bearer channels and simplex high-speed bearer channels. In the upstream (customer-to-provider) transmissions, only low-speed bearer channels are provided.
ITU-T G.994.1	International Telecommunication Union standard describing the types of signals, messages, and procedures exchanged between digital subscriber line (DSL) equipment when the operational modes of equipment need to be automatically established and selected.
ITU-T G.997.1	International Telecommunication Union standard describing the physical layer management for asymmetric digital subscriber line (ADSL) transmission systems. The standard specifies the means of communication on a transport transmission channel defined in the physical layer recommendations. In addition, the standard describes the content and syntax of network elements for configuration, fault management, and performance management.
symmetric high-speed digital subscriber line (G.SHDSL)	Physical WAN symmetric DSL interface capable of sending and receiving high-speed symmetrical data streams over a single pair of copper wires at rates between 192 Kbps and 2.31 Mbps. G.SHDSL incorporates features of other DSL technologies such as asymmetric DSL and transports T1, E1, ISDN, Asynchronous Transfer Mode (ATM), and IP signals.
symmetric high-speed digital subscriber line (SHDSL) transceiver unit–remote (STU–R)	Equipment that provides symmetric high-speed digital subscriber line (SHDSL) connections to remote user terminals such as data terminals or telecommunications equipment.

Before You Begin

Before you begin configuring DSL interfaces, complete the following tasks:

- Establish basic connectivity. See the Getting Started Guide for your device.
- If you do not already have a basic understanding of physical and logical interfaces and Juniper Networks interface conventions, read “Interfaces Overview” on page 11.
- Configure network interfaces as necessary. See “Configuring Ethernet, DS1, DS3, and Serial Interfaces” on page 73.

Configuring ATM-over-ADSL Interfaces

J-series devices with ADSL Annex A or Annex B PIMs can use an Asynchronous Transfer Mode (ATM) interface to send network traffic through a point-to-point connection to a DSL access multiplexer (DSLAM).

NOTE: You can configure J-series devices with ADSL PIMs for connections through ADSL only, not for direct ATM connections.

To configure Point-to-Point Protocol (PPP), see the *JUNOS Network Interfaces Configuration Guide*.

You configure the underlying ADSL interface as an ATM interface, with an interface name of at-*pim*/0/*port*. (For information about interface names, see "Network Interface Naming" on page 16.) Multiple encapsulation types are supported on both the physical and logical ATM-over-ADSL interface.

This section contains the following topics:

Configuring an ATM-over-ADSL Interface with Quick Configuration

The Quick Configuration pages allow you to configure ATM-over-ADSL interfaces on a J-series device.

To configure an ATM-over-ADSL interface with Quick Configuration:

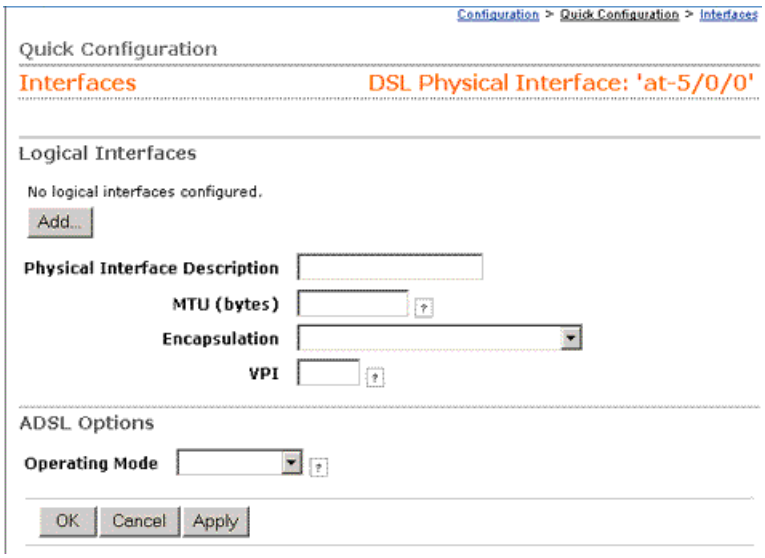
1. In the J-Web user interface, select **Configuration > Quick Configuration > Interfaces**.

   A list of the network interfaces present on the device is displayed. (see "Network Interface Naming" on page 16.)

2. Select the at-*pim*/0/*port* interface name for the ADSL port you want to configure.

   The ATM-over-ADSL Quick Configuration page is displayed, as shown in Figure 22 on page 128.

Figure 22: ATM-over-ADSL Interfaces Quick Configuration Page



3. Enter information into the ATM-over-ADSL Quick Configuration pages, as described in Table 39 on page 128.
4. From the ATM-over-ADSL Quick Configuration main page, click one of the following buttons:
   - To apply the configuration and stay on the ATM-over-ADSL Quick Configuration main page, click **Apply**.
   - To apply the configuration and return to the Interfaces Quick Configuration page, click **OK**.
   - To cancel your entries and return to the Interfaces Quick Configuration page, click **Cancel**.
5. To verify that the ATM-over-ADSL interface is configured properly, see "Verifying DSL Interface Configuration" on page 147.

Table 39: ATM-over-ADSL Interface Quick Configuration Pages Summary

Field	Function	Your Action
Configuring Logical Interfaces		
Logical Interfaces	Lists the logical interfaces for this ATM-over-ADSL physical interface.	■ To add a logical interface, click **Add**. ■ To edit a logical interface, select the interface from the list. ■ To delete a logical interface, select the check box next to the name and click **Delete**.
Adding or Editing a Logical Interface		
Add logical interfaces	Defines one or more logical units that you connect to this physical ADSL interface.	Click **Add**.

Table 39: ATM-over-ADSL Interface Quick Configuration Pages Summary *(continued)*

Field	Function	Your Action
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to more clearly identify it in monitoring displays.
Encapsulation	Specifies the type of encapsulation on the DSL logical interface.	From the list, select one of the following types of encapsulations. For ATM-over-ADSL interfaces that use inet (IPv4) protocols only, select one of the following: ■ **ATM VC multiplexing**—Use ATM virtual circuit multiplex encapsulation. ■ **ATM NLPID**—Use ATM network layer protocol identifier (NLPID) encapsulation. ■ **Cisco-compatible ATM NLPID**—Use Cisco NLPID encapsulation. ■ **Ethernet over ATM (LLC/SNAP)**—For interfaces that carry IPv4 traffic, use Ethernet over logical link control (LLC) encapsulation. You cannot configure multipoint interfaces if you use this type of encapsulation. For ATM-over-ADSL for PPP-over-ATM (PPPoA) interfaces only, select one of the following: ■ **ATM PPP over AAL5/LLC**—Use AAL5 logical link control (LLC) encapsulation. ■ **ATM PPP over Raw AAL5**—Use AAL5 multiplex encapsulation. For other encapsulation types on the ATM-over-ADSL interfaces, select one of the following: ■ **PPPoE over ATM (LLC/SNAP)**—Use PPP over Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure the interface address. Instead you configure the interface address on the PPP interface. ■ **Ethernet over ATM (LLC/SNAP)**—Use ATM subnetwork attachment point (SNAP) encapsulation.
VCI	Configures the ATM virtual circuit identifier (VCI) for the interface.	In the VCI box, type the number for the VCI.
Add IPv4 address prefixes and destinations	Specifies one or more IPv4 addresses and destination addresses.	Click **Add**.

Table 39: ATM-over-ADSL Interface Quick Configuration Pages Summary *(continued)*

Field	Function	Your Action
IPv4 Address Prefix	Specifies an IPv4 address for the interface.	Type an IPv4 address and prefix. For example: 10.10.10.10/24
Destination Address	Specifies the destination address.	1. Type an IPv4 address for the destination. 2. Click **OK**.
Configuring Physical Interface Properties		
Physical Interface Description	(Optional) Adds supplementary information about the physical ATM-over-ADSL interface.	Type a text description of the physical ATM-over-ADSL interface to more clearly identify it in monitoring displays. Specify that it is an ADSL interface.
MTU (bytes)	Specifies the maximum transmit size of a packet for the ATM-over-ADSL interface.	Type a value from 256 to 9192.
Encapsulation	Selects the type of encapsulation for traffic on this physical interface.	From the list, select the type of encapsulation for this ATM-over-ADSL interface: ■ **ATM permanent virtual circuits**—Use this type of encapsulation for PPP over ATM (PPPoA) over ADSL interfaces. This is the default encapsulation for ATM-over-ADSL interfaces. ■ **Ethernet over ATM encapsulation**—Use this type of encapsulation for PPP over Ethernet (PPPoE) over ATM-over-ADSL interfaces that carry IPv4 traffic.
VPI	Configures the ATM virtual path identifier for the interface.	Type a VPI value between 0 and 255.
Configuring ADSL Options		

Table 39: ATM-over-ADSL Interface Quick Configuration Pages Summary *(continued)*

Field	Function	Your Action
Operating Mode	Specifies the type of DSL operating mode for the ATM-over-ADSL interface.	From the list, select one of the following types of DSL operating modes—for example **auto**. For Annex A or Annex B, select one of the following: ■ **auto**—Configure the ADSL interface to autonegotiate settings with the DSLAM located at the central office. For Annex A, the ADSL interface trains in either ANSI T1.413 Issue II mode or ITU G.992.1 mode. For Annex B, the ADSL interface trains in ITU G.992.1 mode. ■ **itu-dmt**—Configure the ADSL interface to train in ITU G.992.1 mode. For Annex A only, select one of the following: ■ **adsl2plus**—Configure the ADSL interface to train in ITU G.992.5 mode. You can configure this mode only when it is supported on the DSLAM. ■ **itu-dmt-bis**—Configure the ADSL interface to train in ITU G.992.3 mode. You can configure this mode only when it is supported on the DSLAM. ■ **ansi-dmt**—Configure the ADSL interface to train in the ANSI T1.413 Issue II mode. For Annex B only, select one of the following: ■ **etsi**—Configure the ADSL line to train in the ETSI TS 101 388 V1.3.1 mode. ■ **itu-annexb-ur2**—Configure the ADSL line to train in the G.992.1 Deutsche Telekom UR-2 mode. ■ **itu-annexb-non-ur2**—Configure the ADSL line to train in the G.992.1 Non-UR-2 mode.

Adding an ATM-over-ADSL Network Interface with a Configuration Editor

To configure ATM-over-ADSL network interfaces for the J-series device with a configuration editor:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 40 on page 132.
3. If you are finished configuring the J-series device, commit the configuration.
4. Go on to one of the following procedures:

- To enable authentication on the interface, see "Configuring CHAP on DSL Interfaces (Optional)" on page 146.
- To configure PPP over Ethernet (PPPoE) encapsulation on an Ethernet interface or on an ATM-over-ADSL interface, see "Configuring Point-to-Point Protocol over Ethernet" on page 157.

5. To check the configuration, see "Verifying DSL Interface Configuration" on page 147.

Table 40: Adding an ATM-over-ADSL Network Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces at-2/0/0
Create the new interface—for example, at-2/0/0.	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type at-2/0/0. 3. Click **OK**.	
Configuring Physical Properties		

Table 40: Adding an ATM-over-ADSL Network Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor

Configure ATM virtual path identifier (VPI) options for the interface—for example, at-2/0/0.
■ ATM VPI—A number between 0 and 255—for example, 25.
■ Operation, Maintenance, and Administration (OAM) F5 loopback cell thresholds ("liveness") on ATM virtual circuits. The range is between 1 and 255, and the default is 5 cells.
■ Down count—Number of consecutive OAM loopback cells an ATM virtual circuit must lose to be identified as unavailable—for example, 200.
■ Up count—Number of consecutive OAM loopback cells an ATM virtual interface must receive to be identified as operational—for example, 200.
■ OAM period—Interval, in seconds, at which OAM cells are transmitted on ATM virtual circuits—for example, 100. The range is between 1 and 900 seconds.
	1. In the Interface name box, select **at-2/0/0**.
	2. Next to Atm options, click **Configure**.
	3. Next to Vpi, click **Add new entry**.
	4. In the Vpi number box, type 25.
	5. Click **OK**.
	6. In the Actions box, click **Edit**.
	7. Next to Oam liveness, click **Configure**.
	8. In the Down count box, type 200.
	9. In the Up count box, type 200.
	10. Click **OK**.
	11. Next to Oam period, click **Configure**.
	12. From the Oam period choices list, select **Oam period**.
	13. In the Oam period box, type 100.
	14. Click **OK** until you return to the Interface page.
		1. To configure the VPI value, enter
		`set atm-options vpi 25`
		2. To configure OAM liveness values on a VPI, enter
		`set atm-options vpi 25 oam-liveness up-count 200 down-count 200`
		3. To configure the OAM period, enter
		`set atm-options vpi 25 oam-period 100`

Table 40: Adding an ATM-over-ADSL Network Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure the type of DSL operating mode for the ATM-over-ADSL interface—for example auto. Annex A and Annex B support the following operating modes: ■ auto—Configures the ADSL interface to autonegotiate settings with the DSLAM located at the central office. For Annex A, the ADSL interface trains in either ANSI T1.413 Issue II mode or ITU G.992.1 mode. For Annex B, the ADSL interface trains in ITU G.992.1 mode. ■ itu-dmt—Configures the ADSL interface to train in ITU G.992.1 mode. Annex A supports the following operating modes: ■ adsl2plus—Configures the ADSL interface to train in ITU G.992.5 mode. You can configure this mode only when it is supported on the DSLAM. ■ itu-dmt-bis—Configures the ADSL interface to train in ITU G.992.3 mode. You can configure this mode only when it is supported on the DSLAM. ■ ansi-dmt—Configures the ADSL interface to train in the ANSI T1.413 Issue II mode. Annex B supports the following operating modes: ■ etsi—Configures the ADSL line to train in the ETSI TS 101 388 V1.3.1 mode. ■ itu-annexb-ur2—Configures the ADSL line to train in the G.992.1 Deutsche Telekom UR-2 mode. ■ itu-annexb-non-ur2—Configures the ADSL line to train in the G.992.1 Non-UR-2 mode.	1. Next to Dsl options, click **Configure**. 2. From the Operating Mode list, select **auto**. 3. Click **OK**.	Enter set dsl-options operating-mode auto
Configure the encapsulation type—for example, ethernet-over-atm. ■ atm-pvc—ATM permanent virtual circuits is the default encapsulation for ATM-over-ADSL interfaces. For PPP over ATM (PPPoA) over ADSL interfaces, use this type of encapsulation. ■ ethernet-over-atm—Ethernet over ATM encapsulation. For PPP over Ethernet (PPPoE) over ATM-over-ADSL interfaces that carry IPv4 traffic, use this type of encapsulation.	From the Encapsulation list, select **ethernet-over-atm**.	Enter set encapsulation ethernet-over-atm
Configuring Logical Properties		

Table 40: Adding an ATM-over-ADSL Network Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Add the logical interface. Set a value from 0 and 16385—for example, 3. Add other values if required by your network.	1. Scroll down the page to Unit, and click **Add new entry**. 2. In the Interface unit number box, type 3. 3. Enter other values in the fields required by your network.	Enter set unit 3
Configure encapsulation for the ATM-for-ADSL logical unit—for example, atm-nlpid. The following encapsulations are supported on the ATM-over-ADSL interfaces that use inet (IP) protocols only: ■ atm-vc-mux–Use ATM virtual circuit multiplex encapsulation. ■ atm-nlpid–Use ATM network layer protocol identifier (NLPID) encapsulation. ■ atm-cisco-nlpid–Use Cisco NLPID encapsulation. ■ ether-over-atm-llc–For interfaces that carry IPv4 traffic, use Ethernet over LLC encapsulation. You cannot configure multipoint interfaces if you use this type of encapsulation. The following encapsulations are supported on the ATM-over-ADSL for PPP-over-ATM (PPPoA) interfaces only. (For a sample PPPoA configuration, see "Verifying Interface Configuration" on page 106.) ■ atm-ppp-llc— AAL5 logical link control (LLC) encapsulation. ■ atm-ppp-vc-mux—Use AAL5 multiplex encapsulation. Other encapsulation types supported on the ATM-over-ADSL interfaces: ■ ppp-over-ether-over-atm-llc—Use PPP over Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure the interface address. Instead you configure the interface address on the PPP interface. ■ atm-snap—Use ATM subnetwork attachment point (SNAP) encapsulation.	From the Encapsulation list, select **atm-nlpid**.	Enter set unit 3 encapsulation atm-nlpid

Table 40: Adding an ATM-over-ADSL Network Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure Operation, Maintenance, and Administration (OAM) options for ATM virtual circuits: ■ OAM F5 loopback cell thresholds ("liveness") on ATM virtual circuits. The range is between 1 and 255, and the default is 5 cells. ■ Down count—Number of consecutive OAM loopback cells an ATM virtual circuit must lose to be identified as unavailable—for example, 200. ■ Up count—Number of consecutive OAM loopback cells an ATM virtual interface must receive to be identified as operational—for example, 200. ■ OAM period—Interval, in seconds, at which OAM cells are transmitted on ATM virtual circuits—for example, 100. The range is between 1 and 900 seconds.	1. Next to Oam liveness, click **Configure**. 2. In the Down count box, type 200. 3. In the Up count box, type 200. 4. Click **OK**. 5. Next to Oam period, click **Configure**. 6. From the Oam period choices list, select **Oam period**. 7. In the Oam period box, type 100. 8. Click **OK**.	1. To configure OAM liveness values for an ATM virtual circuit, enter set unit 3 oam-liveness up-count 200 down-count 200 2. To configure the OAM period, enter set unit 3 oam-period 100
Add the Family protocol type—for example, inet.	1. In the Inet box, select **Yes** and click **Configure**. 2. Enter the values in the fields required by your network. 3. Click **OK**.	Enter set unit 3 family inet Commands vary depending on the protocol type.
Configure ATM virtual channel identifier (VCI) options for the interface. ■ ATM VCI type—vci. ■ ATM VCI value—A number between 0 and 4089—for example, 35— with VCIs 0 through 31 reserved.	1. From the Vci Type list, select **vci**. 2. In the Vci box, type 35. 3. Click **OK** until you return to the Interfaces page.	1. To configure the VCI value, enter set unit 3 vci 35

Configuring ATM-over-SHDSL Interfaces

J-series devices with G.SHDSL interfaces can use an Asynchronous Transfer Mode (ATM) interface to send network traffic through a point-to-point connection to a DSL access multiplexer (DSLAM).

NOTE: You can configure J-series devices with a G.SHDSL interface for connections through SHDSL only, not for direct ATM connections.

J-series devices with a 2-port G.SHDSL interface installed support the following modes. You can configure only one mode on each interface.

- 2-port two-wire mode (Annex A or Annex B)—Supports autodetection of the line rate or fixed line rates and provides network speeds from 192 Kbps to 2.3 Mbps

in 64-Kbps increments. Two-wire mode provides two separate, slower SHDSL interfaces.

- 1-port four-wire mode (Annex A or Annex B)—Supports fixed line rates only and provides network speeds from 384 Kbps to 4.6 Mbps in 128-Kbps increments, doubling the bandwidth. Four-wire mode provides a single, faster SHDSL interface.

To configure Point-to-Point Protocol (PPP), see the *JUNOS Network Interfaces Configuration Guide*.

You configure the underlying G.SHDSL interface as an ATM interface, with an interface name of at-*pim*/0/*port*. (see "Network Interface Naming" on page 16.) Multiple encapsulation types are supported on both the physical and logical ATM-over-SHDSL interface.

This section contains the following topics:

Configuring an ATM-over-SHDSL Interface with Quick Configuration

The ATM-over-SHDSL Quick Configuration pages allow you to configure ATM-over-SHDSL interfaces and SHDSL options.

To configure an ATM-over-SHDSL interface with Quick Configuration:

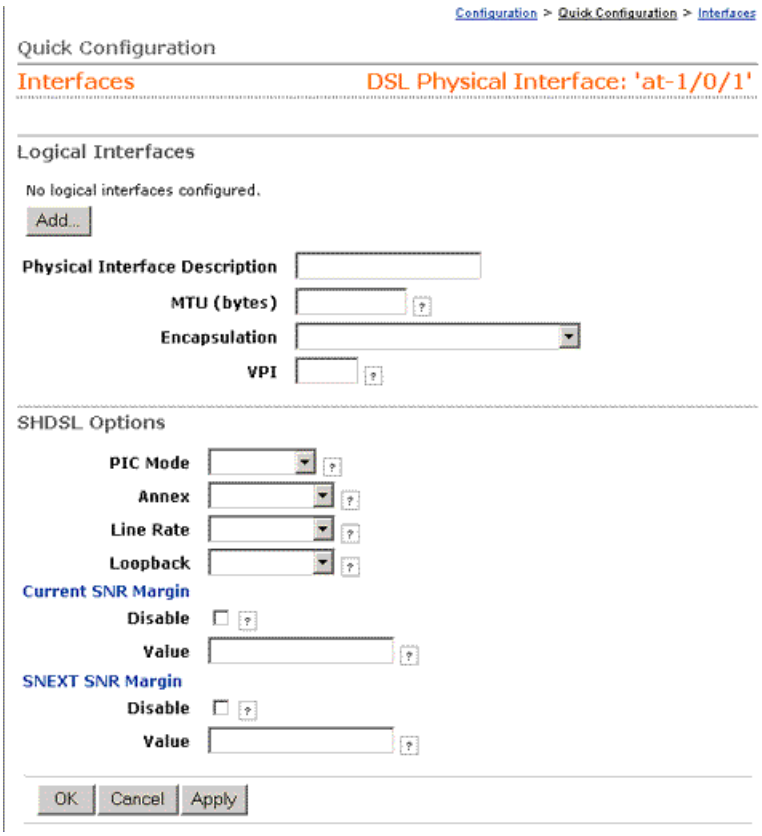
1. In the J-Web interface, select **Configuration > Quick Configuration > Interfaces**.

   A list of the network interfaces installed on the device is displayed. (see "Network Interface Naming" on page 16.)

2. Select an at-*pim*/0/*port* interface from the list.

   The ATM-over-SHDSL Interface Quick Configuration page is displayed, as shown in Figure 23 on page 138.

Figure 23: ATM-over-SHDSL Interfaces Quick Configuration Main Page



3. Enter information into the ATM-over-SHDSL Quick Configuration page, as described in Table 41 on page 138.
4. From the ATM-over-SHDSL Quick Configuration main page, click one of the following buttons:
   - To apply the configuration and stay in the ATM-over-SHDSL interface Quick Configuration main page, click **Apply**.
   - To apply the configuration and return to the Interfaces Quick Configuration page, click **OK**.
   - To cancel your entries and return to the Interfaces Quick Configuration page, click **Cancel**.
5. To verify the ATM-over-SHDSL interface properties, see "Verifying DSL Interface Configuration" on page 147.

Table 41: ATM-over-SHDSL Interface Quick Configuration Pages Summary

Field	Function	Your Action
Configuring Logical Interfaces		

Table 41: ATM-over-SHDSL Interface Quick Configuration Pages Summary *(continued)*

Field	Function	Your Action
Logical Interface Name	Lists the logical interfaces for the ATM-over-SHDSL physical interface.	If you have not added an at-*pim*/0/*port* interface, click **Add** and enter the information required in the Interfaces Quick Configuration fields. If you have already configured a logical interface, select the interface name from the **Logical Interface Name** list. To delete a logical interface, select the interface and click **Delete**.
Adding or Editing a Logical Interface		
Add logical interfaces	Defines one or more logical units that you connect to this physical ADSL interface.	Click **Add**.
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to clearly identify it in monitoring displays.
Encapsulation	Specifies the type of encapsulation on the SHDSL logical interface.	From the list, select one of the following types of encapsulations. For ATM-over-SHDSL interfaces that use inet (IPv4) protocols only, select one of the following: ■ **Cisco-compatible ATM NLPID**—Use Cisco NLPID encapsulation. ■ **ATM NLPID**—Use ATM network layer protocol identifier (NLPID) encapsulation. ■ **ATM PPP over AA5/LLC**—Use AAL5 logical link control (LLC) encapsulation. ■ **ATM PPP over raw AAL5**—Use AAL5 multiplex encapsulation. ■ **ATM LLC/SNAP**—For interfaces that carry IPv4 traffic, use ATM over logical link control (LLC) encapsulation. You cannot configure multipoint interfaces if you use this type of encapsulation. ■ **ATM VC multiplexing**—Use ATM virtual circuit multiplex encapsulation. For other encapsulation types on the ATM-over-SHDSL interfaces, select one of the following: ■ **Ethernet over ATM (LLC/SNAP)**—Use ATM subnetwork attachment point (SNAP) encapsulation. ■ **PPPoE over ATM (LLC/SNAP)**—Use PPP over Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure the interface address. Instead you configure the interface address on the PPP interface.

Table 41: ATM-over-SHDSL Interface Quick Configuration Pages Summary *(continued)*

Field	Function	Your Action
VCI	Configures the ATM virtual circuit identifier (VCI) for the interface.	In the VCI box, type the number for the VCI.
Add IPv4 address prefixes and destinations	Specifies one or more IPv4 addresses and destination addresses.	Click **Add**.
IPv4 Address Prefix	Specifies an IPv4 address for the interface.	Type an IPv4 address and prefix. For example: 10.10.10.10/24
Destination Address	Specifies the destination address.	1. Type an IPv4 address for the destination. 2. Click **OK**.
Configuring Physical Properties		
Physical Interface Description	Describes the physical interface description information. (Optional)	Type a description of the interface.
MTU (bytes)	Specifies the maximum transmission unit (MTU) size, in bytes, of a packet on the ATM-over-SHDSL interface.	Type a value for the byte size—for example, 1500.
Encapsulation	Selects the type of encapsulation for traffic on the physical interface.	Select one of the following types of encapsulation: ■ **ATM permanent virtual circuits**—Use this type of encapsulation for PPP over ATM (PPPoA) over SHDSL interfaces. This is the default encapsulation for ATM-over-SHDSL interfaces. ■ **Ethernet over ATM encapsulation**—Use this type of encapsulation for PPP over Ethernet (PPPoE) over ATM-over-SHDSL interfaces that carry IPv4 traffic.
VPI	Configures the ATM virtual path identifier (VPI) for the interface.	In the VPI field, type a number between 0 and 255—for example, 25.
Configuring SHDSL Options		
PIC Mode	Specifies the mode on the ATM-over-SHDSL interface.	Select either of the following: ■ **1-port-atm**—1-port four-wire mode ■ **2-port-atm**—2-port two-wire mode
Annex	Specifies the type of annex for the interface. Annex defines the System Reference Model for connecting DSL networks to the plain old telephone service (POTS).	Select one of the following: ■ **Annex A**—Used in North American network implementations. ■ **Annex B**—Used in European network implementations.
Line Rate	Specifies the available line rates, in kilobits per second, to use on an G.SHDSL interface.	Select the appropriate value. For 2-port-atm mode only, you can select **auto**, which automatically selects a line rate.

Table 41: ATM-over-SHDSL Interface Quick Configuration Pages Summary *(continued)*

Field	Function	Your Action
Loopback	Specifies the type of loopback testing for the interface. Loopback testing is a diagnostic procedure in which a signal is transmitted and returned to the sending device after passing through all or a portion of a network or circuit. The returned signal is compared with the transmitted signal in order to evaluate the integrity of the equipment or transmission path. TEST	Select one of the following: ■ **local**—Used for testing the SHDSL equipment with local network devices. ■ **payload**—Used to command the remote configuration to send back the received payload. ■ **remote**—Used to test SHDSL with a remote network configuration.
Current SNR Margin Disable Value	Specifies the signal-to-noise ratio (SNR) margin or disables SNR.	To disable Current SNR Margin, select Disable. To configure a specific value, type a number from 0 to 10—for example, 5. The range is 0 dB to 10 dB with a default value of 0.
SNEXT SNR Margin Disable Value	Sets a value, from –10 dB to 10 dB, for the self-near-crosstalk (SNEXT) SNR margin, or disables SNEXT.	To disable SNEXT SNR Margin, select **Disable**. To configure a specific value, type a number from –10 to 10—for example, 5.

Adding an ATM-over-SHDSL Interface with a Configuration Editor

To configure ATM-over-SHDSL network interfaces for the J-series device with a configuration editor:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 42 on page 142.
3. If you are finished configuring the J-series device, commit the configuration.
4. Go on to one of the following procedures:
   - To enable authentication on the interface, see "Configuring CHAP on DSL Interfaces (Optional)" on page 146.
   - To configure PPP over Ethernet (PPPoE) encapsulation on an Ethernet interface or on an ATM-over-SHDSL interface, see "Configuring Point-to-Point Protocol over Ethernet" on page 157.
5. To check the configuration, see "Verifying DSL Interface Configuration" on page 147.

Table 42: Adding an ATM-over-SHDSL Network Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Chassis** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Chassis, click **Configure**.	From the [edit] hierarchy level, enter set chassis fpc 6 pic 0 shdsl pic-mode 1-port-atm
Set the ATM-over-SHDSL mode on the G.SHDSL interface, if required. By default, G.SHDSL interfaces are enabled in two-wire Annex B mode. To configure the four-wire mode on the G.SHDSL interface, follow the tasks in this table.	1. Next to Fpc, click **Add new entry**. 2. In the Slot box, type 6. 3. Next to Pic, click **Add new entry**. 4. In the Slot box, type 0. 5. Next to Shdsl, click **Configure**. 6. From the Pic mode menu, select **1-port-atm**. 7. Click **OK** until you return to the main Configuration page.	
Navigate to the **Interfaces** level in the configuration hierarchy.	On the main Configuration page next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces at-2/0/0
Create the new interface—for example, at-2/0/0.	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type at-2/0/0. 3. Click **OK**.	
Configuring Physical Properties		

Table 42: Adding an ATM-over-SHDSL Network Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure ATM virtual path identifier (VPI) options for the interface—for example, at-2/0/0. ■ ATM VPI—A number between 0 and 255—for example, 25. ■ Operation, Maintenance, and Administration (OAM) F5 loopback cell thresholds ("liveness") on ATM virtual circuits. The range is between 1 and 255, and the default is 5 cells. ■ Down count—Number of consecutive OAM loopback cells an ATM virtual circuit must lose to be identified as unavailable—for example, 200. ■ Up count—Number of consecutive OAM loopback cells an ATM virtual interface must receive to be identified as operational—for example, 200. ■ OAM period—Interval, in seconds, at which OAM cells are transmitted on ATM virtual circuits—for example, 100. The range is between 1 and 900 seconds.	1. In the Interface name box, select **at-2/0/0**. 2. Next to Atm options, click **Configure**. 3. Next to Vpi, click **Add new entry**. 4. In the Vpi number box, type 25. 5. Click OK. 6. In the Actions box, click **Edit**. 7. Next to Oam liveness, click **Configure**. 8. In the Down count box, type 200. 9. In the Up count box, type 200. 10. Click **OK**. 11. Next to Oam period, click **Configure**. 12. From the Oam period choices list, select **Oam period**. 13. In the Oam period box, type 100. 14. Click **OK** until you return to the Interface page.	1. To configure the VPI value, enter set atm-options vpi 25 2. To configure OAM liveness values on a VPI, enter set atm-options vpi 25 oam-liveness up-count 200 down-count 200 3. To configure the OAM period, enter set atm-options vpi 25 oam-period 100
Configure the encapsulation type—for example, ethernet-over-atm. ■ atm-pvc—ATM permanent virtual circuits is the default encapsulation for ATM-over-SHDSL interfaces. For PPP over ATM (PPPoA) over SHDSL interfaces, use this type of encapsulation. ■ ethernet-over-atm—Ethernet over ATM encapsulation. For PPP over Ethernet (PPPoE) over ATM-over-SHDSL interfaces that carry IPv4 traffic, use this type of encapsulation.	From the Encapsulation list, select **ethernet-over-atm**.	Enter set encapsulation ethernet-over-atm
Set the annex type. ■ **Annex A**—Used in North American network implementations. ■ **Annex B**—Used in European network implementations.	1. Next to Shdsl options, click **Configure**. 2. From the Annex list, select **Annex-a**.	Enter set shdsl-options annex annex-a

Table 42: Adding an ATM-over-SHDSL Network Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure the SHDSL line rate for the ATM-over-SHDSL interface–for example, automatic selection of the line rate. The following values are available: ■ auto—Automatically selects a line rate. This option is available only in two-wire mode and is the default value. ■ 192 Kbps or higher—Speed of transmission of data on the SHDSL connection. In the four-wire mode, the default line rate is 4608 Kbps.	From the Line Rate list, select **auto**.	Enter set shdsl-options line-rate auto
Configure the loopback option for testing the SHDSL connection integrity–for example, local loopback. The following values are available: ■ local—Used for testing the SHDSL equipment with local network devices. ■ payload—Used to command the remote configuration to send back the received payload. ■ remote—Used to test SHDSL with a remote network configuration.	From the Loopback list, select **local**.	Enter set shdsl-options loopback local
Configure the signal-to-noise ratio (SNR) margin—for example, 5 dB for either or both of the following thresholds: ■ current—Line trains at higher than current noise margin plus SNR threshold. The range is 0 to 10 dB. The default value is 0. ■ snext—Line trains at higher than self-near-end crosstalk (SNEXT) threshold. The default value is disabled. Setting the SNR creates a more stable SHDSL connection by making the line train at a SNR margin higher than the threshold. If any external noise below the threshold is applied to the line, the line remains stable. You can also disable the SNR margin thresholds.	1. Next to Snr margin, select **Yes**, then click **Configure**. 2. From the Current list, select **Enter Specific Value**. 3. In the Value box, type 5. 4. From the Snext list, select **Enter Specific Value**. 5. In the Value box, type 5. 6. Click **OK** until you return to the Interface page.	1. Enter set shdsl-options snr-margin current 5 2. Enter set shdsl-options snr-margin snext 5
Configuring Logical Properties		
Add the logical interface. Set a value from 0 and 16385—for example, 3. Add other values if required by your network.	1. Scroll down the page to Unit, and click **Add new entry**. 2. In the Interface unit number box, type 3. 3. Enter other values in the fields required by your network.	Enter set unit 3

Table 42: Adding an ATM-over-SHDSL Network Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure encapsulation for the ATM-for-SHDSL logical unit—for example, atm-nlpid. The following encapsulations are supported on the ATM-over-SHDSL interfaces that use inet (IP) protocols only: ■ atm-vc-mux—Use ATM virtual circuit multiplex encapsulation. ■ atm-nlpid—Use ATM network layer protocol identifier (NLPID) encapsulation. ■ atm-cisco-nlpid—Use Cisco NLPID encapsulation. ■ ether-over-atm-llc—For interfaces that carry IPv4 traffic, use Ethernet over LLC encapsulation. You cannot configure multipoint interfaces if you use this type of encapsulation. The following encapsulations are supported on the ATM-over-SHDSL for PPP-over-ATM (PPPoA) interfaces only. (For a sample PPPoA configuration, see "Verifying Interface Configuration" on page 106.) ■ atm-ppp-llc—AAL5 logical link control (LLC) encapsulation. ■ atm-ppp-vc-mux—Use AAL5 multiplex encapsulation. Other encapsulation types supported on the ATM-over-SHDSL interfaces: ■ ppp-over-ether-over-atm-llc—Use PPP over Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure the interface address. Instead you configure the interface address on the PPP interface. ■ atm-snap—Use ATM subnetwork attachment point (SNAP) encapsulation.	From the Encapsulation list, select **atm-nlpid**.	Enter set unit 3 encapsulation atm-nlpid

Configure Operation, Maintenance, and Administration (OAM) options for ATM virtual circuits:
■ OAM F5 loopback cell thresholds ("liveness") on ATM virtual circuits. The range is between 1 and 255, and the default is 5 cells.
  ■ Down count—Number of consecutive OAM loopback cells an ATM virtual circuit must lose to be identified as unavailable—for example, 200.
  ■ Up count—Number of consecutive OAM loopback cells an ATM virtual interface must receive to be identified as operational—for example, 200.
■ OAM period—Interval, in seconds, at which OAM cells are transmitted on ATM virtual circuits—for example, 100. The range is between 1 and 900 seconds.
	1. Next to Oam liveness, click **Configure**.
	2. In the Down count box, type 200.
	3. In the Up count box, type 200.
	4. Click **OK**.
	5. Next to Oam period, click **Configure**.
	6. From the Oam period choices list, select **Oam period**.
	7. In the Oam period box, type 100.
	8. Click **OK**.
		1. To configure OAM liveness values for an ATM virtual circuit, enter
		
		set unit 3 oam-liveness up-count 200 down-count 200
		
		2. To configure the OAM period, enter
		
		set unit 3 oam-period 100

Table 42: Adding an ATM-over-SHDSL Network Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Add the Family protocol type—for example, inet.	1. In the Inet box, select **Yes** and click **Configure**. 2. Enter the values in the fields required by your network. 3. Click **OK**.	Enter set unit 3 family inet Commands vary depending on the protocol type.
Configure ATM virtual channel identifier (VCI) options for the interface. ■ ATM VCI type—vci. ■ ATM VCI value—A number between 0 and 4089—for example, 35—with VCIs 0 through 31 reserved.	1. From the Vci type list, select **vci**. 2. In the Vci box, type 35. 3. Click **OK** until you return to the Interfaces page.	1. To configure the VCI value, enter set unit 3 vci 35

Configuring CHAP on DSL Interfaces (Optional)

For interfaces with PPPoA encapsulation, you can optionally configure interfaces to support the PPP Challenge Handshake Authentication Protocol (CHAP). When you enable CHAP on an interface, the interface can authenticate its peer and be authenticated by its peer.

If you set the passive option to handle incoming CHAP packets only, the interface does not challenge its peer. However, if the interface is challenged, it responds to the challenge. If you do not set the passive option, the interface always challenges its peer.

For more information about CHAP, see the *JUNOS Network Interfaces Configuration Guide*.

To configure CHAP on the ATM-over-ADSL or ATM-over-SHDSL interface:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 43 on page 147.
3. If you are finished configuring the J-series device, commit the configuration.
4. To check the configuration, see "Verifying DSL Interface Configuration" on page 147.

Table 43: Configuring CHAP

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the Access level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Access, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit access
Define a CHAP access profile—for example, A-ppp-client—with a client named client 1 and the secret (password) my-secret.	1. Next to Profile, click **Add new entry**. 2. In the Profile name box, type A-ppp-client. 3. Next to Client, click **Add new entry**. 4. In the Name box, type client1. 5. In the Chap secret box, type my-secret. 6. Click **OK** until you return to the main Configuration page.	Enter set profile A-ppp-client client client1 chap-secret my-secret.
Navigate to the appropriate ATM interface level in the configuration hierarchy—for example, at-3/0/0 unit 0.	1. On the main Configuration page next to Interfaces, click **Configure** or **Edit**. 2. In the Interface name box, click **at-3/0/0**. 3. Under Interface unit number box, click **0**.	From the [edit] hierarchy level, enter edit interfaces at-3/0/0 unit 0
Configure CHAP on the ATM-over-ADSL or ATM-over-SHDSL interface and specify a unique profile name containing a client list and access parameters—for example, A-ppp-client.	1. Next to Ppp options, click **Configure**. 2. Next to Chap, click **Configure**. 3. In the Access profile box, type A-ppp-client.	Enter set ppp-options chap access-profile A-ppp-client
Specify a unique hostname to be used in CHAP challenge and response packets—for example, A-at-3/0/0.0.	In the Local name box, type, A-at-3/0/0.0	Enter set ppp-options chap local-name A-at-3/0/0.0.
Set the passive option to handle incoming CHAP packets only.	1. In the Passive box, click **Yes**. 2. Click **OK**.	Enter set ppp-options chap passive

Verifying DSL Interface Configuration

To verify ATM-over-ADSL or ATM-over-SHDSL, perform these tasks:

- Verifying ADSL Interface Properties on page 148
- Displaying a PPPoA Configuration for an ATM-over-ADSL Interface on page 151
- Verifying an ATM-over-SHDSL Configuration on page 152

Verifying ADSL Interface Properties

Purpose Verify that the interface properties are correct.

Action From the CLI, enter the show interfaces *interface-name* extensive command.

Sample Output

```
user@host> show interfaces at-3/0/0 extensive
Physical interface: at-3/0/0, Enabled, Physical link is Up
  Interface index: 141, SNMP ifIndex: 23, Generation: 48
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, ADSL mode, Speed: ADSL,

  Loopback: None
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 8 supported
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:c7:44:3c
  Last flapped   : 2005-05-16 05:54:41 PDT (00:41:42 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                 4520                    0 bps
   Output bytes  :                39250                    0 bps
   Input  packets:                   71                    0 pps
   Output packets:                 1309                    0 pps
  Input errors:
    Errors: 0, Drops: 0, Invalid VCs: 0, Framing errors: 0, Policed discards: 0,

    L3 incompletes: 0, L2 channel errors: 1, L2 mismatch timeouts: 0, Resource
errors: 0
  Output errors:
    Carrier transitions: 3, Errors: 0, Drops: 0, Aged packets: 0, MTU errors: 0,

    Resource errors: 0
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                   4                    4                    0

    1 expedited-fo                  0                    0                    0

    2 assured-forw                  0                    0                    0

    3 network-cont               2340                 2340                    0

  ADSL  alarms   : LOS, LOM, LOCDNI, FAR_LOF,  FAR_LOS, FAR_LOCDNI
  ADSL  defects  : LOF, LOS, LOCDNI, FAR_LOF,  FAR_LOS, FAR_LOCDNI
  ADSL media:            Seconds        Count  State
    LOF                  239206            2  OK
    LOS                  239208            1  OK
    LOM                       3            1  OK
    LOP                       0            0  OK
    LOCDI                     3            1  OK
    LOCDNI               239205            1  OK
  ADSL status:
    Modem status  : Showtime
    DSL mode      :     Auto    Annex A
    Last fail code: ATU-C not detected
  ADSL Statistics:                        ATU-R                  ATU-C
    Attenuation (dB)         :             0.5                    0.0
    Capacity used (%)        :              81                     72
    Noise margin (dB)        :             9.0                    9.5
    Output power (dBm)       :             7.5                    8.5
```

```
                                 Interleave       Fast  Interleave       Fast
    Bit rate (kbps)   :                 0       8128           0        896
    CRC               :                 0          3           0          0
    FEC               :                 0          0           0          0
    HEC               :                 0          3           0          0
    Received cells    :                 0        287
    Transmitted cells :                 0       4900
    Bit error rate    :                 0          0
  ATM status:
    HCS state:     Hunt
    LOC      :       OK
  ATM Statistics:
    Uncorrectable HCS errors: 0, Correctable HCS errors: 0, Tx cell FIFO overruns:
 0,
    Rx cell FIFO overruns: 0, Rx cell FIFO underruns: 0, Input cell count: 0,
    Output cell count: 0, Output idle cell count: 0, Output VC queue drops: 0,
    Input no buffers: 0, Input length errors: 0, Input timeouts: 0, Input invalid
 VCs: 0,
    Input bad CRCs: 0, Input OAM cell no buffers: 0
  Packet Forwarding Engine configuration:
    Destination slot: 3
    CoS transmit queue               Bandwidth               Buffer Priority
Limit
                               %           bps     %          bytes
    0 best-effort             95       7600000    95              0      low
none
    3 network-control          5        400000     5              0      low
none

  Logical interface at-3/0/0.0 (Index 66) (SNMP ifIndex 28) (Generation 23)
    Flags: Point-To-Point SNMP-Traps 16384 Encapsulation: ATM-PPP-LLC
    Traffic statistics:
     Input  bytes  :                 2432
     Output bytes  :                    0
     Input  packets:                  116
     Output packets:                    0
    Local statistics:
     Input  bytes  :                 1810
     Output bytes  :                    0
     Input  packets:                   78
     Output packets:                    0
    Transit statistics:
     Input  bytes  :                  622                    0 bps
     Output bytes  :                    0                    0 bps
     Input  packets:                   38                    0 pps
     Output packets:                    0                    0 pps
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive statistics:
    Input : 33 (last seen 00:00:03 ago)
    Output: 34 (last sent 00:00:03 ago)
  LCP state: Opened
  NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls:
Not-configured
  CHAP state: Success
    Protocol inet, MTU: 4470, Generation: 24, Route table: 0
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 155.55.5.1, Local: 155.55.5.2, Broadcast: Unspecified,
Generation: 45
    VCI 0.35
      Flags: Active, 1024
```

```
      Total down time: 0 sec, Last down: Never
      ATM per-VC transmit statistics:
       Tail queue packet drops: 0
      Traffic statistics:
       Input  bytes  :                 2432
       Output bytes  :                    0
       Input  packets:                  116
       Output packets:                    0

  Logical interface at-3/0/0.32767 (Index 69) (SNMP ifIndex 25) (Generation 21)
    Flags: Point-To-Multipoint No-Multicast SNMP-Traps 16384 Encapsulation:
ATM-VCMUX
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Local statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    VCI 0.4
      Flags: Active, 1024
      Total down time: 0 sec, Last down: Never
      ATM per-VC transmit statistics:
       Tail queue packet drops: 0
      Traffic statistics:
       Input  bytes  :                  208
       Output bytes  :                  208
       Input  packets:                    4
       Output packets:                    4
```

Meaning The output shows a summary of interface information. Verify the following information:

- The physical interface is Enabled. If the interface is shown as Disabled, do either of the following:
  - In the CLI configuration editor, delete the disable statement at the [edit interfaces *interface-name*] level of the configuration hierarchy.
  - In the J-Web configuration editor, clear the **Disable** check box on the **Interfaces > *interface-name*** page.
- The physical link is Up. A link state of Down indicates a problem with the interface module, interface port, or physical connection (link-layer errors).
- The Last Flapped time is an expected value. The Last Flapped time indicates the last time the physical interface became unavailable and then available again. Unexpected flapping indicates likely link-layer errors.
- The traffic statistics reflect expected input and output rates. Verify that the number of inbound and outbound bytes and packets matches expected

throughput for the physical interface. To clear the statistics and see only new changes, use the clear interfaces statistics *interface-name* command.

- No ADSL alarms and defects appear that can render the interface unable to pass packets. When a defect persists for a certain amount of time, it is promoted to an alarm. The following are ADSL-specific alarms:
  - LOCDI–Loss of cell delineation for interleaved channel
  - LOCDNI—Loss of cell delineation for non-interleaved channel
  - LOF—Loss of frame
  - LOM—Loss of multiframe
  - LOP—Loss of power
  - LOS—Loss of signal
  - FAR_LOF—Loss of frame in ADSL transceiver unit-central office (ATU-C)
  - FAR_LOS—Loss of signal in ATU-C
  - FAR_LOCDI—Loss of cell delineation for interleaved channel in ATU-C
  - FAR_LOCDNI—Loss of cell delineation for non-interleaved channel in ATU-C

Examine the operational statistics for an ADSL interface. Statistics in the ATU-R (ADSL transceiver unit–remote) column are for the near end. Statistics in the ATU-C (ADSL transceiver unit–central office) column are for the far end.

- Attenuation (dB)—Reduction in signal strength measured in decibels.
- Capacity used (%)—Amount of ADSL usage in %.
- Noise Margin (dB)—Maximum extraneous signal allowed without causing the output to deviate from an acceptable level.
- Output Power (dBm)—Amount of power used by the ADSL interface.
- Bit Rate (kbps)—Data transfer speed on the ADSL interface.

Related Topics For a complete description of show interfaces extensive output, see the *JUNOS Interfaces Command Reference*.

Displaying a PPPoA Configuration for an ATM-over-ADSL Interface

Purpose Verify the PPPoA configuration for an ATM-over-ADSL interface.

Action From the J-Web interface, select **Configuration > View and Edit > View Configuration Text**. Alternatively, from configuration mode in the CLI, enter the show interfaces *interface-name* and the show access commands from the top level.

```
[edit]
user@host# show interfaces at-3/0/0
at-3/0/0 {
```

```
encapsulation atm-pvc;
  atm-options {
    vpi 0;
  }
  dsl-options {
    operating-mode auto;
  }
  unit 0 {
    encapsulation atm-ppp-llc;
    vci 0.100;
    ppp-options {
      chap {
        access-profile A-ppp-client;
          local-name A-at-3/0/0.0;
          passive;
        }
      }
      family inet {
        negotiate address;
      }
    }
  }
user@host# show access
profile A-ppp-client {
  client A-ppp-server chap-secret "$9$G4ikPu0ISyKP5clKv7Nik.PT3"; ## SECRET-DATA
}
```

Meaning Verify that the output shows the intended configuration of PPPoA.

Related Topics For more information about the format of a configuration file, see the *J-Web Interface User Guide* or the *JUNOS CLI User Guide*.

Verifying an ATM-over-SHDSL Configuration

Purpose Verify that the interface properties are correct.

Action From the CLI, enter the show interfaces *interface-name* extensive command.

Sample Output

```
user@host> show interfaces at-6/0/0 extensive
Physical interface: at-6/0/0, Enabled, Physical link is Up
  Interface index: 141, SNMP ifIndex: 23, Generation: 48
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, ADSL mode, Speed: ADSL,

  Loopback: None
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 8 supported
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:c7:44:3c
  Last flapped   : 2005-05-16 05:54:41 PDT (00:41:42 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                 4520                    0 bps
   Output bytes  :                39250                    0 bps
   Input  packets:                   71                    0 pps
   Output packets:                 1309                    0 pps
  Input errors:
```

```
    Errors: 0, Drops: 0, Invalid VCs: 0, Framing errors: 0, Policed discards: 0,

    L3 incompletes: 0, L2 channel errors: 1, L2 mismatch timeouts: 0, Resource
errors: 0
  Output errors:
    Carrier transitions: 3, Errors: 0, Drops: 0, Aged packets: 0, MTU errors: 0,

    Resource errors: 0
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                    4                    4                    0

    1 expedited-fo                   0                    0                    0

    2 assured-forw                   0                    0                    0

    3 network-cont                2340                 2340                    0

  SHDSL  alarms   : None
  SHDSL  defects  : None
  SHDSL media:            Seconds        Count  State
    LOSD                  239206            2  OK
    LOSW                  239208            1  OK
    ES                         3            1  OK
    SES                        0            0  OK
    UAS                        3            1  OK

  SHDSL status:
    Line termination  :STU-R
  Annex     :Annex B
  Line Mode    :2-wire
  Modem Status   :Data
  Last fail code  :0
  Framer mode   :ATM
  Dying Gasp    :Enabled
  Chipset version  :1
  Firmware version  :R3.0
  SHDSL Statistics:
   Loop Attenuation (dB)   :0.600
 Transmit power (dB)     :8.5
 Receiver gain (dB)    :21.420
 SNR sampling (dB)     :39.3690
 Bit rate  (kbps)     :2304
 Bit error rate      :0
 CRC errors       :0
 SEGA errors       :1
 LOSW errors       :0
 Received cells      :1155429
 Transmitted cells     :1891375
 HEC errors       :0
 Cell drop       :0
```

Meaning The output shows a summary of interface information. Verify the following information:

- The physical interface is Enabled. If the interface is shown as Disabled, do either of the following:
  - In the CLI configuration editor, delete the disable statement at the [edit interfaces *interface-name*] level of the configuration hierarchy.

  - In the J-Web configuration editor, clear the **Disable** check box on the **Interfaces > *interface-name*** page.

- The physical link is Up. A link state of Down indicates a problem with the interface module, interface port, or physical connection (link-layer errors).
- The Last Flapped time is an expected value. The Last Flapped time indicates the last time the physical interface became unavailable and then available again. Unexpected flapping indicates likely link-layer errors.
- The traffic statistics reflect expected input and output rates. Verify that the number of inbound and outbound bytes and packets matches expected throughput for the physical interface. To clear the statistics and see only new changes, use the clear interfaces statistics *interface-name* command.
- No SHDSL alarms and defects appear that can render the interface unable to pass packets. When a defect persists for a certain amount of time, it is promoted to an alarm.
  - LOS—Loss of signal. No signal was detected on the line.
  - LOSW—Loss of sync word. A message ID was sent.
  - Power status—A power failure has occurred.
  - LOSD—Loss of signal was detected at the remote application interface.
  - ES—Errored seconds. One or more cyclic redundancy check (CRC) anomalies were detected.
  - SES—Severely errored seconds. At least 50 CRC anomalies were detected.
  - UAS—Unavailable seconds. An interval has occurred during which one or more LOSW defects were detected.

Examine the SHDSL interface status:

- Line termination—SHDSL transceiver unit–remote (STU–R). (Only customer premises equipment is supported.)
- Annex—Either Annex A or Annex B. Annex A is supported in North America, and Annex B is supported in Europe.
- Line Mode—SHDSL mode configured on the G.SHDSL interface pair, either 2-wire or 4-wire.
- Modem Status—Data. Sending or receiving data.
- Last fail code—Code for the last interface failure.
- Framer mode—Framer mode of the underlying interface: ATM.
- Dying Gasp—Ability of a J-series device that has lost power to send a message informing the attached DSL access multiplexer (DSLAM) that it is about to go offline.
- Chipset version—Version number of the chipset on the interface
- Firmware version—Version number of the firmware on the interface.

Examine the operational statistics for a SHDSL interface.

- Loop Attenuation (dB)—Reduction in signal strength measured in decibels.
- Transmit power (dB)—Amount of SHDSL usage in %.
- Receiver gain (dB)—Maximum extraneous signal allowed without causing the output to deviate from an acceptable level.
- SNR sampling (dB)—Signal-to-noise ratio at a receiver point, in decibels.
- Bit Rate (kbps)—Data transfer speed on the SHDSL interface.
- CRC errors—Number of cyclic redundancy check errors.
- SEGA errors—Number of segment anomaly errors. A regenerator operating on a segment received corrupted data.
- LOSW errors—Number of loss of signal defect errors. Three or more consecutively received frames contained one or more errors in the framing bits.
- Received cells—Number of cells received through the interface.
- Transmitted cells—Number of cells sent through the interface.
- HEC errors—Number of header error checksum errors.
- Cell drop—Number of dropped cells on the interface.

Related Topics For a complete description of show interfaces extensive output, see the *JUNOS Interfaces Command Reference*.

Chapter 7

Configuring Point-to-Point Protocol over Ethernet

The Point-to-Point Protocol over Ethernet (PPPoE) connects multiple hosts on an Ethernet LAN to a remote site through a single customer premises equipment (CPE) device—a J-series device. Hosts share a common digital subscriber line (DSL), a cable modem, or a wireless connection to the Internet. To use PPPoE, you must initiate a PPPoE session, encapsulate Point-to-Point Protocol (PPP) packets over Ethernet, and configure the device as a PPPoE client.

NOTE: J-series devices with asymmetric digital subscriber line (ADSL) or symmetric high-speed DSL (SHDSL) interfaces can use PPPoE over Asynchronous Transfer Mode (ATM) to connect through DSL lines only, not for direct ATM connections.

You can use the J-Web Quick Configuration, J-Web configuration editor, or CLI configuration editor to configure PPPoE.

This chapter contains the following topics:

PPPoE Terms

Before configuring PPPoE, become familiar with the terms defined in Table 44 on page 157.

Table 44: PPPoE Terms

Term	Definition
access concentrator	Device that acts as a server in a PPPoE session—for example, an E-series device.

Table 44: PPPoE Terms *(continued)*

Term	Definition
customer premises equipment (CPE)	Device that acts as a PPPoE client in a PPPoE session—for example, a J-series device.
Logical Link Control (LLC)	Encapsulation protocol that allows transport of multiple protocols over a single ATM virtual connection.
Point-to-Point Protocol (PPP)	Encapsulation protocol for transporting IP traffic over point-to-point links.
PPP over Ethernet (PPPoE)	Network protocol that encapsulates PPP frames in Ethernet frames and connects multiple hosts over a simple bridging access device to a remote access concentrator.
PPPoE Active Discovery Initiation (PADI) packet	Initiation packet that is broadcast by the client to start the discovery process.
PPPoE Active Discovery Offer (PADO) packet	Offer packets sent to the client by one or more access concentrators in reply to a PADI packet.
PPPoE Active Discovery Request (PADR) packet	Packet sent by the client to one selected access concentrator to request a session.
PPPoE Active Discovery Session-Confirmation (PADS) packet	Packet sent by the selected access concentrator to confirm the session.
PPPoE Active Discovery Termination (PADT) packet	Packet sent by either the client or the access concentrator to terminate a session.
PPPoE over ATM	Network protocol that encapsulates PPPoE frames in Asynchronous Transfer Mode (ATM) frames for asymmetric digital subscriber line (ADSL) or symmetric high-speed DSL (SHDSL) transmission, and connects multiple hosts over a simple bridging access device to a remote access concentrator.
virtual path identifier (VPI)	An identifier of the virtual path that establishes a route between two devices in a network.
virtual channel identifier (VCI)	An identifier of the virtual channel inside a virtual path. Each virtual path identifier (VPI) can contain multiple virtual channels, each represented by a VCI.

PPPoE Overview

On the J-series device, PPPoE establishes a point-to-point connection between the client (J-series device) and the server, also called an access concentrator. Multiple hosts can be connected to the device, and their data can be authenticated, encrypted, and compressed before the traffic is sent to the PPPoE session on the J-series device's Fast Ethernet, Gigabit Ethernet, ATM-over-ADSL, or ATM-over-SHDSL interface. PPPoE is easy to configure and allows services to be managed on a per user basis rather than on a per site basis.

This overview contains the following topics:

PPPoE Interfaces

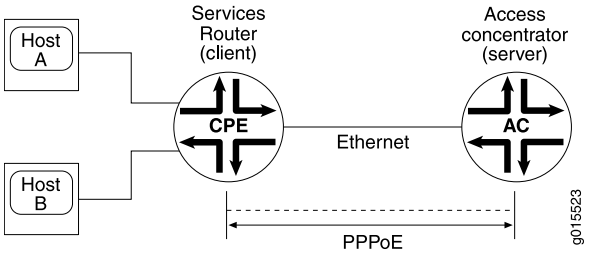
The device's PPPoE interface to the access concentrator can be a Fast Ethernet interface, a Gigabit Ethernet interface, an ATM-over-ADSL interface, or an ATM-over-SHDSL interface. The PPPoE configuration is the same for both interfaces. The only difference is the encapsulation for the underlying interface to the access concentrator:

- If the interface is Ethernet, use a PPPoE encapsulation.
- If the interface is ATM-over-ADSL or ATM-over-SHDSL, use a PPPoE over ATM encapsulation.

Ethernet Interface

The device encapsulates each PPP frame in an Ethernet frame and transports the frames over an Ethernet loop. Figure 24 on page 159 shows a typical PPPoE session between a device and an access concentrator on the Ethernet loop.

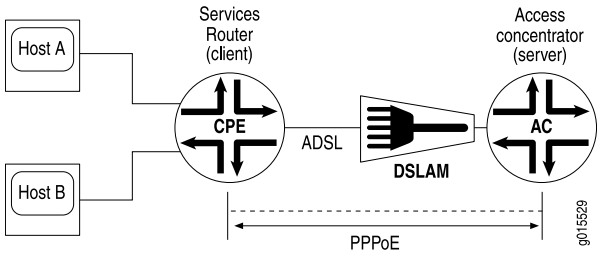
Figure 24: PPPoE Session on the Ethernet Loop



ATM-over-ADSL or ATM-over-SHDSL Interface

When an ATM network is configured with a point-to-point connection, PPPoE can use ATM Adaptation Layer 5 (AAL5) for framing PPPoE-encapsulated packets. The AAL5 protocol provides a virtual connection between the client and the server within the same network. The device encapsulates each PPPoE frame in an ATM frame and transports each frame over an ADSL or SHDSL loop and a digital subscriber line access multiplexer (DSLAM). For example, Figure 25 on page 160 shows a typical PPPoE over ATM session between a device and an access concentrator on an ADSL loop.

Figure 25: PPPoE Session on an ADSL Loop



PPPoE Stages

PPPoE has two stages, the discovery stage and the PPPoE session stage. In the discovery stage, the client discovers the access concentrator by identifying the Ethernet media access control (MAC) address of the access concentrator and establishing a PPPoE session ID. In the PPPoE session stage, the client and the access concentrator build a point-to-point connection over Ethernet, based on the information collected in the discovery stage.

PPPoE Discovery Stage

A device initiates the PPPoE discovery stage by broadcasting a PPPoE Active Discovery Initiation (PADI) packet. To provide a point-to-point connection over Ethernet, each PPPoE session must learn the Ethernet MAC address of the access concentrator and establish a session with a unique session ID. Because the network might have more than one access concentrator, the discovery stage allows the client to communicate with all of them and select one.

NOTE: A device cannot receive PPPoE packets from two different access concentrators on the same physical interface.

PPPoE Session Stage

The PPPoE session stage starts after the PPPoE discovery stage is over. The access concentrator can start the PPPoE session after it sends a PPPoE Active Discovery Session-Confirmation (PADS) packet to the client, or the client can start the PPPoE session after it receives a PADS packet from the access concentrator. A device supports multiple PPPoE sessions on each interface, but no more than 256 PPPoE sessions per device.

Each PPPoE session is uniquely identified by the Ethernet address of the peer and the session ID.

Optional CHAP Authentication

For interfaces with PPPoE encapsulation, you can configure interfaces to support the PPP Challenge Handshake Authentication Protocol (CHAP). When you enable CHAP

on an interface, the interface can authenticate its peer and be authenticated by its peer.

If you set the passive option to handle incoming CHAP packets only, the interface does not challenge its peer. However, if the interface is challenged, it responds to the challenge. If you do not set the passive option, the interface always challenges its peer.

You can configure Remote Authentication Dial-In User Service (RADIUS) authentication of PPP sessions using CHAP. CHAP enables you to send RADIUS messages through a routing instance to customer RADIUS servers in a private network. For more information, see the *JUNOS System Basics Configuration Guide*.

For more information about CHAP, see the *JUNOS Network Interfaces Configuration Guide*.

Before You Begin

Before you begin configuring PPPoE, complete the following tasks:

- Establish basic connectivity. See the Getting Started Guide for your device.
- Configure network interfaces. See "Configuring a Fast Ethernet Interface with Quick Configuration" on page 82, "Configuring Gigabit Ethernet Interfaces—Quick Configuration" on page 86, or "Configuring Digital Subscriber Line Interfaces" on page 125.

Configuring PPPoE Interfaces with Quick Configuration

To configure properties on a PPPoE interface:

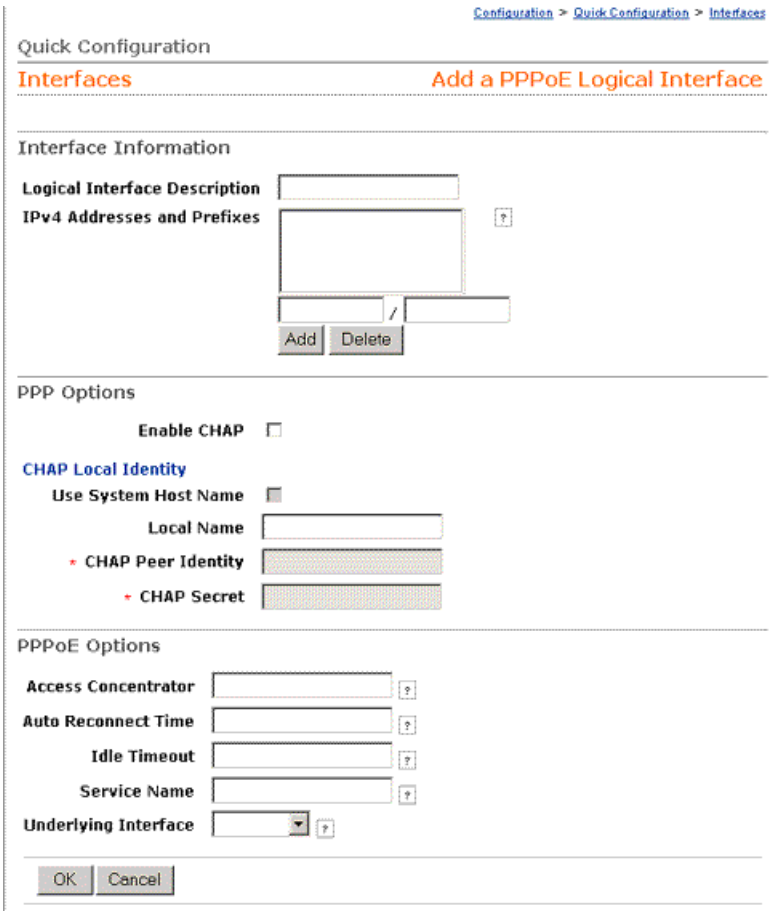
1. In the J-Web user interface, select **Configuration > Quick Configuration > Interfaces**.

   A list of the network interfaces present on the device is displayed, as shown in Figure 12 on page 74. (see "Network Interface Naming" on page 16.) The third column indicates whether the interface has been configured.

2. Select **pp0**.

   The PPPoE Interfaces Quick Configuration main page is displayed, as shown in Figure 26 on page 162.

Figure 26: PPPoE Interfaces Quick Configuration Main Page



3. Enter information into the Quick Configuration pages, as described in Table 45 on page 163.
4. From the PPPoE Interfaces Quick Configuration main page, click one of the following buttons:
   - To apply the configuration and stay on the PPPoE Quick Configuration main page, click **Apply**.
   - To apply the configuration and return to the Interfaces Quick Configuration page, click **OK**.
   - To cancel your entries and return to the Interfaces Quick Configuration page, click **Cancel**.
5. To verify that the PPPoE interface is configured correctly, see "Verifying a PPPoE Configuration" on page 171.

Table 45: PPPoE Quick Configuration Summary

Field	Function	Your Action
Logical Interfaces		
Logical Interfaces	Lists the logical interfaces for the PPPoE physical interface.	■ To add a logical interface, click **Add**. ■ To edit a logical interface, select the interface from the list. ■ To delete a logical interface, select the check box next to the name and click **Delete**.
Add logical interfaces	Defines one or more logical units that you connect to this physical PPPoE interface. You must define at least one logical unit for a PPPoE interface.	Click **Add**.
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to more clearly identify it in monitoring displays.
IPv4 Addresses and Prefixes	Specifies one or more IPv4 addresses for the interface.	1. Type one or more IPv4 addresses and prefixes. For example: 10.10.10.10/24 2. Click **Add**. 3. Click **OK**.
Physical Interface Description	(Optional) Adds supplementary information about the physical PPPoE interface.	Type a text description of the PPPoE interface to more clearly identify it in monitoring displays.
PPP Options		
Enable CHAP	Enables or disables CHAP authentication on a PPPoE interface.	■ To enable CHAP, select the check box. ■ To disable CHAP, clear the check box.
CHAP Local Identity (available if CHAP is enabled)		
Use System Host Name	Specifies that the PPPoE interface uses the device's system hostname in CHAP challenge and response packets.	■ To enable, select the check box (the default). ■ To disable, clear the check box.
Local Name	If Use System Host Name is disabled, specifies the local name for CHAP to use.	Type a local name for this PPPoE interface.
CHAP Peer Identity (required if CHAP is enabled)	Identifies the client or peer with which the device communicates on this PPPoE interface.	Type the CHAP client name.
CHAP Secret (required if CHAP is enabled)	Specifies the secret password for CHAP authentication, known to both sides of the connection.	Type a password that is known to the other side of the connection. Use a combination of letters and numbers that is difficult for others to guess.

Table 45: PPPoE Quick Configuration Summary *(continued)*

Field	Function	Your Action
PPPoE Options		
Access Concentrator	Identifies the access concentrator by a unique name.	Type a name for the access concentrator—for example, ispl.com.
Auto Reconnect Time	Specifies the number of seconds to wait before reconnecting after a PPPoE session is terminated.	Type a value from 1 through 4294947295 for automatic reconnection—for example, 100 seconds. Type 0 (the default) for immediate reconnection.
Idle Timeout	Specifies the number of seconds a session can be idle without disconnecting.	Type a value for the timeout. Type 0 if you do not want the session to time out.
Service Name	Identifies the type of service provided by the access concentrator, such as the name of the Internet service provider (ISP), class, or quality of service.	Type the type of service provided by the access concentrator. For example, video@ispl.com.
Underlying Interface	Specifies the logical Ethernet interface or the logical ATM interface as the underlying interface for the PPPoE session.	From the list, select the underlying interface for the PPPoE session—for example, ge-0/0/1.0 or at-2/0/0.0. see "Network Interface Naming" on page 16.

Configuring PPPoE with a Configuration Editor

To configure PPPoE on a device, you must perform the following tasks marked *(Required)*:

- Setting the Appropriate Encapsulation on the Interface (Required) on page 164
- Configuring PPPoE Interfaces (Required) on page 167
- Configuring CHAP on a PPPoE Interface (Optional) on page 170

Setting the Appropriate Encapsulation on the Interface (Required)

For PPPoE on an Ethernet interface, you must configure encapsulation on the logical interface. To configure encapsulation on an Ethernet logical interface, use PPP over Ethernet encapsulation.

For PPPoE on an ATM-over-ADSL or ATM-over-SHDSL interface, you must configure encapsulation on both the physical and logical interfaces. To configure encapsulation on an ATM-over-ADSL or ATM-over-SHDSL physical interface, use Ethernet over ATM encapsulation. To configure encapsulation on an ATM-over-ADSL or ATM-over-SHDSL logical interface, use PPPoE over AAL5 logical link control (LLC) encapsulation. LLC encapsulation allows a single ATM virtual connection to transport multiple protocols.

When you configure a point-to-point encapsulation such as PPP on a physical interface, the physical interface can have only one logical interface (only one unit statement) associated with it.

Perform the task appropriate for the interface on which you are using PPPoE:

- Configuring PPPoE Encapsulation on an Ethernet Interface on page 165
- Configuring PPPoE Encapsulation on an ATM-over-ADSL or ATM-over-SHDSL Interface on page 166

Configuring PPPoE Encapsulation on an Ethernet Interface

Both the client and the server must be configured to support PPPoE.

To configure PPPoE encapsulation on an Ethernet interface:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 46 on page 165.
3. If you are finished configuring the device, commit the configuration.
4. Go on to one of the following procedures:
   - To configure the PPPoE interface, see "Configuring PPPoE Interfaces (Required)" on page 167.
   - To enable authentication on the interface, see "Configuring CHAP on a PPPoE Interface (Optional)" on page 170.
   - To check the configuration, see "Verifying a PPPoE Configuration" on page 171.

Table 46: Configuring PPPoE Encapsulation on an Ethernet Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces
Configure encapsulation on a logical Ethernet interface—for example, ge-0/0/1.0. For information about interface names, see "Network Interface Naming" on page 16.	1. In the Interface name box, click **ge-0/0/1**. 2. In the Interface unit number box, click **0**. 3. From the Encapsulation list, select **ppp-over-ether**. 4. Click **OK**.	Set PPP encapsulation on unit 0 of the Ethernet interface: set ge-0/0/1 unit 0 encapsulation ppp-over-ether

Configuring PPPoE Encapsulation on an ATM-over-ADSL or ATM-over-SHDSL Interface

To configure PPPoE encapsulation on an ATM-over-ADSL or ATM-over-SHDSL interface:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 47 on page 166.
3. If you are finished configuring the device, commit the configuration.
4. Go on to one of the following procedures:
   - To configure the PPPoE interface, see "Configuring PPPoE Interfaces (Required)" on page 167.
   - To enable authentication on the interface, see "Configuring CHAP on a PPPoE Interface (Optional)" on page 170.
   - To check the configuration, see "Verifying a PPPoE Configuration" on page 171.

Table 47: Configuring PPPoE Encapsulation on an ATM-over-ADSL or ATM-over-SHDSL Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces
Navigate to the ATM-over-ADSL or ATM-over-SHDSL interface—for example, at-2/0/0—and set the ATM virtual path identifier (VPI) to 0.	1. In the Interface name box, click **at-2/0/0**. 2. Next to ATM options, click **Configure**. 3. Next to Vpi, click **Add new entry**. 4. In the Vpi number box, type 0. 5. Click **OK** twice.	Enter set at-2/0/0 atm-options vpi 0

Table 47: Configuring PPPoE Encapsulation on an ATM-over-ADSL or ATM-over-SHDSL Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Do one of the following: ■ Configure the ADSL operating mode on the physical ATM interface—for example, autonegotiation. ■ Configure the SHDSL options: ■ Annex type—for example, Annex A. ■ SHDSL line rate for SHDSL interface—for example, automatic selection of line rate. ■ Loopback option for testing the SHDSL connection integrity on the physical ATM interface—for example, local.	To configure the ADSL operating mode on the physical ATM interface: 1. Next to Dsl options, click **Configure**. 2. From the Operating mode list, select **auto**. 3. Click **OK**.	Enter set at-2/0/0 dsl-options operating-mode auto
	To configure the SHDSL options: 1. Next to Shdsl options, click **Configure**. 2. From the Annex list, select **Annex-a**. 3. From the Line Rate list, select **auto**. 4. From the Loopback list, select **local**. 5. Click **OK** until you return to the Interfaces page.	Enter set at-2/0/0 shdsl-options annex annex-a line-rate auto loopback local
Configure Ethernet over ATM encapsulation on the physical ATM-over-ADSL or ATM-over-SHDSL interface.	From the Encapsulation list, select **ethernet-over-atm**.	Enter set at-2/0/0 encapsulation ethernet-over-atm
Create an ATM-over-ADSL or ATM-over-SHDSL logical interface, configure LLC encapsulation, and specify a VCI number.	1. Next to Unit, click **Add new entry**. 2. In the Interface unit number box, type 0. 3. From the Encapsulation list, select **ppp-over-ether-over-atm-llc**. 4. In the Multicast vci box, type 0.120 and click **OK**.	Enter set at-2/0/0 unit 0 encapsulation ppp-over-ether-over-atm-llc vci 0.120

Configuring PPPoE Interfaces (Required)

To create and configure a PPPoE interface over the underlying Ethernet and ATM interface:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 48 on page 168.
3. If you are finished configuring the device, commit the configuration.
4. Go on to one of the following procedures:
   - To enable authentication on the PPPoE interface, see "Configuring CHAP on a PPPoE Interface (Optional)" on page 170.
   - To check the configuration, see "Verifying a PPPoE Configuration" on page 171.

Table 48: Configuring a PPPoE Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces
Create a PPPoE interface with a logical interface unit 0.	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type pp0 and click **OK**. 3. Under Interface name, click **pp0**. 4. Next to Unit, click **Add new entry**. 5. In the Interface unit number box, type 0.	Enter edit pp0 unit 0
Configure an ISDN interface as the backup interface for the PPPoE interface—for example, dl0.0.	1. Next to Backup options, click **Configure**. 2. In the Interface box, type dl0.0. 3. Click OK.	Enter set backup-options interface dl0.0
Specify the logical Ethernet interface or the logical ATM interface as the underlying interface for the PPPoE session—for example, ge-0/0/1.0 or at-2/0/0.0.	1. Next to Pppoe options, click **Edit**. 2. In the Underlying Interface box, type one of the following interface names: ■ For a logical Ethernet interface, type ge-0/0/1.0. ■ For a logical ATM interface type, at-2/0/0.0.	Enter one of the following commands: ■ set pppoe-options underlying-interface ge-0/0/1.0. ■ set pppoe-options underlying-interface at-2/0/0.0.
Identify the access concentrator by a unique name—for example, ispl.com.	In the Access concentrator box type ispl.com.	Enter set pppoe-options access-concentrator ispl.com
Specify the number of seconds (from 1 through 4294967295) to wait before reconnecting after a PPPoE session is terminated—for example, 100. A 0 value (the default) specifies immediate reconnection.	In the Auto reconnect box, type 100.	Enter set pppoe-options auto-reconnect 100
Specify the number of seconds a session can be idle—for example, 100. A 0 value prevents the session from timing out.	In the Idle timeout box, type 100.	Enter set pppoe-options idle-timeout 100.
Specify the J-Series device as the client for the PPPoE interface.	In the Client box, click **Yes**.	Enter set pppoe-options client.

Table 48: Configuring a PPPoE Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Identify the type of service provided by the access concentrator, such as the name of the Internet service provider (ISP), class, or quality of service—for example, video@ispl.com.	1. In the Service name box, type video@ispl.com. 2. Click **OK**.	Enter set pppoe-options service-name video@ispl.com
Configure the maximum transmission unit (MTU) of the IPv4, IPv6, or Multiprotocol Label Switching (MPLS) protocol families—for example, 1492.	1. Select one of the following protocol families: ■ For the IPv4 family, in the Inet box, select **Yes** and click **Configure**. ■ For the IPv6 family, in the Inet6 box, select **Yes** and click **Configure**. ■ For the MPLS family, in the Mpls box, select **Yes** and click **Configure**. 2. In the Mtu box, type 1492. 3. Click **OK** until you return to the Unit page.	Enter one of the following: ■ set family inet mtu 1492 ■ set family inet6 mtu 1492 ■ set family mpls mtu 1492
Configure the PPPoE logical interface address in one of the following ways: Do one of the following: ■ Assign IPv4 or IPv6 source and destination addresses—for example: ■ 192.168.1.1/32 and 192.168.1.2 for IPv4 ■ 2004:1/128 and 2004:2 for IPv6. ■ Derive the IPv4 source address from a specified interface—for example, the loopback interface, lo0.0—and assign a destination address—for example, 192.168.1.2. The specified interface must include a logical unit number and have a configured IP address. ■ Obtain an IP address by negotiation with the remote end. This method might require the access concentrator to use a RADIUS authentication server.	Select one of the following IP address configurations: To assign the source and destination addresses: 1. Next to Inet, click **Edit**. 2. Next to Address, click **Add new entry**. 3. In the Inet Source box, type 192.168.1.1/32, or in the Inet6 Source box, type 2004::1/128. 4. In the Inet Destination box, type 192.168.1.2, or in the Inet6 Destination box, type 2004::2. 5. Click **OK** until you return to the Unit page. To derive the IPv4 source address and assign the destination address: 1. Next to Inet, click **Edit**. 2. Next to Unnumbered address, select the **Yes** check box and click **Configure**. 3. In the Destination box, type 192.168.1.2. 4. In the Source box, type lo0.0. 5. Click **OK** until you return to the Unit page. To obtain an IP address from the remote end: 1. Next to Inet, click **Edit**. 2. Next to Negotiate address, select the **Yes** check box. 3. Click **OK** until you return to the Unit page.	Do one of the following: ■ To assign source and destination addresses enter one of the following sets of commands: ■ For IPv4 addresses, set family inet address 192.168.1.1/32 destination 192.168.1.2. ■ For IPv6 addresses, set family inet6 address 2004::1/128 destination 2004::2. ■ To derive the IPv4 source address and assign the destination address, enter set family inet unnumbered-address lo0.0 destination 192.168.1.2. ■ To obtain an IP address from the remote end, enter set family inet negotiate-address.

Table 48: Configuring a PPPoE Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Disable the sending of keepalives on a logical interface.	1. From the Keepalive choices list, select **no keepalives**. 2. Click **OK** to apply your entries to the configuration.	Enter set no-keepalives

To clear a PPPoE session on the pp0.0 interface, enter the clear pppoe sessions pp0.0 command. To clear all sessions on the PPPoE inteface, enter the clear pppoe sessions command.

Configuring CHAP on a PPPoE Interface (Optional)

To configure CHAP on the PPPoE interface:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 49 on page 170.
3. If you are finished configuring the J-series device, commit the configuration.
4. To check the configuration, see "Verifying a PPPoE Configuration" on page 171.

Table 49: Configuring CHAP on a PPPoE Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Profile** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Access, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter set access profile A-ppp-client client client1 chap-secret my-secret
Define a CHAP access profile—for example, A-ppp-client—with a client named client 1 and the secret (password) my-secret.	1. Next to Profile, click **Add new entry**. 2. In the Profile name box, type A-ppp-client. 3. Next to Client, click **Add new entry**. 4. In the Name box, type client1. 5. In the Chap secret box, type my-secret. 6. Click **OK** until you return to the main Configuration page.	
Navigate to the pp0 unit 0 interface level in the configuration hierarchy.	1. On the main Configuration page next to Interfaces, click **Configure** or **Edit**. 2. In the Interface name box, click **pp0**. 3. In the Interface unit number box, click **0**.	From the [edit] hierarchy level, enter edit interfaces pp0 unit 0

Table 49: Configuring CHAP on a PPPoE Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure CHAP on the PPPoE interface, and specify a unique profile name containing a client list and access parameters—for example, A-ppp-client.	1. Next to Ppp options, click **Configure**. 2. Next to Chap, click **Configure**. 3. In the Access profile box, type A-ppp-client.	Enter set ppp-options chap access-profile A-ppp-client
Specify a unique hostname to be used in CHAP challenge and response packets—for example, A-ge-0/0/1.0 or A-at-2/0/0.0.	In the Local name box, type one of the following: ■ For an Ethernet interface, type A-ge-0/0/1.0. ■ For an ATM interface, type A-at-2/0/0.0.	Do one of the following: ■ For the Ethernet interface, enter set ppp-options chap local-name A-ge-0/0/1.0. ■ For the ATM interface, enter set ppp-options chap local-name A-at-2/0/0.0.
Set the passive option to handle incoming CHAP packets only.	1. In the Passive box, click **Yes**. 2. Click **OK**.	Enter set ppp-options chap passive

Verifying a PPPoE Configuration

To verify PPPoE configuration perform the following tasks:

Displaying a PPPoE Configuration for an Ethernet Interface

Purpose Verify the PPPoE configuration for an Ethernet interface.

Action From the J-Web interface, select **Configuration > View and Edit > View Configuration Text**. Alternatively, from configuration mode in the CLI, enter the show interfaces command from the top level.

```
[edit]
user@host#show interfaces
ge-3/0/0 {
    unit 1 {
    }
}
pp0 {
    unit 1 {
```

```
    pppoe-options {
      underlying-interface ge-3/0/0.0;
        idle-timeout 123;
        access-concentrator myac;
        service-name myserv;
        auto-reconnect 10;
        client;
      }
      family inet {
        address 22.2.2.1/32 {
          destination 22.2.2.2;
        }
      }
      family inet6 {
        address 3004::1/128 {
          destination 3004::2;
        }
      }
    }
  }
```

Meaning Verify that the output shows the intended configuration of PPPoE.

Related Topics For more information about the format of a configuration file, see the *J-Web Interface User Guide* or the *JUNOS CLI User Guide*.

Displaying a PPPoE Configuration for an ATM-over-ADSL or ATM-over-SHDSL Interface

Purpose Verify the PPPoE configuration for an ATM-over-ADSL or ATM-over-SHDSL interface.

Action From the J-Web interface, select **Configuration > View and Edit > View Configuration Text**. Alternatively, from configuration mode in the CLI, enter the show interfaces command from the top level.

```
[edit]
user@host#show interfaces
at-6/0/0 {
  encapsulation ethernet-over-atm;
  atm-options {
    vpi 0;
  }
  dsl-options {
    operating-mode itu-dmt;
  }
  unit 0 {
    encapsulation ppp-over-ether-over-atm-llc;
    vci 35;
  }
}
pp0 {
  unit 0 {
    pppoe-options {
      underlying-interface at-6/0/0.0;
```

```
            idle-timeout 123;
            access-concentrator myac;
            service-name myserv;
            auto-reconnect 10;
            client;
          }
          family inet {
            address 11.1.1.1/32 {
              destination 11.1.1.2;
            }
          }
          family inet6 {
            address 2004::1/128 {
              destination 2004::2;
            }
          }
          family mpls;
        }
      }
```

Meaning Verify that the output shows the intended configuration of PPPoE.

Related Topics For more information about the format of a configuration file, see the *J-Web Interface User Guide* or the *JUNOS CLI User Guide*.

Verifying PPPoE Interfaces

Purpose Verify that the PPPoE device interfaces are configured properly.

Action From the CLI, enter the show interfaces pp0 command.

Sample Output

```
user@host> show interfaces pp0
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 67, SNMP ifIndex: 317
  Type: PPPoE, Link-level type: PPPoE, MTU: 9192
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
  Last flapped   : Never
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface pp0.0 (Index 1) (SNMP ifIndex 330)
    Flags: Point-To-Point SNMP-Traps 16384 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 3304,
      Session AC name: isp1.com, AC MAC address: 00:90:1a:40:f6:4c,
      Service name: video@isp1.com, Configured AC name: isp1.com,
      Auto-reconnect timeout: 60 seconds
      Underlying interface: ge-5/0/0.0 (Index 71)
  Input packets : 23
  Output packets: 22
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 16 (00:00:26 ago), Output: 0 (never)
  LCP state: Opened
  NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls:
```

```
Not-configured
CHAP state: Success
  Protocol inet, MTU: 1492
    Flags: Negotiate-Address
    Addresses, Flags: Kernel Is-Preferred Is-Primary
      Destination: 211.211.211.2, Local: 211.211.211.1
```

Meaning The output shows information about the physical and the logical interface. Verify the following information:

- The physical interface is enabled and the link is up.
- The PPPoE session is running on the correct logical interface.
- Under State, the state is active (up).
- Under Underlying interface, the physical interface on which the PPPoE session is running is correct:
  - For an Ethernet connection, the underlying interface is Fast Ethernet or Gigabit Ethernet—for example, ge-5/0/0.0.
  - For an ATM-over-ADSL or ATM-over-SHDSL connection, the underlying interface is ATM—for example, at-2/0/0.0.

Related Topics For a complete description of show interfaces pp0 output, see the *JUNOS Interfaces Command Reference*.

Verifying PPPoE Sessions

Purpose Verify that a PPPoE session is running properly on the logical interface.

Action From the CLI, enter the show pppoe interfaces command.

Sample Output

```
user@host> show pppoe interfaces
pp0.0 Index 67
  State: Session up, Session ID: 31,
  Service name: video@isp1.com, Configured AC name: isp1.com,
  Session AC name: belur, AC MAC address: 00:90:1a:40:f6:4e,
  Auto-reconnect timeout: 1 seconds,
  Underlying interface: ge-0/0/1.0 Index 69
```

Meaning The output shows information about the PPPoE sessions. Verify the following information:

- The PPPoE session is running on the correct logical interface.
- Under State, the session is active (up).
- Under Underlying interface, the physical interface on which the PPPoE session is running is correct:
  - For an Ethernet connection, the underlying interface is Fast Ethernet or Gigabit Ethernet—for example, ge-0/0/1.0.
  - For an ATM-over-ADSL or ATM-over-SHDSL connection, the underlying interface is ATM—for example, at-2/0/0.0.

Related Topics For a complete description of show pppoe interfaces output, see the *JUNOS Interfaces Command Reference*.

Verifying the PPPoE Version

Purpose Verify the version information of the PPPoE protocol configured on the device interfaces.

Action From the CLI, enter the show pppoe version command.

Sample Output

```
user@host> show pppoe version
Point-to-Point Protocol Over Ethernet, version 1. rfc2516
   PPPoE protocol               = Enabled
   Maximum Sessions             = 256
   PADI resend timeout          = 2 seconds
   PADR resend timeout          = 16 seconds
   Max resend timeout           = 64 seconds
   Max Configured AC timeout    = 4 seconds
```

Meaning The output shows PPPoE protocol information. Verify the following information:

- The correct version of the PPPoE protocol is configured on the interface.
- Under PPPoE protocol, the PPPoE protocol is enabled.

Related Topics For a complete description of show pppoe version output, see the *JUNOS Interfaces Command Reference*.

Verifying PPPoE Statistics

Purpose Display statistics information about PPPoE interfaces.

Action From the CLI, enter the show pppoe statistics command.

Sample Output

```
user@host> show pppoe statistics
Active PPPoE sessions: 4
  PacketType                      Sent         Received
    PADI                           502                0
    PADO                             0              219
    PADR                           219                0
    PADS                             0              219
    PADT                             0              161
    Service name error               0                0
    AC system error                  0               13
    Generic error                    0                0
    Malformed packets                0               41
    Unknown packets                  0                0
  Timeout
    PADI                            42
    PADO                             0
    PADR                             0
```

Meaning The output shows information about active sessions on PPPoE interfaces. Verify the following information:

- Total number of active PPPoE sessions running on the interface.

- Under Packet Type, the number of packets of each type sent and received during the PPPoE session.

Related Topics For a complete description of show pppoe statistics output, see the *JUNOS Interfaces Command Reference*.

Chapter 8

Configuring ISDN

ISDN connectivity is supported on the J-series devices as a backup for a primary Internet connection. The J-series devices can be configured to "fail over" to an ISDN interface when the primary connection experiences interruptions in Internet connectivity.

Use ISDN also at the central office to terminate calls that originate at branch office routers and for central office callback for security, accounting, or cost savings at the branch office.

You can use either J-Web Quick Configuration or a configuration editor to configure ISDN BRI interfaces. To configure ISDN PRI, you use either the J-Web configuration editor or CLI configuration editor.

NOTE: This chapter provides instructions for configuring basic ISDN BRI service and features such as dial backup, dial-in, or callback for both ISDN BRI and ISDN PRI. To configure basic ISDN PRI service, see "Configuring Channelized T1/E1/ISDN PRI Interfaces" on page 109.

This chapter contains the following topics:

ISDN Terms

Before configuring ISDN, become familiar with the terms defined in Table 50 on page 178.

Table 50: ISDN Terminology

Term	Definition
bandwidth on demand	ISDN cost-control feature defining the bandwidth threshold that must be reached on all links before a J-series device initiates additional ISDN data connections to provide more bandwidth.
Basic Rate Interface (BRI)	ISDN service intended for home and small enterprise applications. ISDN BRI consists of two 64-Kbps B-channels to carry voice or data and one 16-Kbps D-channel for control and signaling.
bearer channel (B-channel)	64-Kbps channel used for voice or data transfer on an ISDN interface.
callback	Alternative feature to dial-in that enables a J-series device to call back the caller from the remote end of a backup ISDN connection. Instead of accepting a call from the remote end of the connection, the device rejects the call, waits a configured period of time, and calls a number configured on the device's dialer interface. See also *dial-in*.
caller ID	Telephone number of the caller on the remote end of a backup ISDN connection, used to dial in and also to identify the caller. Multiple caller IDs can be configured on an ISDN dialer interface. During dial-in, the device matches the incoming call's caller ID against the caller IDs configured on its dialer interfaces. Each dialer interface accepts calls from only callers whose caller IDs are configured on it.
delta-channel (D-channel)	Circuit-switched channel that carries signaling and control for B-channels. In ISDN Basic Rate Interface (BRI) applications, a D-channel can also support customer packet data traffic at speeds up to 9.6 Kbps.
demand circuit	Network segment whose cost varies with usage, according to a service level agreement with a service provider. Demand circuits limit traffic based on either bandwidth (bites or packets transmitted) or access time. For example, ISDN interfaces can be configured for dial-on-demand routing backup. In OSPF, the demand circuit reduces the amount of OSPF traffic by removing all OSPF protocols when the routing domain is in a steady state.
dial backup	Feature that reestablishes network connectivity through one or more backup ISDN dialer interfaces after a primary interface fails. When the primary interface is reestablished, the ISDN interface is disconnected.
dialer filter	Stateless firewall filter that enables dial-on-demand routing backup when applied to a physical ISDN interface and its dialer interface configured as a passive static route. The passive static route has a lower priority than dynamic routes. If all dynamic routes to an address are lost from the routing table and the device receives a packet for that address, the dialer interface initiates an ISDN backup connection and sends the packet over it. See also *dial-on-demand routing backup; floating static route*.
dialer interface (dl)	Logical interface for configuring dialing properties and the control interface for a backup ISDN connection.

Table 50: ISDN Terminology *(continued)*

Term	Definition
dial-in	Feature that enables J-series devices to receive calls from the remote end of a backup ISDN connection. The remote end of the ISDN call might be a service provider, a corporate central location, or a customer premises equipment (CPE) branch office. All incoming calls can be verified against caller IDs configured on the device's dialer interface. See also *callback*.
dial-on-demand routing (DDR) backup	Feature that provides a J-series device with full-time connectivity across an ISDN line. When routes on a primary serial T1, E1, T3, E3, Fast Ethernet, Gigabit Ethernet, or PPPoE interface are lost, an ISDN dialer interface establishes a backup connection. To save connection time costs, the device drops the ISDN connection after a configured period of inactivity. Devices with ISDN interfaces support two types of dial-on-demand routing backup: on-demand routing with a dialer filter and dialer watch. See also *dialer filter; dialer watch*.
dialer profile	Set of characteristics configured for the ISDN dialer interface. Dialer profiles allow the configuration of physical interfaces to be separated from the logical configuration of dialer interfaces required for ISDN connectivity. This feature also allows physical and logical interfaces to be bound together dynamically on a per-connection basis.
dialer watch	Dial-on-demand routing (DDR) backup feature that provides reliable connectivity without relying on a dialer filter to activate the ISDN interface. The ISDN dialer interface monitors the existence of each route on a watch list. If all routes on the watch list are lost from the routing table, dialer watch initiates the ISDN interface for failover connectivity. See also *dial-on-demand routing backup*.
floating static route	Route with an administrative distance greater than the administrative distance of the dynamically learned versions of the same route. The static route is used only when the dynamic routes are no longer available. When a floating static route is configured on an interface with a dialer filter, the interface can be used for backup.
Integrated Services Digital Network (ISDN)	Digital communication service provided by telecommunication service providers. It is an all-digital dialup (on-demand) service that carries voice, data, and video transmissions over telephone lines.
Primary Rate Interface (PRI)	ISDN service intended for higher-bandwidth applications than ISDN BRI. ISDN PRI consists of a single D-channel for control and signaling, plus a number of 64-Kbps B-channels—either 23 B-channels on a T1 line or 30 B-channels on an E1 line—to carry network traffic.
service profile identifier (SPID)	Number that specifies the services available to you on the service provider switch and defines the feature set ordered when the ISDN service is provisioned.
terminal endpoint identifier (TEI)	Number that identifies a terminal endpoint, an ISDN-capable device attached to an ISDN network through an ISDN interface on the device. The TEI is a number between 0 and 127. The numbers 0–63 are used for static TEI assignment, 64–126 are used for dynamic assignment, and 127 is used for group assignment.

ISDN Overview

Integrated Services Digital Network (ISDN) is a set of standards for digital transmission over different media created by the Consultative Committee for International Telegraph and Telephone (CCITT) and International Telecommunication Union (ITU). As a dial-on-demand service, it has fast call setup and low latency as well as the ability to carry high-quality voice, data, and video transmissions. ISDN is also a circuit-switched service that can be used on both multipoint and point-to-point connections.

You configure two types of interfaces for ISDN service: at least one physical interface and a logical interface called the dialer interface.

ISDN Interfaces

The following interfaces on a device are available for ISDN connectivity:

- For ISDN BRI, up to six of the following field-replaceable units (FRUs):
  - 4-port S/T PIM supporting ITU-T I.430, ETSI TS 101080, and GR-1089-Core Type III
  - 4-port U PIM supporting ANSI T.601 and GR-1089-Core
- For ISDN PRI, up to six Dual-Port Channelized T1/E1/ISDN PRI PIMs, supporting ITU-T Q.920, Q.921: LAPD, Q.930, and Q.931

ISDN BRI Interface Types

A J-series device with one or more ISDN BRI ports has the following types of ISDN interfaces:

- Physical ISDN BRI interface—br-*pim*/0/*port*
- Physical B-channel interface—bc-*pim*/0/*port*
- Physical D-channel interface—dc-*pim*/0/*port*
- Logical dialer interface—dl*n*

For information about interface names, see “Network Interface Naming” on page 16.

To configure ISDN BRI service on a J-series device, you configure the physical ISDN BRI interface and the logical dialer interface.

Each ISDN BRI port has two B-channels for transport, identified as bc-*pim*/0/*port*:1 and bc-*pim*/0/*port*:2, and one D-channel for control, identified as dc-*pim*/0/*port*. On ISDN BRI interfaces, the B-channels and D-channel have no configurable settings, but you can monitor them for interface status and statistics.

ISDN PRI Interface Types

On a J-series device with one or more Dual-Port Channelized T1/E1/ISDN PRI PIMs, you can configure each port on the PIM for either T1, E1, or ISDN PRI service, or for a combination of ISDN PRI and either T1 or E1 service. For ISDN PRI service, you configure the following types of ISDN interfaces as channels on the channelized T1 or E1 interface:

- Physical B-channel interface—bc-*pim*/0/*port*:*channel*
  - On a channelized T1 interface, up to 23 time slots can be configured as ISDN PRI B-channels.
  - On a channelized E1 interface, up to 30 time slots can be configured as ISDN PRI B-channels.
- Physical D-channel interface—dc-*pim*/0/*port*:*channel*
  - On a channelized T1 interface, you configure time slot 24 as the D-channel.
  - On a channelized E1 interface, you configure time slot 16 as the D-channel.
- Logical dialer interface—dl*n*

For information about interface names, see "Network Interface Naming" on page 16.

For more information about channelized T1/E1/ISDN PRI interfaces, see "Configuring Channelized T1/E1/ISDN PRI Interfaces" on page 109.

Dialer Interface

The dialer (dl*n*) interface is a logical interface on which you configure dialing properties for ISDN connections. The interface can be configured in two modes:

- Multilink mode using Multilink PPP encapsulation
- Normal mode using PPP or Cisco High-Level Data Link Control (HDLC) encapsulation

The dialer interface can perform backup, dialer filter, and dialer watch functions, but these operations are mutually exclusive. You can configure a single dialer interface to operate in only one of the following ways:

- As a backup interface—for one primary interface
- As a dialer filter
- As a dialer watch interface

Before You Begin

Before you configure ISDN interfaces, you need to perform the following tasks:

- Install J-series device hardware. For more information, see the *J-series Services Routers Hardware Guide*.
- Establish basic connectivity. For more information, see the Getting Started Guide for your device.
- Order an ISDN line from your telecommunications service provider. Contact your service provider for more information.
- If you do not already have a basic understanding of physical and logical interfaces and Juniper Networks interface conventions, read "Interfaces Overview" on page 11.

Although it is not a requirement, you might also want to plan how you are going to use the ISDN interfaces on your network before you begin configuring them. (To display a list of installed ISDN BRI interfaces, select **Configuration > Quick Configuration > Interfaces**.)

Configuring ISDN BRI Interfaces with Quick Configuration

You can use the ISDN Interfaces Quick Configuration pages to configure ISDN BRI interfaces on a J-series device. The Quick Configuration pages allow you to configure ISDN BRI connectivity on a device to back up a primary Internet connection.

NOTE: To configure an ISDN *PRI* interface, you must use the J-Web or CLI configuration editor.

You configure the physical ISDN BRI interface first and then the backup method on the logical dialer interface.

This section contains the following topics:

Configuring ISDN BRI Physical Interfaces with Quick Configuration

To configure ISDN BRI physical interfaces with Quick Configuration:

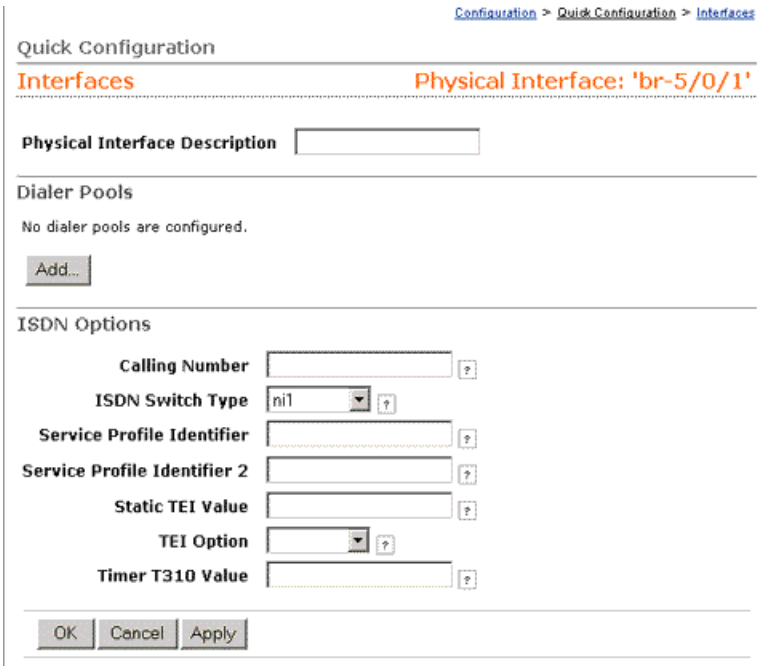
1. In the J-Web interface, select **Configuration > Quick Configuration > Interfaces**.

   A list of network interfaces installed on the device is displayed.

2. Click the br-*pim*/0/*port* interface name for the ISDN BRI port you want to configure.

   The ISDN BRI Physical Interface Quick Configuration page is displayed as shown in Figure 27 on page 183.

Figure 27: ISDN BRI Physical Interface Quick Configuration Page



3. Enter information into the ISDN Quick Configuration pages, as described in Table 51 on page 183.
4. From the ISDN Physical Interfaces Quick Configuration page:
   - To apply the configuration and stay on the ISDN Physical Interfaces Quick Configuration page, click **Apply**.
   - To apply the configuration and return to the Interfaces Quick Configuration page, click **OK**.
   - To cancel your entries and return to the Interfaces Quick Configuration page, click **Cancel**.
5. Go on to "Configuring ISDN BRI Dialer Interfaces with Quick Configuration" on page 185.

Table 51: ISDN BRI Quick Configuration Page Summary

Field	Function	Your Action
Configuring ISDN Interfaces		
Physical Interface Description	(Optional) Adds supplemental information about the ISDN physical interface on the device.	Type a text description of the physical ISDN BRI interface in the box to clearly identify it in monitoring displays.

Table 51: ISDN BRI Quick Configuration Page Summary *(continued)*

Field	Function	Your Action
Clocking	Enables internal or external clocking sources for the interface on the device. ■ **internal**—Device's own system clock (the default) ■ **external**—Clock received from the T1 interface	Select **internal** or **external** from the list.
Dialer Pool Options		
Dialer Pools	Displays the list of configured ISDN dialer pools on the device.	■ To add a dialer pool to the interface, click **Add**. ■ To edit a dialer pool, select the name from the list. You can change the priority, but not the name. ■ To delete a dialer pool, select the check box and click **Delete**.
Dialer Pool Name (required)	Specifies the group of physical interfaces to be used by the dialer interface.	Type the dialer pool name—for example, isdn-dialer-pool.
Priority	Specifies the priority of this interface within the dialer pool. Interfaces with a higher priority are the first to interact with the dialer interface.	1. Type a priority value from 0 (lowest) to 255 (highest). The default is 0. 2. Click **OK** to return to the Quick Configuration page.
ISDN Options		
Calling Number	Configures the dialing number used to connect with the service provider.	Type the outgoing calling number for the service provider.
ISDN Switch Type	Specifies the type of ISDN switch used by the service provider.	Select one of the following switch types: ■ **att5e**—AT&T 5ESS ■ **etsi**—NET3 for the UK and Europe ■ **ni1**—National ISDN-1 ■ **ntdms-100**—Northern Telecom DMS-100 ■ **ntt**—NTT Group switch for Japan
Service Profile Identifier Service Profile Identifier 2	Configures the service profile identifier (SPID) provided by your ISDN service.	Type the SPID in the box. If you have a NTDMS-100 or NI1 switch, an additional SPID field is provided.

Table 51: ISDN BRI Quick Configuration Page Summary *(continued)*

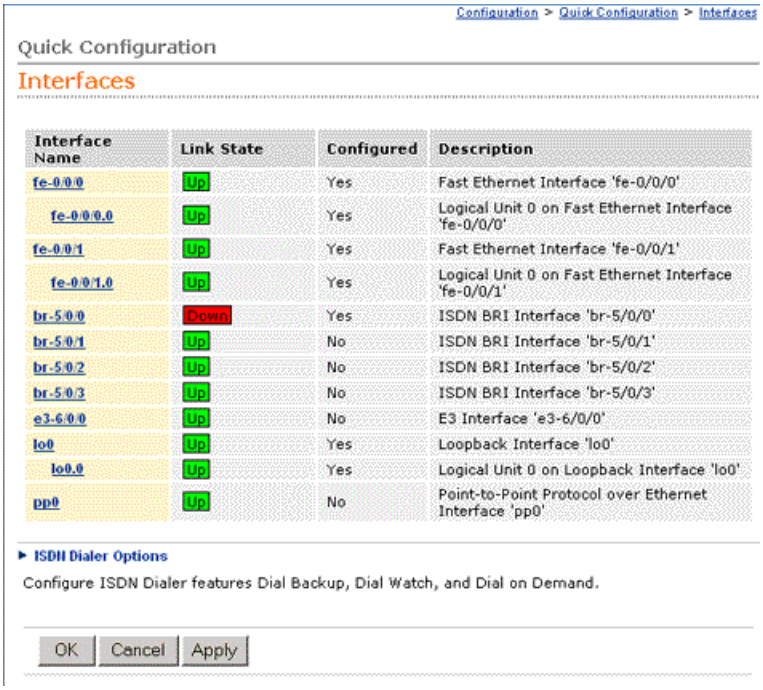
Field	Function	Your Action
Static TEI Value	Configures the static terminal endpoint identifier (TEI) value from your service provider. The TEI number identifies a terminal endpoint, an ISDN-capable device attached to an ISDN network through an ISDN interface on the device. The TEI is a number between 0 and 127. The numbers 0–63 are used for static TEI assignment, 64–126 are used for dynamic assignment, and 127 is used for group assignment.	Type a value between 0 and 63. If this value is not supplied, the device dynamically acquires a TEI. If you configured more than one SPID, the TEI must be acquired dynamically.
TEI Option	Configures when the TEI negotiates with the ISDN provider.	■ Select **first-call** to activate the connection when the call setup is sent to the ISDN provider. ■ Select **power-up** (the default) to activate the connection when the device is powered on.
Timer T310 Value	Sets the Q.931 timer value in seconds.	Type a value between 1 and 65536. The default value is 10 seconds.

Configuring ISDN BRI Dialer Interfaces with Quick Configuration

When ISDN BRI interfaces are installed on the device, links to ISDN Quick Configuration pages for dialer options are displayed on the Interfaces Quick Configuration page as shown in Figure 28 on page 186.

You can use these Quick Configuration pages to configure an ISDN BRI dialer interface for either dial backup or dialer watch. For dial backup you specify the serial interface to back up. For dialer watch you specify a watch list of one or more routes to monitor.

Figure 28: ISDN BRI Dialer Options Quick Configuration Page



To configure ISDN BRI dialer interfaces with Quick Configuration:

1. In the J-Web interface, select **Configuration > Quick Configuration > Interfaces**.

   A list of network interfaces installed on the device is displayed.

2. Click **ISDN Dialer Options** under the interfaces list.

3. Select a backup method to configure on the dialer interface:

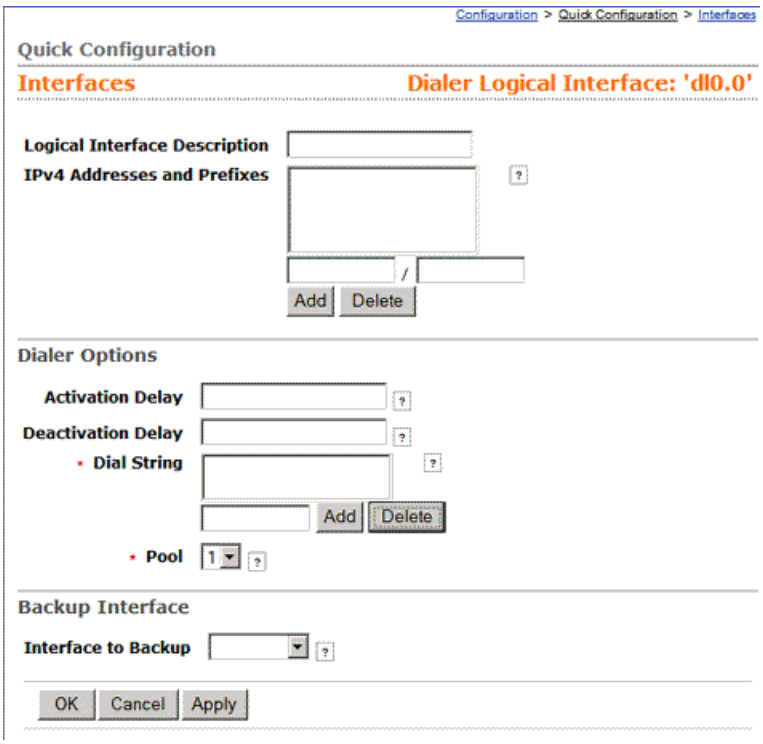
   - Click **Dial Backup** to allow one or more dialer interfaces to back up the primary interface. The backup interfaces are activated only when the primary interface fails.
   - Click **Dialer Watch** to monitor a specified route and initiate dialing of the backup link if that route is not present.

4. Do one of the following:

   - To edit an existing dialer interface, click the dialer interface name. For example, click **dl0** to edit the dialer physical interface, and then click **dl0.0** to edit the dialer logical interface.
   - To add a dialer interface, click **Add**. In the Interface Name box, type a name for the logical interface—for example, dl1—then click **Add** under Logical Interfaces.

   Figure 29 on page 187 shows the ISDN Quick Configuration page for dialer logical interfaces.

Figure 29: ISDN BRI Dialer Interface Quick Configuration Page



5. Enter information into the ISDN Quick Configuration page for dialer logical interfaces, as described in Table 52 on page 187.
6. Click one of the following buttons on the ISDN Quick Configuration page:
   - To apply the configuration and stay on the current Quick Configuration page, click **Apply**.
   - To apply the configuration and return to the previous Quick Configuration page, click **OK**.
   - To cancel your entries and return to the previous Quick Configuration page, click **Cancel**.
7. To verify that the ISDN interface is configured correctly, see "Verifying the ISDN Configuration" on page 211.

Table 52: ISDN BRI Dialer Interface Quick Configuration Page Summary

Field	Function	Your Action
Configuring Dialer Interfaces		
Logical Interface Description	Describes the logical interface.	Type a text description of the interface in the box.

Table 52: ISDN BRI Dialer Interface Quick Configuration Page Summary *(continued)*

Field	Function	Your Action
IPv4 Addresses and Prefixes	Displays the IPv4 addresses for the interfaces to which the dialer interface is assigned. **NOTE:** Ensure that the same IP subnet address is not configured on different dialer interfaces. Configuring the same IP subnet address on different dialer interfaces can result in inconsistency in the route and packet loss. Packets can be routed through any of the dialer interfaces with the IP subnet address, instead of being routed through the dialer interface to which the ISDN call is mapped.	Type an IP address and a prefix in the boxes. Click **Add**. To delete an IP address, highlight it in the list, and click **Delete**.
Dialer Options		
Activation Delay	Displays the time to wait before activating the backup interface once the primary interface is down.	Type a value, in seconds—for example, 30. The default value is 0 seconds with a maximum value of 60 seconds.
Deactivation Delay	Displays the time to wait before deactivating the backup interface once the primary interface is up.	Type a value, in seconds—for example, 30. The default value is 0 seconds with a maximum value of 60 seconds.
Dial String (required)	Displays the dialing number from your ISDN service provider.	Type the dialing number and click **Add**. To delete a dial string, highlight it and click **Delete**.
Pool (required)	Displays a list of dialer pools configured on br-*pim*/0/*port* interfaces.	Select a dialer pool from the list.
Multilink Dialer Options		
Load Interval	Defines the interval used to calculate the average load on the dialer interface for bandwidth on demand.	Type a value, in seconds—for example, 30. The default value is 60 seconds with a range of 20–80. The value must be a multiple of 10.
Load Threshold	Defines the threshold at which an additional ISDN interface is activated for bandwidth-on-demand. You specify the threshold as a percentage of the cumulative load of all UP links.	Type a percentage—for example, 80. The default value is 100 with a range of 0–100.
Backup Interface (for dial backup only)		
Interface to Backup	Displays a list of interfaces for ISDN backup.	Select an interface from the list for ISDN backup.
Dialer Watch List (for dialer watch only)		

Table 52: ISDN BRI Dialer Interface Quick Configuration Page Summary *(continued)*

Field	Function	Your Action
IPv4 Addresses and Prefixes	Displays the IPv4 addresses in the list of routes to be monitored by the dialer interface.	Type an IP address and a prefix in the boxes. Click **Add**. To delete an IP address, highlight it in the list, and click **Delete**.

Configuring ISDN Interfaces and Features with a Configuration Editor

To configure ISDN interfaces on a J-series device, you first configure the basic ISDN interface—either "Adding an ISDN BRI Interface (Required)" on page 189 or "Configuring Channelized T1/E1/ISDN PRI Interfaces for ISDN PRI Operation" on page 117. Second, configure the dialer interface by performing "Configuring Dialer Interfaces (Required)" on page 192.

To configure ISDN interfaces to back up primary device interfaces, you then configure a backup method—either "Configuring Dial Backup" on page 195, "Configuring Dialer Filters for Dial-on-Demand Routing Backup" on page 196, or "Configuring Dialer Watch" on page 198.

To configure ISDN interfaces for dial-in or callback, configure the basic ISDN BRI or PRI interface and then perform "Configuring Dial-In and Callback (Optional)" on page 205.

Perform other tasks as needed on your network.

This section contains the following topics:

Adding an ISDN BRI Interface (Required)

To enable ISDN BRI interfaces installed on your J-series device to work properly, you must configure the interface properties.

To configure an ISDN BRI network interface for the J-series device:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 53 on page 190.
3. Go on to "Configuring Dialer Interfaces (Required)" on page 192.

Table 53: Adding an ISDN BRI Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces br-1/0/3
Create the new interface—for example, br-1/0/3.	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type the name of the new interface, br-1/0/3. 3. Click **OK**.	
Configure dialer options. ■ Name the dialer pool—for example, isdn-dialer-group. ■ Set the dialer pool priority—for example, 255. Dialer pool priority has a range from 1 to 255, with 1 designating lowest-priority interfaces and 255 designating the highest-priority interfaces.	1. In the Encapsulation column, next to the new interface, click **Edit**. 2. Next to Dialer options, select **Yes**, and then click **Configure**. 3. Next to Pool, click **Add new entry**. 4. In the Pool identifier box, type isdn-dialer-group. 5. In the Priority box, type 255. 6. Click **OK** twice.	From the [edit interfaces br-1/0/3] hierarchy, enter set dialer-options pool isdn-dialer-group priority 255

Table 53: Adding an ISDN BRI Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure ISDN BRI properties. ■ Calling number sent to the ISDN switch during the call setup, which represents the caller's number—for example, 18005555555. ■ Service provider ID (SPID)—for example, 00108005555555. ■ Static TEI between 0 and 63 from your service provider—for example, 23. If the field is left blank, the device dynamically acquires a TEI. Also, if you have configured a second SPID, you cannot set a static TEI value. If you have a NTDMS-100 or NI1 switch, an additional box for a service provider ID is provided. If you are using a service provider that requires SPIDs, you cannot place calls until the interface sends a valid, assigned SPID to the service provider when accessing the ISDN connection. ■ Incoming called number—for example, 18883333456. Configure incoming call properties if you have remote locations dialing into the device through the ISDN interface.	1. Next to Isdn options, click **Configure**. 2. In the Calling number box, type 18005555555. 3. In the Spid1 box, type 00108005555555. 4. In the Static tei val box, type 23. 5. Next to Incoming called number, click **Add new entry**. 6. In the Called number box, type 18883333456. 7. Click **OK**.	1. To set the ISDN options, enter set isdn-options calling-number 18005555555 2. Enter set isdn-options spid1 00108005555555 3. Enter set isdn-options static-tei-val 23 4. set isdn-options incoming-called-number 18883333456
Select the type of ISDN switch—for example, ATT5E. The following switches are compatible with J-series devices: ■ ATT5E—AT&T 5ESS ■ ETSI—NET3 for the UK and Europe ■ NI1—National ISDN-1 ■ NTDMS–100—Northern Telecom DMS-100 ■ NTT—NTT Group switch for Japan	From the Switch type list, select **att5e**.	To select the switch type, enter set isdn-options switch-type att5e
Configure the Q.931 timer. Q.931 is a Layer 3 protocol for the setup and termination of connections. The default value for the timer is 10 seconds, but can be configured between 1 and 65536 seconds—for example, 15.	In the T310 box, type 15.	set isdn-options t310 15

Table 53: Adding an ISDN BRI Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure when the TEI negotiates with the ISDN provider. ■ **first-call**—Activation does not occur until a call is sent. ■ **power-up**—Activation occurs when the device is powered on. This is the default value.	1. From the Tei option list, select **power-up**. 2. Click **OK** to return to the Interfaces page.	To initiate activation at power-up, enter set isdn-options tei-option power-up

Configuring Dialer Interfaces (Required)

The dialer interface (dl) is a logical interface configured to establish ISDN connectivity. You can configure multiple dialer interfaces for different functions on the J-series device.

After configuring the dialer interface, you must configure a backup method—either dial backup, a dialer filter, or dialer watch.

To configure a logical dialer interface for the J-series device:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 54 on page 192.
3. To configure a backup method, go on to one of the following tasks:
   - "Configuring Dial Backup" on page 195.
   - "Configuring Dialer Filters for Dial-on-Demand Routing Backup" on page 196.
   - "Configuring Dialer Watch" on page 198.

Table 54: Adding a Dialer Interface to a Device

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces

Table 54: Adding a Dialer Interface to a Device *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Create the new interface—for example, dl0. Adding a description can differentiate between different dialer interfaces—for example, T1–backup.	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type dl0. 3. In the Description box, type T1–backup. 4. Click **OK**.	Create and name the interface: 1. edit dl0 2. set description T1-backup
Configure encapsulation options—for example, Cisco HDLC. ■ **Cisco HDLC**—For normal mode (when the device is using only one B-channel). Cisco-compatible High-Level Data Link Control is a group of protocols for transmitting data between network points. ■ **PPP**—For normal mode (when the device is using only one ISDN B-channel per call). Point-to-Point Protocol is for communication between two computers using a serial interface. ■ **Multilink PPP**—For multilink mode, when the device is using multiple B-channels per call. Multilink Point-to-Point Protocol (MLPPP) is a protocol for aggregating multiple constituent links into one larger PPP bundle. You can bundle up to eight B-channels.	1. In the Encapsulation column, next to the new interface, click **Edit**. 2. From the Encapsulation list, select **cisco-hdlc**.	Enter set encapsulation cisco-hdlc
Enter a hold-time value in milliseconds—for example, 60. The hold-time value is used to damp interface transitions. When an interface goes from up to down, it is not advertised as down to the rest of the system until it remains unavailable for the hold-time period. Similarly, an interface is not advertised as up until it remains operational for the hold-time period. The hold time is three times the interval at which keepalive messages are sent.	1. In the Hold time section, type 60 in the Down box. 2. In the Up box, type 60.	1. Enter set hold-time down 60 2. Enter set hold-time up 60
Create the logical unit—for example, 0. **NOTE:** You can set the logical unit to 0 only, unless you are configuring the dialer interface for Multilink PPP encapsulation.	1. Next to Unit, click **Add new entry**. 2. In the Interface unit number box, type 0. 3. Next to Dialer options, select **Yes**, and then click **Configure**.	Enter set unit 0

Table 54: Adding a Dialer Interface to a Device *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure dialer options. ■ **Activation delay**—Time to wait before activating the backup interface once the primary interface is down—for example, 30. Default value is 0 seconds with a maximum value of 60 seconds. Use only for dialer backup and dialer watch. ■ **Deactivation delay**—Time to wait before deactivating the backup interface once the primary interface is up—for example, 30. Default value is 0 seconds with a maximum value of 60 seconds. Use only for dialer backup and dialer watch. ■ **Idle timeout**—Time a connection is idle before disconnecting—for example, 30. Default value is 120 seconds with a range from 0 to 4294967295. This option is used only to configure a dialer filter. ■ **Initial route check**—Time to wait before checking if the primary interface is up—for example, 30. Default value is 120 seconds with a range of 1 to 300 seconds. This option is used only to configure dialer watch. ■ **Pool**—Name of a group of ISDN interfaces configured to use the dialer interface—for example, isdn-dialer-group. ■ **Redial delay**—Number of seconds to wait before redialing a failed outgoing ISDN call. Default value is 3 seconds with a range from 2 to 255.	1. In the Activation delay box, type 60. 2. In the Deactivation delay box, type 30. 3. In the Pool box, type isdn-dialer-group. 4. In the Redial delay box, type 5.	1. Enter edit unit 0 dialer-options 2. Enter set activation-delay 60 3. Enter set deactivation-delay 30 4. Enter set pool isdn-dialer-group 5. Enter set redial-delay 5
Configure the remote destination to call—for example, 5551212.	1. Next to Dial string, click **Add new entry**. 2. In the Value box, type 5551212. 3. Click **OK** until you return to the Unit page.	Enter set dial-string 5551212
Configure source and destination IP addresses for the dialer interface—for example, 172.20.10.2 and 172.20.10.1. (The destination IP address is optional.) **NOTE:** If you configure multiple dialer interfaces, ensure that the same IP subnet address is not configured on different dialer interfaces. Configuring the same IP subnet address on multiple dialer interfaces can result in inconsistency in the route and packet loss. The device might route packets through another dialer interface with the IP subnet address instead of through the dialer interface to which the ISDN modem call is mapped.	1. Select **Inet** under Family, and click **Edit**. 2. Next to Address, click **Add new entry**. 3. In the Source box, type 172.20.10.2. 4. In the Destination box, type 172.20.10.1. 5. Click **OK**.	1. From the [edit] hierarchy level, enter edit interfaces dl0 unit 0 2. Enter set family inet address 172.20.10.2 destination 172.20.10.1

Configuring Dial Backup

Dial backup allows one or more dialer interfaces to be configured as the backup link for a primary interface. The backup dialer interfaces are activated only when the primary interface fails. ISDN backup connectivity is supported on all interfaces except ls-0/0/0.

To configure a primary interface for backup connectivity:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 55 on page 195.
3. If you are finished configuring the device, commit the configuration.
4. Go on to any of the following optional tasks:
   - "Configuring Dial-on-Demand Routing Backup with OSPF Support (Optional)" on page 199.
   - "Configuring Bandwidth on Demand (Optional)" on page 200.
   - Configuring Dial-In and Callback (Optional) on page 205
5. To verify that the network interface is configured correctly, see "Verifying the ISDN Configuration" on page 211.

Table 55: Configuring an Interface for ISDN Backup

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces ge-0/0/0 unit 0
Select the physical interface for backup ISDN connectivity.	1. In the Interface name column, click the physical interface name. 1. Under Unit, in the Nested Configuration column, click **Edit**.	
Configure the backup dialer interface—for instance, dl0.0.	1. Next to Backup options, click **Configure**. 2. In the Interface box, type dl0.0. 3. Click **OK** until you return to the Interfaces page.	Enter set backup-options interface dl0.0

Configuring Dialer Filters for Dial-on-Demand Routing Backup

This dial-on-demand routing backup method allows an ISDN line to be activated only when network traffic configured as an "interesting packet" arrives on the network. Once the network traffic is sent, an inactivity timer is triggered and the connection is closed after the timer expires.

You define an interesting packet using the dialer filter feature of the device. There are two steps to configuring dial-on-demand routing backup using a dialer filter:

- Configuring the Dialer Filter on page 196
- Applying the Dial-on-Demand Dialer Filter to the Dialer Interface on page 197

Configuring the Dialer Filter

To configure the dialer filter:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 56 on page 196.
3. Go on to "Applying the Dial-on-Demand Dialer Filter to the Dialer Interface" on page 197.

Table 56: Configuring a Dialer Filter for Interesting Packets

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Firewall** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Firewall, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit firewall
Configure the dialer filter name—for example, int-packet.	1. Next to Inet, click **Configure** or **Edit**. 2. Next to Dialer filter, click **Add new entry**. 3. In the Filter name box, type int-packet.	1. Enter edit family inet 2. Then enter edit dialer-filter int-packet
Configure the dialer filter rule name—for example, term1. Configure term behavior. For example, you might want to configure your interesting packet as an ICMP packet. To configure the term completely, include both from and then statements.	1. Next to Term, click **Add new entry**. 2. In the Rule name box, type term1. 3. Next to From, click **Configure**. 4. From the Protocol choice list, select **Protocol**. 5. Next to Protocol, click **Add new entry**. 6. From the Value keyword list, select **icmp**. 7. Click **OK** twice to return to the Term page.	Enter set term term1 from protocol icmp

Table 56: Configuring a Dialer Filter for Interesting Packets *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure the then part of the dialer filter.	1. Next to Then, click **Configure**. 2. From the Designation list, select **Note**. 3. Click **OK**.	Enter set term1 then note

Applying the Dial-on-Demand Dialer Filter to the Dialer Interface

To complete dial-on-demand routing with dialer filter configuration:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 57 on page 197.
3. When you are finished configuring the device, commit the configuration.
4. Go on to any of the following optional tasks:
   - "Configuring Dial-on-Demand Routing Backup with OSPF Support (Optional)" on page 199.
   - "Configuring Bandwidth on Demand (Optional)" on page 200.
   - Configuring Dial-In and Callback (Optional) on page 205
5. To verify that the network interface is configured correctly, see "Verifying the ISDN Configuration" on page 211.

Table 57: Applying the Dialer Filter to the Dialer Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces dl0 unit 0
Select the dialer interface to apply the filter—for example, dl0.	1. In the Interface name column, click **dl0**. 2. Under Unit, in the Mtu column, click **Edit**.	
Select the dialer filter and apply it to the dialer interface.	1. In the Family section, next to Inet, click **Edit**. 2. Next to Filter, click **Configure**. 3. In the Dialer box, type int-packet, the dialer-filter configured in "Configuring the Dialer Filter" on page 196, as the dialer-filter. 4. Click **OK**.	1. Enter edit family inet filter 2. Enter set dialer int-packet

Configuring Dialer Watch

Dialer watch is a backup method that integrates backup dialing with routing capabilities and provides reliable connectivity without relying on a dialer filter to trigger outgoing ISDN connections. With dialer watch, the device monitors the existence of a specified route and if the route disappears, the dialer interface initiates the ISDN connection as a backup connection.

Adding a Dialer Watch Interface on the Device

To configure dialer watch:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 58 on page 198.
3. Go on to "Configuring the ISDN Interface for Dialer Watch" on page 198.

Table 58: Adding a Dialer Watch Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces
Select a dialer interface—for example, dl0. Adding a description, such as dialer-watch, can help you identify one dialer interface from another.	1. Under Interface name, select **dl0**. 2. In the Description box, type dialer-watch.	1. Enter edit dl0 2. Enter set description dialer-watch
On a logical interface—for example, 0—specify a dial pool—for example, dw-group—to link the dialer interface to at least one ISDN physical interface. Then configure the list of routes for dialer watch—for example, 172.27.27.0/24.	1. Under Unit, click the logical unit number 0. 2. Next to Dialer options, click **Edit**. 3. In the Pool box, type dw-group. 4. Next to Watch list, click **Add new entry**. 5. In the Prefix box, type 172.27.27.0/24. 6. Click **OK**.	1. Enter edit unit 0 dialer-options 2. Enter set pool dw-group 3. Enter set watch-list 172.27.27.0/24

Configuring the ISDN Interface for Dialer Watch

To configure the ISDN interface to participate as a dialer watch interface:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 59 on page 199.
3. If you are finished configuring the device, commit the configuration.
4. Go on to any of the following optional tasks:
   - "Configuring Dial-on-Demand Routing Backup with OSPF Support (Optional)" on page 199.
   - "Configuring Bandwidth on Demand (Optional)" on page 200.
   - Configuring Dial-In and Callback (Optional) on page 205
5. To verify that the network interface is configured correctly, see "Verifying the ISDN Configuration" on page 211.

Table 59: Configuring an ISDN Interface for Dialer Watch

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy, and select an ISDN physical interface—for example, br-1/0/3 for ISDN BRI. For ISDN PRI, select a channelized T1/E1/ISDN PRI interface—for example, ct1-1/0/1.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**. 3. Under Interface name: ■ For ISDN BRI, click **br-1/0/3**. ■ For ISDN PRI, click **ct1-1/0/1**.	From the [edit] hierarchy level: ■ For ISDN BRI, enter edit interfaces br-1/0/3 dialer-options pool isdn-dialer-group ■ For ISDN PRI, enter edit interfaces ct1-1/0/1 dialer-options isdn-dialer-group
Configure dialer watch options for each ISDN interface participating in the dialer watch feature. Each ISDN interface must have the same pool identifier to participate in dialer watch. Therefore, the dialer pool name isdn-dialer-group, for the dialer watch interface configured in Table 58 on page 198, is used when configuring the ISDN interface.	1. Next to Dialer options, click **Edit**. 2. Next to Pool, click **Add new entry**. 3. In the Pool identifier box, type isdn-dialer-group. 4. Click **OK**.	

Configuring Dial-on-Demand Routing Backup with OSPF Support (Optional)

Two types of routing protocol traffic are used by OSPF to establish and maintain network structure. First, periodic hello packets are sent over each link for neighbor discovery and maintenance. Second, OSPF protocol traffic achieves and maintains link-state database synchronization between devices. The OSPF demand circuit feature removes the periodic nature of both traffic types and reduces the amount of OSPF traffic by removing all OSPF protocol traffic from a demand circuit when the routing domain is in a steady state. This feature allows the underlying data-link connections to be closed when no application traffic is on the network.

You must configure OSPF on the device before configuring on-demand routing backup with OSPF support. For information on configuring OSPF, see "Configuring an OSPF Network" on page 359.

To configure OSPF demand circuits:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 60 on page 200.
3. If you are finished configuring the device, commit the configuration.
4. To verify that the network interface is configured correctly, see "Verifying the ISDN Configuration" on page 211.

Table 60: Configuring OSPF Demand Circuits

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Protocols** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Configure** or **Edit**. 3. Next to Ospf, click **Configure**. 4. Next to Area, click **Add new entry**. 5. In the Area id box, type 0.0.0.0.	From the [edit] hierarchy level, enter edit protocols ospf area 0.0.0.0
Configure OSPF on-demand circuits for each ISDN dialer interface participating as an on-demand routing interface—for example, dl0.	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type dl0.0. 3. Select **Demand circuit**. 4. Click **OK**.	1. Enter edit interface dl0 2. Enter set demand-circuit

Configuring Bandwidth on Demand (Optional)

You can define a threshold for network traffic on the device using the dialer interface and ISDN interfaces. A number of ISDN interfaces are aggregated together into a bundle and assigned a single dialer profile. Initially, only one ISDN link is active and all packets are sent through this interface. When a configured threshold is exceeded, the dialer interface activates another ISDN link and initiates a data connection. The threshold is specified as a percentage of the cumulative load of all UP links that are part of the bundle. When the cumulative load of all UP links, not counting the most recently activated link, is at or below the threshold, the most recently activated link is deactivated.

Configuring Dialer Interfaces for Bandwidth on Demand

To configure a dialer interface for bandwidth-on-demand:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 61 on page 201.
3. Go on to "Configuring an ISDN Interface for Bandwidth on Demand" on page 204.

Table 61: Configuring a Dialer Interface for Bandwidth on Demand

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy, and select a dialer interface—for example, dl0.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Edit**. 3. Next to dl0, click **Edit**.	From the [edit] hierarchy level, enter `edit interfaces dl0`
Configure multilink properties on the dialer interface.	1. Select **multilink-ppp** as the encapsulation type.	Enter `set encapsulation multilink-ppp`

Configure the dialer options.
■ **Dial string**—Telephone number for the interface to dial that establishes ISDN connectivity—for example, 4085550115. You can configure a maximum of 15 dial strings per dialer interface.
■ **Load interval**—Interval of time used to calculate the average load on the dialer interface—for example, 90. Default value is 60 seconds with a range of 20-180 seconds. The value must be a multiple of 10.
■ **Load threshold**—Threshold above which an additional ISDN interface is activated, specified as a percentage of the cumulative load of all UP links—for example 95. Default value is 100 with a range of 0–100.
■ **Pool**—Name of a group of ISDN interfaces configured to use the dialer interface—for example, bw-pool.
	1. In the Unit section, click **Dialer options** under Encapsulation.
	2. Next to Dial string, click **Add new entry**.
	3. In the Value box, type 4085550115 and click **OK**.
	4. In the Load interval box, type 90.
	5. In the Load threshold box, type 95.
	6. In the Pool box, type bw-pool.
	7. Click **OK**.
		1. Enter
		`edit unit 0`
		2. Enter
		`edit dialer-options`
		3. Enter
		`set dial-string 4085550115`
		4. Enter
		`set load-interval 90`
		5. Enter
		`set load-threshold 95`
		6. Enter
		`set pool bw-pool`

Table 61: Configuring a Dialer Interface for Bandwidth on Demand *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure unit properties. To configure multiple dialer interfaces for bandwidth-on-demand, increment the unit number—for example, dl0.1, dl0.2, and so on. **F max period**—Maximum number of compressed packets allowed between the transmission of full packets—for example, 100. The value can be between 1 and 65535.	1. Next to Compression, select **Yes**, and then click **Configure**. 2. Select **Rtp**, and then click **Configure**. 3. In the F max period box, type 100. 4. Next to Queues, click **Add new entry**. 5. From the Value list, select **q3**. 6. Click **OK** until you return to the Unit page.	1. From the [edit] hierarchy level, enter edit interfaces dl0 unit 0 2. Enter set compression rtp f-max-period 500 queues q3
Configure logical properties. ■ **Fragment threshold**–Maximum size, in bytes, for multilink packet fragments. The value can be between 128 and 16320 bytes, for example, 1024. The default is 0 bytes (no fragmentation). Any nonzero value must be a multiple of 64 bytes. ■ **Maximum received reconstructed unit (MRRU)**—This value is expressed as a number between 1500 and 4500 bytes—for example, 1500.	1. In the Fragment threshold box, type 1024. 2. In the Mrru box, type 1500. 3. Click **OK** until you return to the main Configuration page.	1. Enter set fragment-threshold 1024 2. Enter set mrru 1500
Define a CHAP access profile with a client and a secret password. For example, define bw–profile with client 1 and password my-secret.	1. On the main Configuration page next to Access, click **Configure** or **Edit**. 2. Next to Profile, click **Add new entry**. 3. In the Profile name box, type bw-profile. 4. Next to Client, click **Add new entry**. 5. In the Name box, type client1. 6. In the Chap secret box, type my-secret. 7. Click **OK** until you return to the main Configuration page.	From the [edit] hierarchy level, enter set access profile bw-profile client client1 chap-secret my-secret

Table 61: Configuring a Dialer Interface for Bandwidth on Demand *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the appropriate dialer interface level in the configuration hierarchy—for example, dl0 unit 0.	1. On the main Configuration page next to Interfaces, click **Configure** or **Edit**. 2. In the interface name box, click **dl0**. 3. In the Interface unit number box, click **0**.	From the [edit] hierarchy level, enter edit interfaces dl0 unit 0
Configure CHAP on the dialer interface and specify a unique profile name containing a client list and access parameters—for example, bw-profile.	1. Next to Ppp options, click **Configure**. 2. Next to Chap, click **Configure**. 3. Next to Access data, select Access profile. 4. In the Access profile box, type bw-profile. 5. Click **OK**.	Enter set ppp-options chap access-profile bw-profile
Configure packet compression. You can configure the following compression types: ■ **ACFC (address and control field compression)**—Conserves bandwidth by compressing the address and control fields of PPP-encapsulated packets. ■ **PFC (protocol field compression)**—Conserves bandwidth by compressing the protocol field of a PPP-encapsulated packet.	1. Under Compression, select **Acfc**. 2. Click **OK** until you return to the Unit page.	Enter set ppp-options compression acfc

Table 61: Configuring a Dialer Interface for Bandwidth on Demand *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure the dialer interface to be assigned an IP address in one of the following ways: ■ Assign source and destination IP addresses as described in Table 54 on page 192—for example, 172.20.10.2 and 172.20.10.1. (The destination IP address is optional.) ■ Obtain an IP address by negotiation with the remote end. This method might require the access concentrator to use a RADIUS authentication server. ■ Derive the source address from a specified interface—for example, the loopback interface, lo0.0—and assign a destination address—for example, 192.168.1.2. The specified interface must include a logical unit number and have a configured IP address.	Next to Inet, select **Yes** and click **Configure**. Select one of the following IP address configurations: To assign source and destination IP addresses: 1. Next to Address, click **Add new entry**. 2. In the Source box, type 172.20.10.2. 3. In the Destination box, type 172.20.10.1. 4. Click **OK**. To obtain an IP address from the remote end: 1. Next to Negotiate address, select the **Yes** check box. 2. Click **OK**. To derive the source address and assign the destination address: 1. Next to Unnumbered address, select the **Yes** check box and click **Configure**. 2. In the Destination box, type 192.168.1.2. 3. In the Source box, type lo0.0. 4. Click **OK**.	Do one of the following: ■ To assign source and destination IP addresses, enter set family inet address 172.20.10.2 destination 172.20.10.1 ■ To obtain an IP address from the remote end, enter set family inet negotiate-address ■ To derive the source address and assign the destination address, enter set family inet unnumbered-address lo0.0 destination 192.168.1.2

Configuring an ISDN Interface for Bandwidth on Demand

To configure bandwidth on demand on the ISDN interface:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 62 on page 205. Repeat these tasks for each ISDN interface participating in the aggregated link.
3. If you are finished configuring the device, commit the configuration.
4. To verify that the network interface is configured correctly, see "Verifying the ISDN Configuration" on page 211.

Table 62: Configuring an ISDN Interface for Bandwidth on Demand

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy, and select an ISDN BRI physical interface—for example, br-1/0/3. For ISDN PRI, select a channelized T1/E1/ISDN PRI interface—for example, ct1-1/0/1.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Edit**. 3. Under Interface name: ■ For ISDN BRI, click **br-1/0/3**. ■ For ISDN PRI, click **ct1-1/0/1**.	From the [edit] hierarchy level: ■ For ISDN BRI, enter edit interfaces br-1/0/3 ■ For ISDN PRI, enter edit interfaces ct1-1/0/1
Because each ISDN interface must have the same pool identifier to participate in bandwidth on demand, use the dialer pool name bw-pool, the dialer interface configured in Table 61 on page 201, to configure the ISDN interfaces participating in the pool. For ISDN BRI, you can group up to four ISDN interfaces together when configuring bandwidth on demand, for a total of eight B-channels (two channels per interface) providing connectivity. For ISDN PRI, the pool limit is eight B-channels per channelized T1/E1/ISDN PRI port.	1. Next to Dialer options, click **Dialer options**. 2. Next to Pool, click **Add new entry**. 3. In the Pool identifier box, type the name of the dialer pool—for example, bw-pool. 4. Click **OK**.	Enter set dialer-options pool bw-pool

Configuring Dial-In and Callback (Optional)

If you are a service provider or a corporate data center into which a remote location dials in during an emergency, you can configure your Juniper Networks device to accept incoming ISDN calls originating from the remote location, or reject the incoming calls and call back the remote location. The callback feature lets you control access by allowing only specific remote locations to connect to the device. You can also configure the device to reject all incoming ISDN calls.

NOTE: Incoming voice calls are currently not supported.

When it receives an incoming ISDN call, the Juniper Networks device matches the incoming call's caller ID against the caller IDs configured on its dialer interfaces. If an exact match is not found and the incoming call's caller ID has more digits than the configured caller IDs, the device performs a right-to-left match of the incoming call's caller ID with the configured caller IDs and accepts the incoming call if a match is found. For example, if the incoming call's caller ID is 4085550115 and the caller ID configured on a dialer interface is 5550115, the incoming call is accepted. Each dialer interface accepts calls from only callers whose caller IDs are configured on it.

The dialer interface of the device and the dialer interface of the remote device must have the same encapsulation—PPP, Multilink PPP, or Cisco HDLC. If the encapsulation is different, the ISDN call is dropped. Table 63 on page 206 describes how the device performs encapsulation monitoring.

Table 63: Encapsulation Monitoring by Juniper Networks Devices

Encapsulation on Juniper Networks Device's Interface	Encapsulation on Remote Router's Dialer Interface	Possible Action on Juniper Networks Device's Dialer Interface	Encapsulation Monitoring and Call Status
PPP	PPP	■ Accepts an incoming call ■ Rejects an incoming call and calls back the incoming number when callback is enabled on the dialer interface	Device performs encapsulation monitoring. ISDN call is *successful* because encapsulation matches.
Multilink PPP	Multilink PPP		
PPP	Multilink PPP or Cisco HDLC		Device performs encapsulation monitoring. ISDN call is *dropped* because of encapsulation mismatch.
Multilink PPP	PPP or Cisco HDLC		
PPP or Multilink PPP	PPP, Multilink PPP, or Cisco HDLC	■ Dials out ■ Accepts an incoming call as a result of having originally dialed out, because the dialer interface of the remote device has callback enabled	Device does not perform encapsulation monitoring. Success of the ISDN call depends on the encapsulation monitoring capability of the remote device.
Cisco HDLC	PPP, Multilink PPP, or Cisco HDLC	■ Dials out ■ Accepts an incoming call ■ Accepts an incoming call as a result of having originally dialed out, because the dialer interface of the remote device has callback enabled ■ Rejects an incoming call and calls back the incoming number when callback is enabled on the dialer interface	

This section contains the following topics:

Configuring Dialer Interfaces for Dial-In and Callback

To configure a dialer interface for dial-in and callback:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 64 on page 207.
3. If you are finished configuring the device, commit the configuration.
4. To verify that the network interface is configured correctly, see "Verifying the ISDN Configuration" on page 211.

Table 64: Configuring the Dialer Interface for Dial-In and Callback

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy, and select a dialer interface—for example, dl0.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or**Edit**. 3. Next to dl0, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces dl0
On a logical interface—for example, 0—configure the incoming map options for the dialer interface. To use dial-in, you must configure an incoming map on the dialer interface. ■ **Accept all**—Dialer interface accepts all incoming calls. You can configure this option for only one of the dialer interfaces associated with an ISDN physical interface. The dialer interface configured to accept all calls is used only if the incoming call's caller ID does not match the caller IDs configured on other dialer interfaces. ■ **Caller**—Dialer interface accepts calls from a specific caller ID—for example, 4085550115. You can configure a maximum of 15 caller IDs per dialer interface. The same caller ID must not be configured on different dialer interfaces. However, you can configure caller IDs with more or fewer digits on different dialer interfaces. For example, you can configure the caller IDs 14085550115, 4085550115, and 5550115 on different dialer interfaces.	1. In the Unit section, for logical unit number 0, click **Dialer options** under Encapsulation. 2. Next to Incoming map, click **Configure**. 3. From the Caller type menu, select **Caller**. Next to Caller, click **Add new entry**. 4. In the Caller id box, type 4085550115. 5. Click **OK** until you return to the Dialer option page.	1. Enter edit unit 0 2. Enter edit dialer-options 3. Enter set incoming-map caller 4085550115

Table 64: Configuring the Dialer Interface for Dial-In and Callback *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure callback options for the dialer interface ■ **Callback**—Enable this feature to allow the ISDN interface to reject incoming calls, wait for 5 seconds (the default callback wait period), and then call back the incoming number. Before configuring callback on a dialer interface, ensure that the following conditions exist: ■ The dialer interface is not configured as a backup for a primary interface. ■ The dialer interface does not have a watch list configured. ■ Only one dial string is configured for the dialer interface. ■ Dial-in is configured on the dialer interface of the remote device that is dialing in. ■ **Callback wait period**—Number of seconds to wait before redialing an incoming ISDN call.	1. Select **Callback**. 2. In the Callback wait period box, type 5.	1. Enter set callback 2. Enter set callback-wait-period 5

Configuring an ISDN Interface to Screen Incoming Calls

By default, an ISDN interface is configured to accept all incoming calls. If multiple devices are connected to the same ISDN line, you can configure an ISDN interface to screen incoming calls based on the incoming called number.

You can configure the incoming called numbers that you want an ISDN interface to accept. You can also use the reject option to configure a called number that you want an ISDN interface to ignore because the number belongs to another device connected to the same ISDN line. For example, if another device on the same ISDN line has the called number 4085551091, you can configure the called number 4085551091 with the reject option on the ISDN interface so that it does not accept calls with that number.

When it receives an incoming ISDN call, the device matches the incoming called number against the called numbers configured on its ISDN interfaces. If an exact match is not found, or if the called number is configured with the reject option, the incoming call is ignored. Each ISDN interface accepts only the calls whose called numbers are configured on it.

To configure an ISDN interface to screen incoming ISDN calls:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 65 on page 209.

3. If you are finished configuring the device, commit the configuration.
4. To verify that the network interface is configured correctly, see "Verifying the ISDN Configuration" on page 211.

Table 65: Configuring an ISDN Interface to Screen Incoming ISDN Calls

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy, and select an ISDN physical interface—for example, br-1/0/3. For ISDN PRI, select a channelized T1/E1/ISDN PRI interface—for example, ct1-1/0/1.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**. 3. Under Interface name: ■ For ISDN BRI, click **br-1/0/3**. ■ For ISDN PRI, click **ct1-1/0/1**.	From the [edit] hierarchy level: ■ For ISDN BRI, enter edit interfaces br-1/0/3 ■ For ISDN PRI, enter edit interfaces ct1-1/0/1
Configure the incoming called number—for example, 4085550115—for the ISDN interface. To configure the ISDN interface to ignore the incoming called number, use the reject option.	1. Next to Isdn options, click **Edit**. 2. Next to Incoming called number, click **Add new entry**. 3. In the Called number box, type 4085550115. 4. Click **OK**.	Enter set isdn-options incoming-called-number 4085550115

Configuring the Device to Reject Incoming ISDN Calls

By default, the device is configured to accept incoming ISDN calls. The incoming calls are accepted if dial-in is configured on the device. You can configure the device to reject all incoming ISDN calls.

To configure the device to reject incoming ISDN calls:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 66 on page 210.
3. If you are finished configuring the device, commit the configuration.
4. To verify that the network interface is configured correctly, see "Verifying the ISDN Configuration" on page 211.

Table 66: Configuring the Device to Reject Incoming ISDN Calls

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Processes** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to System, click **Configure** or **Edit**. 3. Next to Processes, click **Configure**. 4. Next to Isdn signaling, click **Configure**.	From the [edit] hierarchy level, enter set system processes isdn-signaling reject-incoming
Configure the device to reject incoming calls.	1. Select the **Reject Incoming** check box. 2. Click **OK**.	

Disabling Dialing Out Through Dialer Interfaces

The JUNOS ISDN dialer services process manages dialing out through dialer interfaces. You can disable dialing out through all dialer interfaces by disabling the dialer services process.

CAUTION: Never disable a software processes unless instructed to do so by a Customer Support engineer.

To disable dialing out through dialer interfaces:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 67 on page 210.
3. If you are finished configuring the device, commit the configuration.

Table 67: Disabling Dialing Out Through Dialer Interfaces

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Processes** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to System, click **Configure** or **Edit**. 3. Next to Processes, click **Configure**. 4. Next to Dialer services, click **Configure**.	From the [edit] hierarchy level, enter set system processes dialer-services disable
Disable dialing out through dialer interfaces.	1. Select the **Disable** check box. 2. Click **OK**.	

Disabling ISDN Signaling

The JUNOS ISDN signaling process manages ISDN signaling by initializing ISDN connections. You can disable ISDN signaling by disabling the ISDN signaling process.

CAUTION: Never disable a software processes unless instructed to do so by a Customer Support engineer.

To disable ISDN signaling:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 68 on page 211.
3. If you are finished configuring the device, commit the configuration.

Table 68: Disabling ISDN Signaling

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Processes** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to System, click **Configure** or **Edit**. 3. Next to Processes, click **Configure**. 4. Next to Isdn signaling, click **Configure**.	From the [edit] hierarchy level, enter set system processes isdn-signaling disable
Disable ISDN signaling on the device.	1. Select the **Disable** check box. 2. Click **OK**.	

Verifying the ISDN Configuration

To verify an ISDN configuration, perform the following tasks:

- Displaying the ISDN Status on page 212
- Verifying an ISDN BRI Interface on page 213
- Verifying an ISDN PRI Interface and Checking B-Channel Interface Statistics on page 214
- Checking D-Channel Interface Statistics on page 215
- Displaying the Status of ISDN Calls on page 217
- Verifying Dialer Interface Configuration on page 218

Displaying the ISDN Status

Purpose Display the status of ISDN service on the ISDN interface. For example, you can display ISDN BRI status on the br-6/0/0 interface and ISDN PRI status on the ct1-2/0/0 interface.

Action From the operational mode in the CLI, enter show isdn status.

Sample Output

```
user@host> show isdn status
Interface: br-6/0/0
 Layer 1 status: active
 Layer 2 status:
  CES: 0, Q.921: up, TEI: 12
 Layer 3 status: 1 Active calls
  Switch Type           : ETSI
  Interface Type        : USER
  T310                  : 10 seconds
 Tei Option            : Power Up


user@host> show isdn status
Interface: ct1-2/0/0
 Layer 1 status: active
 Layer 2 status:
  CES: 0, Q.921: up, TEI: 0
 Layer 3 status: 8 Active calls
  Switch Type           : NI2
  Interface Type        : USER
  T310                  : 10 seconds
  Tei Option            : Power Up
```

Meaning The output shows a summary of interface information. Verify the following information:

- Interface—ISDN interface currently active on the device. For ISDN BRI, the interface is a br-*pim*/0/*port* interface, as shown in the first example for br-6/0/0. For ISDN PRI, the interface displayed is a channelized T1 or channelized E1 interface, as shown in the second example for ct1-2/0/0.
- Layer 1 status—Indicates whether Layer 1 is active or inactive.
- Layer 2 status—Indicates whether Q.921 (the D-channel protocol) is up or down.
- TEI—Assigned terminal endpoint identifier (TEI) number.
- Layer 3 status—Number of active calls.
- Switch Type—Type of ISDN switch connected to the device interface.
- Interface Type—Default value for the local interface.
- Calling number—Telephone number configured for dial-out.
- T310—Q.931-specific timer.
- TEI Option—Indicates when TEI negotiations occur on the interface.

Related Topics For a complete description of show isdn status output, see the *JUNOS Interfaces Command Reference*.

Verifying an ISDN BRI Interface

Purpose Verify that the ISDN BRI interface is correctly configured.

Action From the CLI, enter the show interfaces extensive command. Alternatively, from the J-Web interface select **Monitor > Interfaces > br-6/0/0**.

Sample Output

```
user@host> show interfaces br-6/0/0 extensive
Physical interface: br-6/0/0, Enabled, Physical link is Up
  Interface index: 143, SNMP ifIndex: 59, Generation: 24
  Type: BRI, Link-level type: Controller, MTU: 4092, Clocking: 1, Speed: 144kbps,

  Parent: None
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps Internal: 0x4000
  Link type      : Full-Duplex
  Link flags     : None
  Physical info  : S/T
  Hold-times     : Up 0 ms, Down 0 ms
  Last flapped   : 2005-12-07 12:21:11 UTC (04:07:26 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
 0,
    Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
 0
```

Meaning The output shows a summary of interface information. Verify the following information:

- The physical interface is Enabled. If the interface is shown as Disabled, do either of the following:
  - In the CLI configuration editor, delete the disable statement at the [edit interfaces *interface-name*] level of the configuration hierarchy.
  - In the J-Web configuration editor, clear the **Disable** check box on the **Interfaces > *interface-name*** page.
- The physical link is Up. A link state of Down indicates a problem with the interface module, interface port, or physical connection (link-layer errors).
- The Last Flapped time is an expected value. The Last Flapped time indicates the last time the physical interface became unavailable and then available again. Unexpected flapping indicates likely link-layer errors.
- The traffic statistics reflect expected input and output rates. Verify that the number of inbound and outbound bytes and packets matches expected throughput for the physical interface. To clear the statistics and see only new changes, use the clear interfaces statistics *interface-name* command.

Related Topics For a complete description of show interfaces (ISDN BRI) output, see the *JUNOS Interfaces Command Reference*.

Verifying an ISDN PRI Interface and Checking B-Channel Interface Statistics

Purpose Verify that an ISDN B-channel interface is operating properly. For ISDN PRI, verify that a B-channel interface is configured correctly. (To display a list of B-channel interfaces, enter the show isdn calls command.)

Action From the CLI, enter the show interfaces extensive command. Alternatively, from the J-Web interface select **Monitor > Interfaces > bc-0/0/4:1**.

Sample Output

```
user@host> show interfaces bc-0/0/4:1 extensive
Physical interface: bc-0/0/4:1, Administratively down, Physical link is Up
  Interface index: 145, SNMP ifIndex: 75, Generation: 26
  Type: Serial, Link-level type: Multilink-PPP, MTU: 1510, Clocking: Internal,
  Speed: 64kbps,
  Parent: br-0/0/4 Interface index 143
  Device flags   : Present Running
  Interface flags: Admin-Test SNMP-Traps 16384
  Link type      : Full-Duplex
  Link flags     : None
  Physical info  : Unspecified
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: Unspecified, Hardware address: Unspecified
  Alternate link address: Unspecified
  CoS queues     : 8 supported, 8 maximum usable queues
  Last flapped   : Never
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                 5787                    0 bps
   Output bytes  :                 3816                    0 bps
   Input  packets:                  326                    0 pps
   Output packets:                  264                    0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
 6,
    Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
 0
    Queue counters      Queued packets   Transmitted Packets   Dropped packets
       0 best-effort            314335                     0                 0
       1 best-effort                 0                     0                 0
       2 best-effort                 5                     0                 0
       3 best-effort              5624                  5624                 0
    Packet Forwarding Engine configuration:
    Destination slot: 5, PLP byte: 1 (0x00)
    CoS transmit queue               Bandwidth               Buffer Priority
Limit
                              %           bps     %           usec
    0 best-effort            95          60800    95              0      low
none
    3 network-control         5           3200     5              0      low
none

  Logical interface bc-0/0/4:1.0 (Index 71) (SNMP ifIndex 61) (Generation 33)
    Flags: Device Down Point-To-Point SNMP-Traps Encapsulation: PPP
    Protocol mlppp, Multilink bundle: dl0.0, MTU: 1506, Generation: 18, Route
```

```
table: 0
```

Meaning The output shows a summary of B-channel interface information. Verify the following information:

- The physical interface is Enabled. If the interface is shown as Disabled, do either of the following:
  - In the CLI configuration editor, delete the disable statement at the [edit interfaces *interface-name*] level of the configuration hierarchy.
  - In the J-Web configuration editor, clear the **Disable** check box on the **Interfaces > *interface-name*** page.
- The physical link is Up. A link state of Down indicates a problem with the interface module, interface port, or physical connection (link-layer errors).
- For ISDN BRI, the Parent interface is a br-*pim*/0/*port* interface—br-0/0/4 in this example. For ISDN PRI, the Parent interface is a channelized T1 or channelized E1 interface—ct1–*pim*/0/*port* or ce1–*pim*/0/*port*.
- The Last Flapped time is an expected value. The Last Flapped time indicates the last time the physical interface became unavailable and then available again. Unexpected flapping indicates possible link-layer errors.
- The traffic statistics reflect expected input and output rates. Verify that the number of inbound and outbound bytes and packets matches expected throughput for the physical interface. To clear the statistics and see only new changes, use the clear interfaces statistics *interface-name* command.

Related Topics For a complete description of show interfaces (ISDN B-channel) output, see the *JUNOS Interfaces Command Reference*.

Checking D-Channel Interface Statistics

Purpose Verify that the ISDN D-channel interface is operating properly. For ISDN PRI, verify that the D-channel interface is configured correctly.

Action From the CLI, enter the show interfaces extensive command. Alternatively, from the J-Web interface select **Monitor > Interfaces > dc-0/0/4**.

Sample Output

```
user@host> show interfaces dc-0/0/4 extensive
Physical interface: dc-0/0/4, Enabled, Physical link is Up
  Interface index: 144, SNMP ifIndex: 60, Generation: 25
  Type: Serial, Link-level type: 55, MTU: 4092, Clocking: Internal, Speed: 16kbps,

  Parent: br-0/0/4 Interface index 143
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link type      : Full-Duplex
  Link flags     : None
  Physical info  : Unspecified
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: Unspecified, Hardware address: Unspecified
  Alternate link address: Unspecified
  Last flapped   : 2005-12-07 12:21:12 UTC (05:46:00 ago)
  Statistics last cleared: Never
```

```
  Traffic statistics:
   Input  bytes  :                13407                    0 bps
   Output bytes  :                16889                    0 bps
   Input  packets:                 3262                    0 pps
   Output packets:                 3262                    0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
0,
    Resource errors: 0
  Output errors:
    Carrier transitions: 1, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
0
  ISDN  alarms   : None
  ISDN media:            Seconds        Count  State
    LOF                       0            1  OK
    LOS                       0            0  OK

  Logical interface dc-0/0/4.32767 (Index 70) (SNMP ifIndex 72) (Generation 8)
    Flags: Point-To-Point SNMP-Traps Encapsulation: 60
    Traffic statistics:
     Input  bytes  :                13407
     Output bytes  :                82129
     Input  packets:                 3262
     Output packets:                 3262
    Local statistics:
     Input  bytes  :                13407
     Output bytes  :                82129
     Input  packets:                 3262
     Output packets:                 3262
```

Meaning The output shows a summary of D-channel interface information. Verify the following information:

- The physical interface is Enabled. If the interface is shown as Disabled, do either of the following:
  - In the CLI configuration editor, delete the disable statement at the [edit interfaces *interface-name*] level of the configuration hierarchy.
  - In the J-Web configuration editor, clear the **Disable** check box on the **Interfaces > *interface-name*** page.
- The physical link is Up. A link state of Down indicates a problem with the interface module, interface port, or physical connection (link-layer errors).
- For ISDN BRI, the Parent interface is a br-*pim*/0/*port* interface—br-0/0/4 in this example. For ISDN PRI, the Parent interface is a channelized T1 or channelized E1 interface—ct1–*pim*/0/*port* or ce1–*pim*/0/*port*.
- The Last Flapped time is an expected value. The Last Flapped time indicates the last time the physical interface became unavailable and then available again. Unexpected flapping indicates possible link-layer errors.
- The traffic statistics reflect expected input and output rates. Verify that the number of inbound and outbound bytes and packets matches expected throughput for the physical interface. To clear the statistics and see only new changes, use the clear interfaces statistics *interface-name* command.

Related Topics For a complete description of show interfaces (ISDN D-channel) output, see the *JUNOS Interfaces Command Reference*.

Displaying the Status of ISDN Calls

Purpose Display the status of ISDN calls. This information helps you to verify the dialer interface configuration as described in "Verifying Dialer Interface Configuration" on page 218. The command also provides a list of the B-channels configured on an ISDN BRI or ISDN PRI interface.

Action From the CLI, enter the show isdn calls command.

Sample Output

```
user@host> show isdn calls
Interface: bc-6/0/0:1
  Status: No call in progress
  Most recent error code: No error
Interface: bc-6/0/0:2
  Status: Connected to 384070
  Call Duration: 43 seconds
  Call Direction: Dialout
  Most recent error code: No error


user@host> show isdn calls
Interface: bc-2/0/0:1
  Status: Connected to 384010
  Call Duration: 49782 seconds
  Call Direction: Dialin
  Most recent error code: destination out of order
Interface: bc-2/0/0:2
  Status: Connected to 384011
  Call Duration: 49782 seconds
  Call Direction: Dialin
  Most recent error code: destination out of order
Interface: bc-2/0/0:3
  Status: Connected to 384020
  Call Duration: 49782 seconds
  Call Direction: Dialin
  Most recent error code: destination out of order
...
Interface: bc-2/0/0:20
  Status: No call in progress
  Most recent error code: No error
Interface: bc-2/0/0:21
  Status: No call in progress
  Most recent error code: No error
Interface: bc-2/0/0:22
  Status: No call in progress
  Most recent error code: No error
Interface: bc-2/0/0:23
  Status: No call in progress
  Most recent error code: No error
```

Meaning The output shows a summary of B-channel interfaces and the active ISDN calls on the interfaces. The first example shows the two B-channels on an ISDN BRI interface—bc-2/0/0:1 and bc-2/0/0:2. The second example indicates B-channels bc-2/0/0:1 through bc-2/0/0:23, the 23 B-channels on an ISDN PRI interface. Determine the following information:

- The interfaces on which ISDN calls are in progress
- Whether the call is a dial-in call, dial-out call, or a callback call

Related Topics For a complete description of show isdn calls output, see the *JUNOS Interfaces Command Reference*.

Verifying Dialer Interface Configuration

Purpose Verify that the dialer interface is correctly configured. To determine the ISDN interfaces on which calls are taking place, see "Displaying the Status of ISDN Calls" on page 217.

Action From the CLI, enter the show interfaces dl0 extensive command. Alternatively, from the J-Web interface select **Monitor > Interfaces > dl0**.

Sample Output

```
user@host> show interfaces dl0 extensive
Physical interface: dl0, Enabled, Physical link is Up
  Interface index: 173, SNMP ifIndex: 26, Generation: 77
  Type: 27, Link-level type: PPP, MTU: 1504, Clocking: Unspecified, Speed:
Unspecified
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : Keepalives
  Physical info  : Unspecified
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: Unspecified, Hardware address: Unspecified
  Alternate link address: Unspecified
  Last flapped   : Never
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                13859                    0 bps
   Output bytes  :                    0                    0 bps
   Input  packets:                  317                    0 pps
   Output packets:                    0                    0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
 0,
    Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
 0

  Logical interface dl0.0 (Index 76) (SNMP ifIndex 28) (Generation 148)
    Flags: Point-To-Point SNMP-Traps 0x4000 LinkAddress 23-0 Encapsulation: PPP
    Dialer:
      State: Active, Dial pool: 1
      Dial strings: 384070
      Subordinate interfaces: bc-6/0/0:2 (Index 172)
      Watch list: 11.12.13.14/32
      Activation delay: 0, Deactivation delay: 0
      Initial route check delay: 120
      Redial delay: 3
      Callback wait period: 5
      Load threshold: 0, Load interval: 60
    Bandwidth: 64kbps
    Traffic statistics:
     Input  bytes  :                24839
```

```
    Output bytes  :                17792
    Input  packets:                  489
    Output packets:                  340
  Local statistics:
    Input  bytes  :                10980
    Output bytes  :                17792
    Input  packets:                  172
    Output packets:                  340
  Transit statistics:
    Input  bytes  :                13859                    0 bps
    Output bytes  :                    0                    0 bps
    Input  packets:                  317                    0 pps
    Output packets:                    0                    0 pps
 Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
 Keepalive statistics:
   Input : 0 (last seen: never)
   Output: 36 (last sent 00:00:09 ago)
 LCP state: Opened
 NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured,
 mpls: Not-configured
 CHAP state: Success
   Protocol inet, MTU: 1500, Generation: 74, Route table: 0
     Flags: Negotiate-Address
     Addresses, Flags: Kernel Is-Preferred Is-Primary
       Destination: 43.1.1.2, Local: 43.1.1.19, Broadcast: Unspecified,
       Generation: 37


user@host> show interfaces dl0 extensive
Physical interface: dl0, Enabled, Physical link is Up
  Interface index: 140, SNMP ifIndex: 35, Generation: 141
  Link-level type: LinkService, MTU: 1504
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Last flapped   : 2007-02-27 01:50:38 PST (1d 15:48 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :             42980144                    0 bps
   Output bytes  :                  504                    0 bps
   Input  packets:               934346                    0 pps
   Output packets:                    6                    0 pps
  Frame exceptions:
    Oversized frames                 0
    Errored input frames             0
    Input on disabled link/bundle    0
    Output for disabled link/bundle  0
    Queuing drops                    0
  Buffering exceptions:
    Packet data buffer overflow      0
    Fragment data buffer overflow    0
  Assembly exceptions:
    Fragment timeout                 0
    Missing sequence number          0
    Out-of-order sequence number     0
    Out-of-range sequence number     0
  Hardware errors (sticky):
    Data memory error                0
    Control memory error             0
  Egress queues: 8 supported, 8 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
```

```
  0 q1                              6                   6                   0

  1 q2                              0                   0                   0

  2 assured-forw                    0                   0                   0

  3 q3                              0                   0                   0


Logical interface dl0.0 (Index 66) (SNMP ifIndex 36) (Generation 133)
  Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: Multilink-PPP
  Dialer:
    State: Active, Dial pool: 1
    Dial strings: 384010
    Subordinate interfaces: bc-2/0/0:8 (Index 161), bc-2/0/0:7 (Index 160),
    bc-2/0/0:6 (Index 159), bc-2/0/0:5 (Index 158), bc-2/0/0:4 (Index 157),
    bc-2/0/0:3 (Index 156), bc-2/0/0:2 (Index 155), bc-2/0/0:1 (Index 154)
    Activation delay: 0, Deactivation delay: 0
    Initial route check delay: 120
    Redial delay: 3
    Callback wait period: 5
    Load threshold: 100, Load interval: 60
  Bandwidth: 512kbps
  Bundle options:
    MRRU                                1504
    Remote MRRU                         1504
    Drop timer period                   0
    Inner PPP Protocol field compression enabled
    Sequence number format              long (24 bits)
    Fragmentation threshold             0
    Links needed to sustain bundle      1
    Interleave fragments                Disabled
  Bundle errors:
    Packet drops                    0 (0 bytes)
    Fragment drops                  15827 (759696 bytes)
    MRRU exceeded                   0
    Exception events                0
  Statistics         Frames        fps         Bytes          bps
  Bundle:
    Fragments:
      Input :        963116          0      50963104            0
      Output:             6          0           540            0
    Packets:
      Input :        934346          0      42980144            0
      Output:             6          0           504            0
  Link:
    bc-2/0/0:1.0
      Input :        119656          0       6341806            0
      Output:             1          0            90            0
    bc-2/0/0:2.0
      Input :        120176          0       6369366            0
      Output:             1          0            90            0
    bc-2/0/0:3.0
      Input :        119856          0       6352368            0
      Output:             1          0            90            0
    bc-2/0/0:4.0
      Input :        120315          0       6376695            0
      Output:             0          0             0            0
    bc-2/0/0:5.0
      Input :        120181          0       6369593            0
      Output:             0          0             0            0
```

```
    bc-2/0/0:6.0
      Input :        121154          0       6421200            0
      Output:             0          0             0            0
    bc-2/0/0:7.0
      Input :        121181          0       6340321            0
      Output:             0          0             0            0
    bc-2/0/0:8.0
      Input :        120594          0       6391482            0
      Output:             0          0             0            0
NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls:
Not-configured
  Protocol inet, MTU: 1500, Generation: 138, Route table: 0
    Flags: None
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 1.1.1.0/30, Local: 1.1.1.2, Broadcast: Unspecified,
      Generation: 134
```

Meaning The output shows a summary of dialer interface information. The first example is for ISDN BRI service, and the second example is for ISDN PRI service. Verify the following information:

- The physical interface is Enabled. If the interface is shown as Disabled, do either of the following:
  - In the CLI configuration editor, delete the disable statement at the [edit interfaces *interface-name*] level of the configuration hierarchy.
  - In the J-Web configuration editor, clear the **Disable** check box on the **Interfaces > *interface-name*** page.
- The physical link is Up. A link state of Down indicates a problem with the interface module, interface port, or physical connection (link-layer errors).
- The Last Flapped time is an expected value. The Last Flapped time indicates the last time the physical interface became unavailable and then available again. Unexpected flapping indicates possible link-layer errors.
- Subordinate interfaces correctly lists the B-channel interface or interfaces associated with this dialer interface. The ISDN BRI output in the first example shows that dl0 supports bc-6/0/0:2.

  The ISDN PRI output in the second example shows that dl0 supports bc-2/0/0:1 through bc-2/0/0:8.
- The traffic statistics reflect expected input and output rates. Verify that the number of inbound and outbound bytes and packets matches expected throughput for the physical interface. To clear the statistics and see only new changes, use the clear interfaces statistics *interface-name* command.
- The dialer state is Active when an ISDN call is in progress.
- The LCP state is Opened when an ISDN call is in progress. An LCP state of Closed or Not Configured indicates a problem with the dialer configuration that needs to be debugged with the monitor traffic interface *interface-name* command. For information about the monitor traffic command, see the *JUNOS Software Administration Guide*.

Related Topics For a complete description of show interfaces (ISDN dialer) output, see the *JUNOS Interfaces Command Reference*.

Chapter 9

Configuring USB Modems for Dial Backup

You can configure your device to “fail over” to a USB modem connection when the primary Internet connection experiences interruption.

NOTE: Low-latency traffic such as VoIP traffic is not supported over USB modem connections.

NOTE: We recommend using a Multi-Tech MultiModem MT5634ZBA-USB-V92 USB modem.

You use either the J-Web configuration editor or CLI configuration editor to configure the USB modem for dial backup.

This chapter contains the following topics. For more information about modem initialization, administration, verification, and remote connection, see the *J-series Services Router Administration Guide*.

USB Modem Terms

Before configuring USB modems and their supporting dialer interfaces, become familiar with the terms defined in Table 69 on page 224.

Table 69: USB Modem Terminology

Term	Definition
caller ID	Telephone number of the caller on the remote end of a backup USB modem connection, used to dial in and also to identify the caller. Multiple caller IDs can be configured on a dialer interface. During dial-in, the device matches the incoming call's caller ID against the caller IDs configured on its dialer interfaces. Each dialer interface accepts calls from only callers whose caller IDs are configured on it.
dial backup	Feature that reestablishes network connectivity through one or more backup dialer interfaces after a primary interface fails. When the primary interface is reestablished, the USB modem backup is disconnected.
dialer interface	Logical interface for configuring dialing properties and the control interface for a backup USB modem connection.
dialer profile	Set of characteristics configured for the USB modem dialer interface. Dialer profiles allow the configuration of physical interfaces to be separated from the logical configuration of dialer interfaces required for USB modem connectivity. This feature also allows physical and logical interfaces to be bound together dynamically on a per-connection basis.
dial-in	Feature that enables devices to receive calls from the remote end of a backup USB modem connection. The remote end of the USB modem call might be a service provider, a corporate central location, or a customer premises equipment (CPE) branch office. All incoming calls can be verified against caller IDs configured on the device's dialer interface.

USB Modem Interface Overview

You configure two types of interfaces for USB modem connectivity: a physical interface and a logical interface called the dialer interface:

- The USB modem physical interface uses the naming convention umd0. The device creates this interface when a USB modem is connected to the USB port.
- The dialer interface, dl*n*, is a logical interface for configuring dialing properties for USB modem connections. The dialer interface can be configured using Point-to-Point Protocol (PPP) encapsulation. You can also configure the dialer interface to support authentication protocols—PPP Challenge Handshake (CHAP) or Password Authentication Protocol (PAP).

For information about interface names, see “Interface Naming Conventions” on page 17.

The USB modem provides a dial-in remote management interface, and supports dialer interface features by sharing the same dial pool as a dialer interface. The dial pool allows the logical dialer interface and the physical interface to be bound together dynamically on a per-call basis. You can configure the USB modem to operate either as a dial-in console for management or as a dial-in WAN backup interface.

The following rules apply when you configure dialer interfaces for USB modem connections:

- The dialer interface must be configured to use PPP encapsulation. You cannot configure Cisco High-Level Data Link Control (HDLC) or Multilink PPP (MLPPP) encapsulation on dialer interfaces.
- The dialer interface cannot be configured as a constituent link in a multilink bundle. For information about configuring multilink bundles, see "Configuring Link Services Interfaces" on page 241.
- The dialer interface can perform backup, dialer filter, and dialer watch functions, but these operations are mutually exclusive. You can configure a single dialer interface to operate in only one of the following ways:
  - As a backup interface—for one primary interface
  - As a dialer filter
  - As a dialer watch interface

Before You Begin

Before you configure USB modems, you need to perform the following tasks:

- Install device hardware. For more information, see the Getting Started Guide for your device.
- Establish basic connectivity. For more information, see the Getting Started Guide for your device.
- Order a Multi-Tech MultiModem MT5634ZBA-USB-V92 USB modem from Multi-Tech Systems (http://www.multitech.com/).
- Order a public switched telephone network (PSTN) line from your telecommunications service provider. Contact your service provider for more information.
- If you do not already have a basic understanding of physical and logical interfaces and Juniper Networks interface conventions, read "Interfaces Overview" on page 11.

Connecting the USB Modem to the Device's USB Port

NOTE: J-series devices have two USB ports. However, you can connect only one USB modem to the USB ports on these devices. If you connect USB modems to both ports, the device detects only the first modem connected.

NOTE: When you connect the USB modem to the USB port on the device, the USB modem is initialized with the modem initialization string configured for the USB modem interface on the device. For more information, see the USB modem chapter in the *J-series Services Router Administration Guide*.

To connect the USB modem to the USB port on the device:

1. Plug the modem into the USB port.
2. Connect the modem to your telephone network.

Configuring USB Modems for Dial Backup with a Configuration Editor

To configure USB modem interfaces, perform the following tasks.

- Configuring a USB Modem Interface for Dial Backup on page 226
- Configuring a Dialer Interface for USB Modem Dial Backup on page 227
- Configuring Dial-In for a USB Modem Connection on page 235
- Configuring PAP on Dialer Interfaces (Optional) on page 236
- Configuring CHAP on Dialer Interfaces (Optional) on page 237

Configuring a USB Modem Interface for Dial Backup

To configure a USB modem interface for the device:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 70 on page 226.
3. Go on to "Configuring a Dialer Interface for USB Modem Dial Backup" on page 227.

Table 70: Configuring a USB Modem Interface for Dial Backup

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > Edit Configuration > View and Edit**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces umd0
Create the new interface umd0.	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type the name of the new interface, umd0. 3. Click **OK**.	

Table 70: Configuring a USB Modem Interface for Dial Backup *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure dialer options. ■ Name the dialer pool configured on the dialer interface you want to use for USB modem connectivity—for example, usb-modem-dialer-pool. For more information, see "Configuring a Dialer Interface for USB Modem Dial Backup" on page 227. ■ Set the dialer pool priority—for example, 25. Dialer pool priority has a range from 1 to 255, with 1 designating lowest-priority interfaces and 255 designating the highest-priority interfaces.	1. In the Encapsulation column, next to the new interface, click **Edit**. 2. Next to Dialer options, select **Yes**, and then click **Configure**. 3. Next to Pool, click **Add new entry**. 4. In the Pool identifier box, type usb-modem-dialer-pool. 5. In the Priority box, type 25. 6. Click **OK** until you return to the Interface page.	Enter set dialer-options pool usb-modem-dialer-pool priority 25
Configure the modem to automatically answer (autoanswer) calls after a specified number of rings. **NOTE:** The default modem initialization string is AT S7=45 S0=0 V1 X4 &C1 E0 Q0 &Q8 %C0. The modem command S0=0 disables the modem from autoanswering calls.	1. Next to Modem options, click **Configure**. 2. In the Init command string box, type ATS0=2 \n to configure the modem to autoanswer after two rings.	Enter set modem-options init-command-string "ATS0=2 \n"
Configure the modem to act as a dial-in WAN backup interface.	1. On the Modem options page, in the Dialin box, select **routable**. 2. Click **OK**.	Enter set modem-options dialin routable

Configuring a Dialer Interface for USB Modem Dial Backup

The dialer interface (dl) is a logical interface configured to establish USB modem connectivity. You can configure multiple dialer interfaces for different functions on the device.

After configuring the dialer interface, you must configure a backup method—either dialer backup, a dialer filter, or dialer watch.

For example, suppose you have a branch office router and a head office router each with a USB modem interface and a dialer interface. To establish a backup connection between the branch office and head office routers, you can configure them as described in Table 71 on page 228.

Table 71: Configuring Branch Office and Head Office Routers for USB Modem Backup Connectivity

Router Location	Configuration Requirement	Instructions
Branch Office	1. Configure the logical dialer interface on the branch office router for USB modem dial backup. 2. Configure the dialer interface dl0 in one of the following ways on the branch office router: ■ Configure the dialer interface dl0 as the backup interface on the branch office router's primary T1 interface t1-1/0/0. ■ Configure a dialer filter on the branch office router's dialer interface. ■ Configure a dialer watch on the branch office router's dialer interface.	■ To configure the logical dialer interface, see Table 72 on page 228. ■ To configure dl0 as a backup for t1-1/0/0 see "Configuring Dial Backup for a USB Modem Connection" on page 231. ■ To configure a dialer filter on dl0, see "Configuring a Dialer Filter for USB Modem Dial Backup" on page 231. ■ To configure a dialer watch on dl0, see "Configuring Dialer Watch for USB Modem Dial Backup" on page 233.
Head Office	Configure dial-in on the dialer interface dl0 on the head office router.	To configure dial-in on the head office router, see "Configuring Dial-In for a USB Modem Connection" on page 235.

To configure a logical dialer interface for USB modem dial backup:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 72 on page 228.
3. To configure a backup method, go on to one of the following tasks:
   - Configuring Dial Backup for a USB Modem Connection on page 231
   - Configuring a Dialer Filter for USB Modem Dial Backup on page 231
   - Configuring Dialer Watch for USB Modem Dial Backup on page 233

Table 72: Adding a Dialer Interface for USB Modem Dial Backup

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > Edit Configuration > View and Edit**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces

Table 72: Adding a Dialer Interface for USB Modem Dial Backup *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Create the new interface—for example, dl0. Adding a description can differentiate between different dialer interfaces—for example, USB-modem-backup.	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type dl0. 3. In the Description box, type USB-modem-backup. 4. Click **OK**.	Create and name the interface: 1. edit dl0 2. set description USB-modem-backup
Configure Point-to-Point Protocol (PPP) encapsulation. **NOTE:** You cannot configure Cisco High-Level Data Link Control (HDLC) or Multilink PPP (MLPPP) encapsulation on dialer interfaces used in USB modem connections.	1. In the Encapsulation column, next to the new interface, click **Edit**. 2. From the Encapsulation list, select **ppp**.	Enter set encapsulation ppp
Create the logical unit 0. **NOTE:** You can set the logical unit to 0 only.	1. Next to Unit, click **Add new entry**. 2. In the Interface unit number box, type 0. 3. Next to Dialer options, select **Yes**, and then click **Configure**.	Enter set unit 0

Table 72: Adding a Dialer Interface for USB Modem Dial Backup *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure dialer options. ■ **Activation delay**—Number of seconds to wait before activating the backup USB modem interface after the primary interface is down—for example, 30. The default value is 0 seconds, and the maximum value is 60 seconds. Use this option only for dialer backup and dialer watch. ■ **Deactivation delay**—Number of seconds to wait before deactivating the backup USB modem interface after the primary interface is up—for example, 30. The default value is 0 seconds, and the maximum value is 60 seconds. Use this option only for dialer backup and dialer watch. ■ **Idle timeout**—Number of seconds a connection is idle before disconnecting—for example, 30. The default value is 120 seconds, and the range is from 0 to 4294967295. ■ **Initial route check**—Number of seconds to wait before checking if the primary interface is up—for example, 30. The default value is 120 seconds, and the range is from 1 to 300 seconds. ■ **Pool**—Name of the dialer pool to use for USB modem connectivity—for example, usb-modem-dialer-pool.	1. In the Activation delay box, type 60. 2. In the Deactivation delay box, type 30. 3. In the Idle timeout box, type 30. 4. In the Initial route check box, type 30. 5. In the Pool box, type usb-modem-dialer-pool.	1. Enter edit unit 0 dialer-options 2. Enter set activation-delay 60 3. Enter set deactivation-delay 30 4. Enter set idle-timeout 30 initial-route-check 30 pool usb-modem-dialer-pool
Configure the telephone number of the remote destination to call if the primary interface goes down—for example, 5551212.	1. Next to Dial string, click **Add new entry**. 2. In the Dial string box, type 5551212. 3. Click **OK** until you return to the Unit page.	Enter set dial-string 5551212
Configure source and destination IP addresses for the dialer interface—for example, 172.20.10.2 and 172.20.10.1. **NOTE:** If you configure multiple dialer interfaces, ensure that the same IP subnet address is not configured on different dialer interfaces. Configuring the same IP subnet address on multiple dialer interfaces can result in inconsistency in the route and packet loss. Packets can be routed through any of the dialer interfaces with the IP subnet address, instead of being routed through the dialer interface to which the USB modem call is mapped.	1. Select **Inet** under Family, and click **Edit**. 2. Next to Address, click **Add new entry**. 3. In the Source box, type 172.20.10.2. 4. In the Destination box, type 172.20.10.1. 5. Click **OK**.	1. From the [edit] hierarchy level, enter edit interfaces dl0 unit 0 2. Enter set family inet address 172.20.10.2 destination 172.20.10.1

Configuring Dial Backup for a USB Modem Connection

Dial backup allows one or more dialer interfaces to be configured as the backup link for the primary serial interface. The backup dialer interfaces are activated only when the primary interface fails. USB modem backup connectivity is supported on all interfaces except ls-0/0/0.

To configure a primary interface for backup connectivity:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 73 on page 231.
3. If you are finished configuring the device, commit the configuration.

Table 73: Configuring a Primary Interface for USB Modem Dial Backup

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > Edit Configuration > View and Edit**. 2. Next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces t1-1/0/0 unit 0
Select the physical interface for USB modem USB modem backup connectivity—for example, t1–1/0/0.	1. In the Interface name column, click the physical interface name. 2. Under Unit, in the Interface unit number column, click **0**.	
Configure the backup dialer interface—for instance, dl0.0.	1. Next to Backup options, click **Configure**. 2. In the Interface box, type dl0.0. 3. Click **OK** until you return to the Interfaces page.	Enter set backup-options interface dl0.0

Configuring a Dialer Filter for USB Modem Dial Backup

This dial-on-demand routing backup method allows a USB modem connection to be activated only when network traffic configured as an “interesting packet” arrives on the network. Once the network traffic is sent, an inactivity timer is triggered and the connection is closed.

You define an interesting packet using the dialer filter feature of the device.

To configure dial-on-demand routing backup using a dialer filter, you first configure the dialer filter and then apply the filter to the dialer interface.

To configure the dialer filter and apply it to the dialer interface:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 74 on page 232.
3. Go on to Table 75 on page 233.
4. When you are finished configuring the device, commit the configuration.

Table 74: Configuring a Dialer Filter for USB Modem Dial Backup

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Firewall** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > Edit Configuration > View and Edit**. 2. Next to Firewall, click **Edit**.	From the [edit] hierarchy level, enter edit firewall
Configure the dialer filter name—for example, interesting-traffic.	1. Next to Inet, click **Configure** or **Edit**. 2. Next to Dialer filter, click **Add new entry**. 3. In the Filter name box, type interesting-traffic.	1. Enter edit family inet 2. Then enter edit dialer-filter interesting-traffic
Configure the dialer filter rule name—for example, term1. Configure term behavior. For example, you might want to configure the dialer filter to allow only traffic between the branch office router and the head office router over the backup USB modem connection. In this example, the branch office router has the IP address 20.20.90.4/32 and the head office router has the IP address 200.200.201.1/32. To configure the term completely, include both from and then statements.	1. Next to Term, click **Add new entry**. 2. In the Rule name box, type term1. 3. Next to From, click **Configure**. 4. Next to Source address, click **Add new entry**. 5. In the Address box, type 20.20.90.4/32. 6. Click **OK**. 7. Next to Destination address, click **Add new entry**. 8. In the Address box, type 200.200.201.1/32. 9. Click **OK** until you return to the Term page.	1. Enter edit term term1 2. Enter set from source-address 20.20.90.4/32 3. Enter set from destination-address 200.200.201.1/32
Configure the then part of the dialer filter to discard Telnet traffic between the branch office router and the head office router.	1. Next to Then, click **Configure**. 2. From the Designation list, select **Note**. 3. Click **OK**.	Enter set then note

Table 75: Applying the Dialer Filter to the Dialer Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > Edit Configuration > View and Edit**. 2. Next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces dl0 unit 0
Select the dialer interface to apply the filter—for example, dl0.	1. In the Interface name column, click **dl0**. 2. Under Unit, in the Interface unit number column, click **0**.	
Apply the dialer filter to the dialer interface.	1. In the Family section, next to Inet, click **Edit**. 2. Next to Filter, click **Configure**. 3. In the Dialer box, type interesting-traffic, the dialer filter configured in "Configuring the Dialer Filter" on page 196. 4. Click **OK**.	1. Enter edit family inet filter 2. Enter set dialer interesting-traffic

Configuring Dialer Watch for USB Modem Dial Backup

Dialer watch is a backup method that integrates backup dialing with routing capabilities and provides reliable connectivity without relying on a dialer filter to trigger outgoing USB modem connections. With dialer watch, the device monitors the existence of a specified route and if the route disappears, the dialer interface initiates the USB modem connection as a backup connection.

In this example, you configure dialer watch to enable the device to monitor the existence of the route to the head office router and initiate USB modem backup connectivity if the route disappears.

To configure dialer watch, you first add a dialer watch interface and then configure the USB modem interface to participate as a dialer watch interface.

To configure a dialer watch:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 76 on page 234.
3. Go on to Table 77 on page 234.
4. When you are finished configuring the device, commit the configuration.

Table 76: Adding a Dialer Watch Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > Edit Configuration > View and Edit**. 2. Next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces
Select a dialer interface—for example, dl0. Adding a description, such as dialer-watch, can help you identify one dialer interface from another.	1. Under Interface name, select **dl0**. 2. In the Description box, type dialer-watch.	1. Enter edit dl0 2. Enter set description dialer-watch
On a logical interface—for example, 0—configure the route to the head office router for dialer watch—for example, 200.200.201.1/32.	1. Under Unit, click the logical unit number 0. 2. Next to Dialer options, click **Edit**. 3. Next to Watch list, click **Add new entry**. 4. In the Prefix box, type 200.200.201.1/32. 5. Click **OK**.	1. Enter edit unit 0 dialer-options 2. Enter set watch-list 200.200.201.1/32
Configure the name of the dialer pool to use for dialer watch—for example, dw-pool.	1. In the Pool box, type dw-pool. 2. Click **OK**.	Enter set pool dw-pool

Table 77: Configuring a USB Modem Interface for Dialer Watch

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy, and select the USB modem physical interface umd0.	1. In the J-Web interface, select **Configuration > Edit Configuration > View and Edit**. 2. Next to Interfaces, click **Edit**. 3. Under Interface name, click **umd0**.	From the [edit] hierarchy level, enter edit interfaces umd0 dialer-options pool dw-pool
Configure dialer watch options for the USB modem interface participating in the dialer watch. The USB modem interface must have the same pool identifier to participate in dialer watch. Therefore, the dialer pool name dw-pool, for the dialer watch interface configured in Table 76 on page 234, is used when configuring the USB modem interface.	1. Next to Dialer options, click **Edit**. 2. Next to Pool, click **Add new entry**. 3. In the Pool identifier box, type dw-pool. 4. Click **OK**.	

Configuring Dial-In for a USB Modem Connection

You can configure a dialer interface to accept all incoming calls or accept only calls from one or more caller IDs.

If the dialer interface is configured to accept only calls from a specific caller ID, the device matches the incoming call's caller ID against the caller IDs configured on its dialer interfaces. If an exact match is not found and the incoming call's caller ID has more digits than the configured caller IDs, the device performs a right-to-left match of the incoming call's caller ID with the configured caller IDs and accepts the incoming call if a match is found. For example, if the incoming call's caller ID is 4085321091 and the caller ID configured on a dialer interface is 5321091, the incoming call is accepted. Each dialer interface accepts calls from only callers whose caller IDs are configured on it.

To configure a dialer interface for USB modem dial-in:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 78 on page 235.
3. If you are finished configuring the device, commit the configuration.

Table 78: Configuring the Dialer Interface for USB Modem Dial-In

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy, and select a dialer interface—for example, dl0.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Edit**. 3. Next to dl0, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces dl0

Table 78: Configuring the Dialer Interface for USB Modem Dial-In *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
On logical interface 0, configure the incoming map options for the dialer interface. ■ accept-all—Dialer interface accepts all incoming calls. You can configure the accept-all option for only one of the dialer interfaces associated with a USB modem physical interface. The dialer interface with the accept-all option configured is used only if the incoming call's caller ID does not match the caller IDs configured on other dialer interfaces. ■ caller—Dialer interface accepts calls from a specific caller ID—for example, 4085551515. You can configure a maximum of 15 caller IDs per dialer interface. The same caller ID must not be configured on different dialer interfaces. However, you can configure caller IDs with more or fewer digits on different dialer interfaces. For example, you can configure the caller IDs 14085551515, 4085551515, and 5551515 on different dialer interfaces.	1. In the Unit section, for logical unit number 0, click **Dialer options** under Encapsulation. 2. Next to Incoming map, click **Configure**. 3. From the Caller type menu, select **Caller**. 4. Next to Caller, click **Add new entry**. 5. In the Caller id box, type 4085551515.	1. Enter edit unit 0 2. Enter edit dialer-options 3. Enter set incoming-map caller 4085551515

Configuring PAP on Dialer Interfaces (Optional)

You can configure dialer interfaces to support the Password Authentication Protocol (PAP). PAP allows a simple method for a peer to establish its identity using a two-way handshake during initial link establishment. After the link is established, an ID and password pair is repeatedly sent by the peer to the authenticator until authentication is acknowledged or the connection is terminated.

For more information about PAP, see the *JUNOS Network Interfaces Configuration Guide*.

To configure PAP on the dialer interface, create an access profile and then configure the dialer interface:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 79 on page 237.
3. If you are finished configuring the device, commit the configuration.

Table 79: Configuring PAP on Dialer Interfaces

Task	J-Web Configuration Editor	CLI Configuration Editor
Define a PAP access profile—for example, pap-access-profile with a client (username) named pap-access-user and the PAP password my-pap.	1. On the main Configuration page next to Access, click **Configure** or **Edit**. 2. Next to Profile, click **Add new entry**. 3. In the Profile name box, type pap-access-profile. 4. Next to Client, click **Add new entry**. 5. In the Name box, type pap-access-user. 6. In the Pap-password box, type my-pap. 7. Click **OK** until you return to the main Configuration page.	From the [edit] hierarchy level, enter set access profile pap-access-profile client pap-access-user pap-password my-pap
Navigate to the appropriate dialer interface level in the configuration hierarchy—for example, dl0 unit 0.	1. On the main Configuration page next to Interfaces, click **Configure** or **Edit**. 2. In the interface name box, click **dl0**. 3. In the Interface unit number box, click **0**.	From the [edit] hierarchy level, enter edit interfaces dl0 unit 0
Configure PAP on the dialer interface and specify the local name and password—for example, pap-access-profile and my-pap.	1. Next to Ppp options, click **Configure**. 2. Next to Pap, click **Configure**. 3. In the Local name box, type pap-access-profile. 4. In the Local password box, type my-pap. 5. Click **OK**.	Enter set ppp-options pap local-name pap-access-user local-password my-pap

Configuring CHAP on Dialer Interfaces (Optional)

You can optionally configure dialer interfaces to support the PPP Challenge Handshake Authentication Protocol (CHAP). CHAP is a server-driven, three-step authentication method that depends on a shared secret password residing on both the server and the client. When you enable CHAP on a dialer interface, the device can authenticate its peer and be authenticated by its peer.

For more information about CHAP, see the *JUNOS Network Interfaces Configuration Guide*.

To configure CHAP on the dialer interface:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 80 on page 239.
3. If you are finished configuring the device, commit the configuration.

Table 80: Configuring CHAP on Dialer Interfaces

Task	J-Web Configuration Editor	CLI Configuration Editor
Define a CHAP access profile—for example, usb-modem-access-profile with a client (username) named usb-modem-user and the secret (password) my-secret.	1. On the main Configuration page next to Access, click **Configure** or **Edit**. 2. Next to Profile, click **Add new entry**. 3. In the Profile name box, type usb-modem-access-profile. 4. Next to Client, click **Add new entry**. 5. In the Name box, type usb-modem-user. 6. In the Chap secret box, type my-secret. 7. Click **OK** until you return to the main Configuration page.	From the [edit] hierarchy level, enter set access profile usb-modem-access-profile client usb-modem-user chap-secret my-secret
Navigate to the appropriate dialer interface level in the configuration hierarchy—for example, dl0 unit 0.	1. On the main Configuration page next to Interfaces, click **Configure** or **Edit**. 2. In the interface name box, click **dl0**. 3. In the Interface unit number box, click **0**.	From the [edit] hierarchy level, enter edit interfaces dl0 unit 0
Configure CHAP on the dialer interface and specify a unique profile name containing a client list and access parameters—for example, usb-modem-access-profile.	1. Next to Ppp options, click **Configure**. 2. Next to Chap, click **Configure**. 3. In the Access profile box, type usb-modem-access-profile. 4. Click **OK**.	Enter set ppp-options chap access-profile usb-modem-access-profile

Chapter 10

Configuring Link Services Interfaces

Link services include the multilink services Multilink Point-to-Point Protocol (MLPPP), Multilink Frame Relay (MLFR), and Compressed Real-Time Transport Protocol (CRTP). J-series devices support link services on the ls-0/0/0 link services interface.

You can use either J-Web Quick Configuration or a configuration editor to configure the link services interface.

This chapter contains the following topics:

Link Services Terms

Before configuring a link services interface, become familiar with the terms defined in Table 81 on page 241.

Table 81: Link Services Terminology

Term	Definition
Compressed Real-Time Transport Protocol (CRTP)	Protocol defined in RFC 2508 that compresses the size of IP, UDP, and Real-Time Transport Protocol (RTP) headers and works with reliable and fast point-to-point links for voice over IP (VoIP) traffic.
data-link connection identifier (DLCI)	Identifier for a Frame Relay virtual connection, also called a logical interface.
link fragmentation and interleaving (LFI)	For MLFR with Frame Relay traffic or MLPPP with PPP traffic, a method of reducing excessive delays by fragmenting long packets into smaller packets and interleaving them with real-time frames. For example, short delay-sensitive packets, such as those of packetized voice, can race ahead of larger delay-insensitive packets, such as common data packets.

Table 81: Link Services Terminology *(continued)*

Term	Definition
link services	Capabilities on an interface that use Multilink Frame Relay (MLFR) and Multilink Point-to-Point Protocol (MLPPP), link fragmentation and interleaving (LFI), Compressed Real-Time Transport Protocol (CRTP), and certain class-of-service (CoS) components to improve packet transmission, especially for time-sensitive voice packets.
Multilink Frame Relay (MLFR)	Protocol that allows multiple Frame Relay links to be aggregated by inverse multiplexing.
Multilink Point-to-Point Protocol (MLPPP)	Protocol that allows you to bundle multiple Point-to-Point Protocol (PPP) links into a single logical unit. MLPPP improves bandwidth efficiency and fault tolerance and reduces latency.
Point-to-Point Protocol (PPP)	Link-layer protocol defined in RFC 1661 that provides multiprotocol encapsulation. PPP is used for link-layer and network-layer configuration.
shaping rate	In class of service (CoS) classification, a method of controlling the maximum rate of traffic transmitted on an interface.

Link Services Interfaces Overview

You configure the link services interface (ls-0/0/0) on a J-series device to support multilink services and Compressed Real-Time Transport Protocol (CRTP).

The link services interface on a J-series Services Router consists of services provided by the following interfaces on the Juniper M-series and T-series routing platforms: multilink services interface (ml-*fpc/pic/port*), link services interface (ls-*fpc/pic/port*), and link services intelligent queuing interface (lsq-*fpc/pic/port*). Although the multilink services, link services, and link services intelligent queuing (IQ) interfaces on M-series and T-series routing platforms are installed on Physical Interface Cards (PICs), the link services interface on a J-series Services Router is an internal interface only and is not associated with a physical medium or Physical Interface Module (PIM).

For information about interface names, see "Network Interface Naming" on page 16.

For more information about the link services interfaces, see the *JUNOS Services Interfaces Configuration Guide*.

This section contains the following topics.

- Load Balancing with LFI on page 248
- Configuring CoS Components with LFI on page 249

Services Available on J-series Link Services Interface

On a J-series device, the link services interface is a logical interface available by default. Table 82 on page 243 summarizes the services available on a J-series link services interface.

Table 82: Services Available on J-series Link Services Interface

Services	Purpose	More Information
Multilink bundles by means of MLPPP and MLFR encapsulation	Aggregates multiple constituent links into one larger logical bundle to provide additional bandwidth, load balancing, and redundancy.	■ Configuring an MLPPP Bundle on page 254 ■ Configuring MLFR FRF.15 Bundles on page 264 ■ Configuring MLFR FRF.16 Bundles on page 267
Link fragmentation and interleaving (LFI)	Reduces delay and jitter on links by breaking up large data packets and interleaving delay-sensitive voice packets with the resulting smaller packets.	"Link Fragmentation and Interleaving Overview" on page 245
Compressed Real-Time Transport Protocol (CRTP)	Reduces the overhead caused by Real-Time Transport Protocol (RTP) on voice and video packets.	"Compressed Real-Time Transport Protocol Overview" on page 246
Class-of-service (CoS) classifiers, forwarding classes, schedulers and scheduler maps, and shaping rates	Provide a higher priority to delay-sensitive packets—by configuring class of service (CoS) components, such as the following: ■ Classifiers—To classify different type of traffic, such as voice, data and network control packets ■ Forwarding classes—To direct different types of traffic to different output queues ■ Schedulers and scheduler maps—To define properties for the output queues such as delay-buffer, transmission rate, and transmission priority ■ Shaping rate—To define certain bandwidth usage by an interface	■ Defining Classifiers and Forwarding Classes on page 257 ■ Defining and Applying Scheduler Maps on page 259 ■ Applying Shaping Rates to Interfaces on page 263 ■ Class-of-Service Overview on page 553 ■ Configuring Class of Service on page 579

Link Services Exceptions on J-series Services Routers

The link and multilink services implementation on a J-series Services Router is similar to the implementation on the M-series and T-series routing platforms, with the following exceptions:

- J-series devices support link and multilink services on the ls-0/0/0 interface instead of the ml-*fpc/pic/port*, lsq-*fpc/pic/port*, and ls-*fpc/pic/port* interfaces.
- When LFI is enabled, Queue 2 is reserved for voice traffic, while all other queues perform fragmentation. Also, the queuing behavior on the link services interface and constituent links is different. For more information, see "Queuing with LFI on J-series Devices" on page 246.
- When LFI is enabled, fragmented packets are queued in a round-robin fashion on the constituent links to enable per-packet and per-fragment load balancing. For more information, see "Queuing with LFI on J-series Devices" on page 246.
- J-series devices support per-unit scheduling on all types of constituent links (on all types of interfaces).
- J-series devices support Compressed Real-Time Transport Protocol (CRTP) with MLPPP as well as PPP.
- J-series devices do not support multiclass MLPPP.
- J-series devices do not have the ability to apply fragmentation maps to specific queues to enable LFI on specific queues (a multiclass MLPPP feature).

Multilink Bundles Overview

The J-series device supports MLPPP and MLFR multilink encapsulations. MLPPP enables you to bundle multiple PPP links into a single multilink bundle, and MLFR enables you to bundle multiple Frame Relay data-link connection identifiers (DLCIs) into a single multitlink bundle. Multilink bundles provide additional bandwidth, load balancing, and redundancy by aggregating low-speed links, such as T1, E1, and serial links.

You configure multilink bundles as logical units or channels on the link services interface ls-0/0/0:

- With MLPPP and MLFR FRF.15, multilink bundles are configured as logical units on ls-0/0/0—for example, ls-0/0/0.0 and ls-0/0/0.1.
- With MLFR FRF.16, multilink bundles are configured as channels on ls-0/0/0—for example, ls-0/0/0:0 and ls-0/0/0:1.

After creating multilink bundles, you add constituent links to the bundle. The constituent links are the low-speed physical links that are to be aggregated. You can create 64 multilink bundles, and on each multilink bundle you can add up to 8 constituent links. The following rules apply when you add constituent links to a multilink bundle:

- On each multilink bundle, add only interfaces of the same type. For example, you can add either T1 or E1, but not both.
- Only interfaces with a PPP encapsulation can be added to an MLPPP bundle, and only interfaces with a Frame Relay encapsulation can be added to an MLFR bundle.
- If an interface is a member of an existing bundle and you add it to a new bundle, the interface is automatically deleted from the existing bundle and added to the new bundle.

For information about configuring MLPPP bundles, see “Configuring an MLPPP Bundle” on page 254. For information about configuring MLFR bundles, see “Configuring MLFR FRF.15 Bundles” on page 264 and “Configuring MLFR FRF.16 Bundles” on page 267.

Link Fragmentation and Interleaving Overview

As it does on any other interface, priority scheduling on a multilink bundle determines the order in which an output interface transmits traffic from an output queue. The queues are serviced in a weighted round-robin fashion. But when a queue containing large packets starts using the multilink bundle, small and delay-sensitive packets must wait their turn for transmission. Because of this delay, some slow links, such as T1 and E1, can become useless for delay-sensitive traffic.

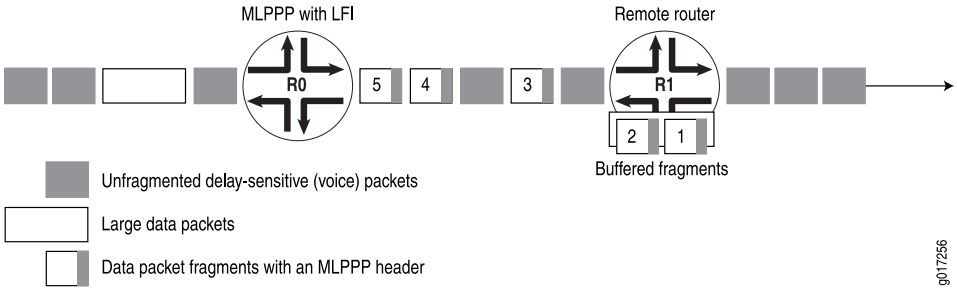
Link fragmentation and interleaving (LFI) solves this problem. It reduces delay and jitter on links by fragmenting large packets and interleaving delay-sensitive packets with the resulting smaller packets for simultaneous transmission across multiple links of a multilink bundle.

Figure 30 on page 245 illustrates how LFI works. In this figure, Device R0 and Device R1 have LFI enabled. When Device R0 receives large and small packets, such as data and voice packets, it divides them into two categories. All voice packets and any other packets configured to be treated as voice packets, such as CRTP packets, are categorized as LFI packets and transmitted without fragmentation or an MLPPP header. The remaining non-LFI (data) packets can be fragmented or unfragmented based on the configured fragmentation threshold. The packets larger than the fragmentation threshold are fragmented. An MLPPP header (containing a multilink sequence number) is added to all non-LFI packets, fragmented and unfragmented.

The fragmentation is performed according to the fragmentation threshold that you configure. For example, if you configure a fragmentation threshold of 128 bytes, all packets larger than 128 bytes are fragmented. When Device R1 receives the packets, it sends the unfragmented voice packets immediately but buffers the packet fragments until it receives the last fragment for a packet. In this example, when Device R1 receives fragment 5, it reassembles the fragments and transmits the whole packet.

The unfragmented data packets are treated as a single fragment. Thus Device R1 does not buffer the unfragmented data packets and transmits them as it receives them.

Figure 30: LFI on a Services Router



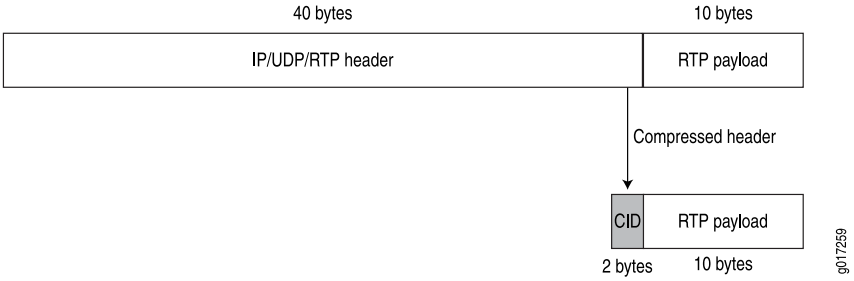
For information about configuring LFI, see “Enabling Link Fragmentation and Interleaving” on page 256.

Compressed Real-Time Transport Protocol Overview

Real-Time Transport Protocol (RTP) can help achieve interoperability among different implementations of network audio and video applications. However, in some cases, the header, which includes the IP, UDP, and RTP headers, can be too large (around 40 bytes) on networks using low-speed lines such as dial-up modems. Compressed Real-Time Transport Protocol (CRTP) can be configured to reduce network overhead on low-speed links. CRTP replaces the IP, UDP, and RTP headers with a 2-byte context ID (CID), reducing the header overhead considerably.

Figure 31 on page 246 shows how CRTP compresses the RTP headers in a voice packet and reduces a 40-byte header to a 2-byte header.

Figure 31: CRTP



On J-series devices, you can configure CRTP with MLPPP or PPP logical interface encapsulation on link services interfaces. For more information about configuring MLPPP, see “Configuring an MLPPP Bundle” on page 254.

When you configure CRTP, link fragmentation and interleaving (LFI) is automatically enabled. Real-time and non-real-time data frames are carried together on lower-speed links without causing excessive delays to the real-time traffic. For more information about LFI, see “Link Fragmentation and Interleaving Overview” on page 245.

Queuing with LFI on J-series Devices

When LFI is enabled, all large packets are fragmented. These packet fragments have a multilink header that contains a multilink sequence number. The sequence numbers on the fragments must be preserved so that the remote device receiving these fragments can correctly reassemble them into a complete packet. To accommodate this requirement, the software queues all fragmented packets on constituent links of a multilink bundle to a single queue (Q0), by default.

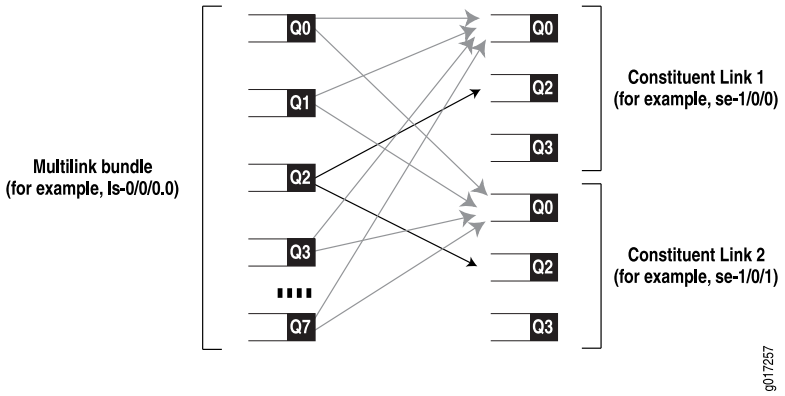
Although they are not fragmented, data packets smaller than the fragmentation threshold are also queued to Q0.

When you configure CRTP with LFI, CRTP packets on a multilink bundle from queues other than Q2 are queued to Q2 (instead of Q0) on the constituent links. Because

CRTP packets are compressed and do not require fragmentation, they are treated as LFI (voice) packets and are sent to Q2 on the constituent links.

Figure 32 on page 247 shows how traffic is queued on an MLPPP or MLFR multilink bundle and its constituent links. Irrespective of the packet queuing on the multilink bundle, the packets on the constituent links are queued according to the default setting so that traffic from all queues except Q2 is mapped to Q0.

Figure 32: Queuing on Constituent Links



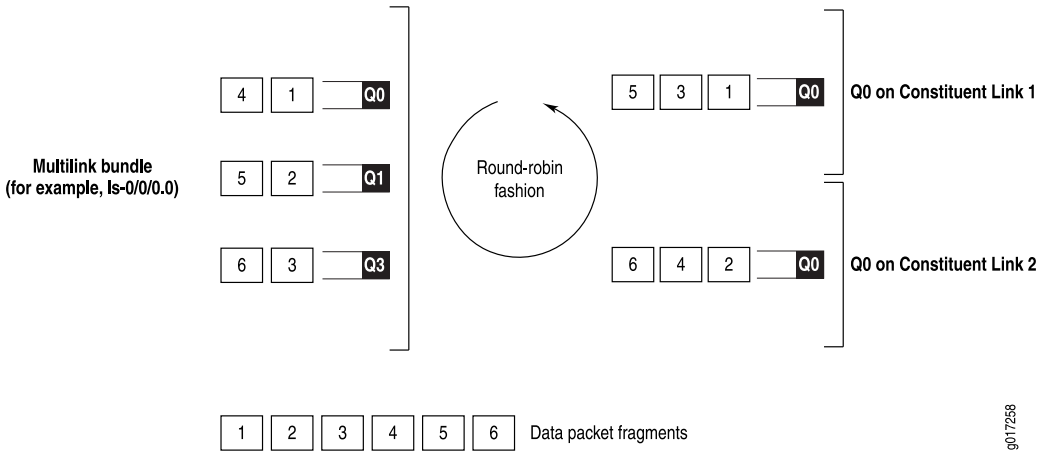
- The packet fragments on Q0, Q1, Q3, Q4, Q5, Q6, and Q7 from the multilink bundle are mapped to Q0 on Constituent Links 1 and 2.
- The LFI packets (such as voice) on Q2 from the multilink bundle are mapped to Q2 on the constituent links.
- The network control packets on Q3 from the multilink bundle are mapped to Q0 on the constituent links. However, Q3 on the constituent links transmits network control packets that exchange protocol information related to constituent links—for example, packets exchanging hello messages on constituent links.

Queuing on Q0s of Constituent Links

On a multilink bundle, packet fragments from all queues except Q2 are transmitted to Q0 on constituent links. On the Q0s of constituent links, the packets are queued in a weighted round-robin fashion to enable per-fragment load balancing.

Figure 33 on page 248 shows how queuing is performed on the constituent links.

Figure 33: Queuing on Q0 of Constituent Links



Packet fragments from the multilink bundle are queued to constituent links one by one in a weighted round-robin fashion. Packet 1 from Q0 on the multilink bundle is queued to Q0 on Constituent Link 1, packet 2 from Q1 on the multilink bundle is queued to Q0 on Constituent Link 2, packet 3 from Q3 on the multilink bundle is queued to Q0 on Constituent Link 1,and so on.

Queuing on Q2s of Constituent Links

On a multilink bundle, all Q2 traffic (LFI traffic) from the multilink bundle is queued to Q2 of constituent links based on a hash computed from the source address, destination address, and IP protocol of the packet. If the IP payload is TCP or UDP traffic, the hash also includes the source port and destination port. As a result of this hash algorithm, all traffic belonging to one traffic flow is queued to Q2 of one constituent link.

Load Balancing with LFI

On link services interfaces, the traffic load is queued and balanced differently for LFI (voice and CRTP packets) and non-LFI packets (data packets) depending on the protocols configured.

Table 83 on page 249 compares queuing and load balancing for LFI and non-LFI packets when MLPPP is configured with LFI and CRTP.

Table 83: LFI Queuing and Load Balancing for Different Protocols

Packet Type	Queuing (MLPPP with LFI)	Queuing (MLPPP with CRTP)	Load Balancing
LFI (voice and CRTP) packets	All incoming packets on Q2 are treated as LFI packets	The following types of incoming packets are treated as LFI packets: ■ Packets matching Q2 (default) ■ Packets from ports configured as LFI ports ■ Packets to queues other than Q2 that are configured as LFI queues **NOTE:** When CRTP is configured without MLPPP traffic traverses only one link thus no load balancing is performed.	Traffic is divided into individual traffic flows, and packets belonging to a flow traverse a single link to avoid packet-ordering issues. The link is selected based on a hash computed from the source address, destination address, and protocol. If the IP payload is TCP or UDP traffic, the hash also includes the source port and destination port.
Non-LFI (data) packets	All data packets, whether fragmented or not, are treated as non-LFI packets and queued to the Q0s of constituent links. (Packets smaller than the size specified in the fragmentation threshold are not fragmented but are treated as non-LFI packets.)	The following types of packets are treated as non-LFI packets and are queued to the Q0s of constituent links: ■ Packets not matching Q2 ■ Packets from ports not configured as LFI ports ■ Packets queued to queues not configured for LFI ■ Packets that are not CRTP packets	All non-LFI packets are queued to the Q0s of constituent links one by one in weighted round-robin fashion.

Configuring CoS Components with LFI

If you configure CoS components with LFI on a J-series device, we recommend that you follow certain recommendations for shaping rate, scheduling priority, and buffer size. For configuration instructions, see "Configuring MLPPP Bundles and LFI on Serial Links" on page 253. For more information about other CoS components, see "JUNOS CoS Components" on page 557.

Shaping Rate

When you configure LFI, we recommend that you configure the shaping rate on each constituent link of the multilink bundle. Shaping rate configuration on the constituent links is required to limit the jitter on the LFI queue. If you anticipate no delay-sensitive or jitter-sensitive traffic on the LFI queue, or if there is no LFI traffic at all, shaping rate configuration is optional.

For information about how to configure a shaping rate, see "Applying Shaping Rates to Interfaces" on page 263.

Scheduling Priority

J-series devices support per-unit scheduling that allows you to configure scheduler maps on each MLPPP or MLFR multilink bundle. You can also configure scheduler maps on constituent links, but you must maintain the same relative priority on the constituent links and on the multilink bundle.

Table 84 on page 250 shows an example of correct and incorrect relative priorities on a multilink bundle and its constituent link. In this example, you have assigned a high priority to LFI packets and a low priority to data packets on the multilink bundle. To maintain the relative priority on the constituent links, you can assign a high priority to the LFI packets and a medium-high priority to the data packets, but you cannot assign a medium-high priority to LFI packets and a high priority to data packets.

Table 84: Relative Priorities on Multilink Bundles and Constituent Links

Multilink Bundle	Correct Constituent Link Priorities	Incorrect Constituent Link Priorities
LFI packets—High priority	LFI packets—High priority	LFI packet—Medium-high priority
Data packets—Low priority	Data packets—Medium-high priority	Data packets—High priority

Buffer Size

All non-LFI traffic from the multilink bundle (from different queues) is transmitted to Q0 on the constituent links. On the constituent links, you must configure a large buffer size for Q0. If the Q0 buffer size on a constituent link is insufficient, the scheduler might drop overflowing packets.

Before You Begin

Before you configure a link services interface, you need to perform the following tasks:

- Install device hardware. For more information, see the *J-series Services Routers Hardware Guide*.
- Establish basic connectivity. For more information, see the Getting Started Guide for your device.
- If you do not already have a basic understanding of physical and logical interfaces and Juniper Networks interface conventions, read “Interfaces Overview” on page 11.

Although it is not a requirement, you might also want to plan how you are going to use the link services interface on your network before you begin configuring it. Read “Link Services Interfaces Overview” on page 242 for a basic understanding of the link services interface implementation.

Configuring the Link Services Interface with Quick Configuration

You can use the services interfaces Quick Configuration pages to do the following:

- Configure the ls-0/0/0 link services interface.
- Configure multilink logical interfaces on the ls-0/0/0 interface. Multilink logical interfaces allow you to bundle multiple serial interfaces such as T1, T3, E1, E3, and serial interfaces into a single logical link as follows:
  - Bundle multiple Point-to-Point Protocol (PPP) links into a single Multilink Point-to-Point Protocol (MLPPP) logical link.
  - Bundle multiple Frame Relay data-link connection identifiers (DLCIs) into a single Multilink Frame Relay (MLFR) logical link.

To configure the link services interface:

1. From the Quick Configuration page, as shown in Figure 12 on page 74, select the link services interface—for example, ls-0/0/0—you want to configure.
2. Enter information into the Quick Configuration page, as described in Table 85 on page 251.
3. Click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration page, click **Apply**.
   - To apply the configuration and return to the main configuration page, click **OK**.
   - To cancel your entries and return to the main page, click **Cancel**.

Table 85: Link Services Interface Quick Configuration Summary

Field	Function	Your Action
Logical Interfaces		
Add logical interfaces	Defines one or more logical units that you connect to this link services interface. You must define at least one logical unit for the link services interface.	Click **Add**.
Logical Interface Description	(Optional) Describes the logical interface.	Type a text description of the logical interface to more clearly identify it in monitoring displays.
IPv4 Addresses and Prefixes	Specifies one or more IPv4 addresses for the interface.	1. Type one or more IPv4 addresses and prefixes. For example: 10.10.10.10/24 2. Click **Add**. 3. Click **OK**.

Table 85: Link Services Interface Quick Configuration Summary *(continued)*

Field	Function	Your Action
Physical Interface Description	(Optional) Adds supplementary information about the physical link services interface.	Type a text description of the link services interface to more clearly identify it in monitoring displays.
Enable subunit queuing	Enables or disables subunit queuing on Frame Relay or VLAN IQ interfaces.	■ To enable subunit queuing, select the check box. ■ To disable subunit queuing, clear the check box.
Multilink Bundle Options		
Bandwidth	Specifies the informational-only bandwidth value for the logical interface.	Type the value.
Drop Timer Period	Specifies a drop timeout value (in milliseconds) to provide a recovery mechanism if individual links in the multilink bundle drop one or more packets. **NOTE:** Ensure that the value you specify is larger than the expected differential delay across the links, so that the timeout period does not elapse under normal jitter conditions, but only when there is actual packet loss.	Type a value between 0 and 2000.
Encapsulation	Specifies the encapsulation type for which you want to create a multilink bundle.	From the list, select one of the following: ■ multilink-ppp—Creates a Multilink Point-to-Point Protocol (MLPPP) bundle. ■ multilink-frame-relay-end-to-end—Creates a Multilink Frame Relay (MLFR) bundle.
Fragmentation Threshold	Specifies the maximum size, in bytes, for multilink packet fragments.	Type a value that is a multiple of 64 bytes between 64 and 16320—for example, 1024.
Links needed to sustain bundle	Specifies the minimum number of links required to sustain the multilink bundle.	Type a value between 1 and 8.
MRRU	Specifies the maximum packet size, in bytes, that the multilink interface can process.	Type a value between 1500 and 4500.
Short Sequence	Sets the length of the packet sequence identification number to 12 bits.	Select this check box.

Table 85: Link Services Interface Quick Configuration Summary *(continued)*

Field	Function	Your Action
Member Interfaces	Specifies the interfaces that are members of the multilink bundle. The Logical Interfaces list displays all the serial interfaces on the device. The Member Interfaces list displays the interfaces that are members of the multilink bundle. The following rules apply when you add interfaces to a multilink bundle: ■ Only interfaces of the same type can be added to a multilink bundle. For example, a T1 and an E1 interface cannot be added to the same bundle. ■ Only interfaces with the PPP encapsulation can be added to an MLPPP bundle and interfaces with the Frame Relay encapsulation can be added to an MLFR bundle. ■ If you add an interface that is a member of an existing bundle, the interface is deleted from the existing bundle and added to the new bundle.	■ To add an interface in the multilink bundle, select the interface in the **Logical Interfaces** list and click the left arrow button to add it in the **Member Interfaces** list. ■ To remove an interface from the multilink bundle, select the interface in the **Member Interfaces** list and click the right arrow button to remove it from the **Member Interfaces** list.

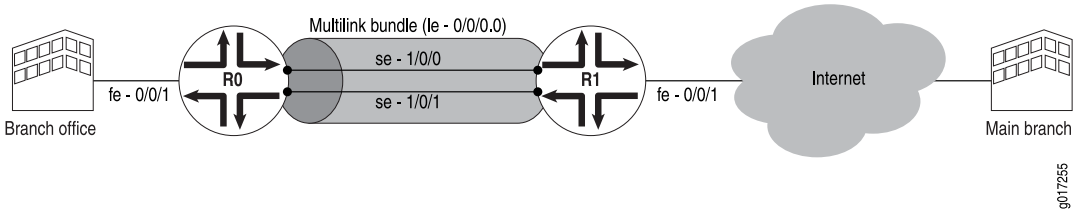
Configuring the Link Services Interface with a Configuration Editor

This section contains the following topics:

Configuring MLPPP Bundles and LFI on Serial Links

Figure 34 on page 254 shows a network topology that is used as an example in this section. In this example, your company's branch office is connected to its main branch using J-series devices R0 and R1. You transmit data and voice traffic on two low-speed 1-Mbps serial links. To increase bandwidth, you configure MLPPP and join the two serial links se-1/0/0 and se-1/0/1 into a multilink bundle ls-0/0/0.0. Then you configure LFI and CoS on R0 and R1 to enable them to transmit voice packets ahead of data packets.

Figure 34: Configuring MLPPP and LFI on Serial Links



Configuring a multilink bundle on the two serial links increases the bandwidth by 70 percent from approximately 1 Mbps to 1.7 Mbps and prepends each packet with a multilink header as specified in the FRF.12 standard. To increase the bandwidth further, you can add up to 8 serial links to the bundle. In addition to a higher bandwidth, configuring the multilink bundle provides load balancing and redundancy. If one of the serial links fails, traffic continues to be transmitted on the other links without any interruption. In contrast, independent links require routing policies for load balancing and redundancy. Independent links also require IP addresses for each link as opposed to one IP address for the bundle. In the routing table, the multilink bundle is represented as a single interface.

This example uses MLPPP for providing multilink services. For information about configuring MLFR, see “Configuring MLFR FRF.15 Bundles” on page 264 and “Configuring MLFR FRF.16 Bundles” on page 267.

You can use the LFI and CoS configurations provided in this example with MLFR FRF.15 and MLFR FRF.16 bundles, too. You can also use the same LFI and CoS configurations for other interfaces, such as on T1 or E1.

To configure MLPPP bundles and LFI, perform the following tasks:

- Configuring an MLPPP Bundle on page 254
- Enabling Link Fragmentation and Interleaving on page 256
- Defining Classifiers and Forwarding Classes on page 257
- Defining and Applying Scheduler Maps on page 259
- Applying Shaping Rates to Interfaces on page 263

Configuring an MLPPP Bundle

In this example, you create an MLPPP bundle (ls-0/0/0.0) at the logical unit level of the link services interface (ls-0/0/0) on J-series devices R0 and R1. Then you add the two serial interfaces se-1/0/0 and se-1/0/1 as constituent links to the multilink bundle. Adding multiple links does not require you to configure and manage more addresses.

To configure an MLPPP bundle on a J-series device:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 86 on page 255 on Device R0 and Device R1.

3. Go on to "Enabling Link Fragmentation and Interleaving" on page 256.

Table 86: Configuring an MLPPP Bundle

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy. Specify the link services interface to be configured.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**. 3. Next to Interface, click **Add new entry**. 4. In the Interface name box, type ls-0/0/0. 5. Click **OK**.	From the [edit] hierarchy level, enter edit interfaces ls-0/0/0
Configure a logical unit on the ls-0/0/0 interface and define the family type—for example, inet. Configure an IP address for the multilink bundle at the unit level of the link services interface.	1. Next to ls-0/0/0, click **Edit**. 2. Next to Unit, click **Add new entry**. 3. In the Interface unit number box, type 0. 4. Under Family, select **Inet** and click **Configure**. 5. Next to Address, click **Add new entry**. 6. In the Source box, type the appropriate source address: ■ On R0—10.0.0.10/24 ■ On R1—10.0.0.9/24 7. Click **OK** until you return to the Interfaces page.	Set the appropriate source address for the interface: ■ On R0, enter set unit 0 family inet address 10.0.0.10/24 ■ On R1, enter set unit 0 family inet address 10.0.0.9/24
From the **Interfaces** level in the configuration hierarchy, specify the names of the constituent links to be added to the multilink bundle—for example, se-1/0/0 and se-1/0/1.	1. On the Interfaces page, Next to Interface, click **Add new entry**. 2. In the Interface name box, type the name of the interface to be added to the multilink bundle—for example se-1/0/0 or se-1/0/1. 3. Click **OK**. 4. Click **Edit** next to the appropriate interface name—for example, se-1/0/0 or se-1/0/1.	From the [edit] hierarchy level, add the constituent links to the multilink bundle. ■ To add se-1/0/0 to the multilink bundle, enter edit interfaces se-1/0/0 ■ To add se-1/0/1 to the multilink bundle, enter edit interfaces se-1/0/1

Table 86: Configuring an MLPPP Bundle *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Create the multilink bundle by specifying a logical unit on each constituent link and defining it as an MLPPP bundle—for example, ls-0/0/0.0.	1. Next to Unit, click **Add new entry**. 2. In the Interface unit number box, type 0. 3. Under Family, select **Mlppp** and click **Configure**. 4. In the Bundle box, type ls-0/0/0.0. 5. Click **OK** until you return to the Interfaces page.	Enter set unit 0 family mlppp bundle ls-0/0/0.0
Set the serial options to the same values for both interfaces on R0—se-1/0/0 and se-1/0/1. For more information about serial options, see "Configuring Serial Interfaces with Quick Configuration" on page 95. **NOTE:** In this example, R0 is set as a data circuit-terminating equipment (DCE) device. The serial options are not set for interfaces on R1. You can set the serial options according to your network setup.	1. On the Interfaces page, click **Edit**. 2. Next to the interface that you want to configure (se-1/0/0 or se-1/0/1), click **Edit**. 3. Next to Serial options, click **Configure**. 4. From the Clocking mode list, select **dce**. 5. From the Clock rate list, select **2.0mhz**. 6. Click **OK** twice.	1. On R0, from the [edit] hierarchy level, set serial options for the interface. ■ To set options on se-1/0/0, enter edit interfaces se-1/0/0 ■ To set options on se-1/0/1, enter edit interfaces se-1/0/1 2. Enter set serial-options clocking-mode dce clock-rate 2.0mhz

Enabling Link Fragmentation and Interleaving

To configure link fragmentation and interleaving (LFI), you define the MLPPP encapsulation type and enable fragmentation and interleaving of packets by specifying the following properties—the fragmentation threshold and fragment interleaving. In this example, a fragmentation threshold of 128 bytes is set on the MLPPP bundle that applies to all traffic on both constituent links, so that any packet larger than 128 bytes transmitted on these links is fragmented.

For more information about LFI, see "Link Fragmentation and Interleaving Overview" on page 245.

To enable LFI:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 87 on page 257 on Device R0 and Device R1.
3. Go on to "Defining Classifiers and Forwarding Classes" on page 257.

Table 87: Enabling LFI

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy. Specify the link services interface for fragmentation.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Edit**. 3. Under Interface, next to ls-0/0/0, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces ls-0/0/0
Specify the multilink encapsulation type, enable LFI, and set the fragmentation threshold for the multilink interface. Fragment Threshold—Set the maximum size, in bytes, for multilink packet fragments—for example, 128. Any nonzero value must be a multiple of 64 bytes. The value can be between 128 and 16320. The default is 0 bytes (no fragmentation). Interleave Fragments—Specify interleaving packet fragments with delay-sensitive (LFI) packets.	1. Under Unit, next to 0, click **Edit** 2. From the Encapsulation list, select **multilink-ppp** as the encapsulation type. 3. In the Fragment threshold box, type 128. 4. Select **Interleave fragments**. 5. Click **OK**.	Enter set unit 0 encapsulation multilink-ppp fragment-threshold 128 interleave-fragments

Defining Classifiers and Forwarding Classes

By defining classifiers you associate incoming packets with a forwarding class and loss priority. Based on the associated forwarding class, you assign packets to output queues. To configure classifiers, you specify the bit pattern for the different types of traffic. The classifier takes this bit pattern and attempts to match it to the type of packet arriving on the interface. If the information in the packet's header matches the specified pattern, the packet is sent to the appropriate queue, defined by the forwarding class associated with the classifier.

In this example, an IP precedence classifier, classify_input, is assigned to all incoming traffic. The precedence bit value in the type of service (ToS) field is assumed to be 000 for all incoming data traffic and 010 for all incoming voice traffic. This classifier assigns all data traffic to Q0 and all voice traffic to Q2. On a J-series device, when LFI is enabled, all traffic assigned to Q2 is treated as LFI (voice) traffic. You do not need to assign network control traffic to a queue explicitly, because it is assigned to Q3 by default.

For more information about configuring CoS components, see "Configuring Class of Service" on page 579.

To define classifiers and forwarding classes:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.

2. Perform the configuration tasks described in Table 88 on page 258 on Device R0 and Device R1.
3. Go on to "Defining and Applying Scheduler Maps" on page 259.

Table 88: Defining Classifiers and Forwarding Classes

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Class of service** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit class-of-service
Configure a behavior aggregate (BA) classifier for classifying packets. In this example, you specify the default IP precedence classifier, which maps IP precedence bits to forwarding classes and loss priorities.	1. Next to Classifiers, click **Configure**. 2. Next to Inet precedence, click **Add new entry**. 3. In the Name box, type classify_input.	Enter edit classifiers inet-precedence classify_input
For the classifier to assign an output queue to each packet, it must associate the packet with a forwarding class. Assign packets with IP precedence bits 000 to the DATA forwarding class, and specify a low loss priority.	1. On the Inet precedence page, next to Forwarding class, click **Add new entry**. 2. In the Class name box, type DATA. 3. Next to Loss priority, click **Add new entry**. 4. From the Loss val list, select **low**. 5. Next to Code points, click **Add new entry**. 6. In the Value box, type 000. 7. Click **OK** until you return to the Inet precedence page.	Enter set forwarding-class DATA loss-priority low code-points 000
Assign packets with IP precedence bits 010 to the VOICE forwarding class, and specify a low loss priority.	1. Next to Forwarding class, click **Add new entry**. 2. In the Class name box, type VOICE. 3. Next to Loss priority, click **Add new entry**. 4. From the Loss val list, select **low**. 5. Next to Code points, click **Add new entry**. 6. In the Value box, type 010. 7. Click **OK** until you return to the Class of service page.	Enter set forwarding-class VOICE loss-priority low code-points 010

Table 88: Defining Classifiers and Forwarding Classes *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Assign each forwarding class one-to-one with the output queues. ■ DATA—Assign to Queue 0. ■ VOICE—Assign to Queue 2. ■ NC (Network Control)—Assign to Queue 3. NC is assigned to Queue 3 by default.	1. On the Class of service page, next to Forwarding classes, click **Configure**. 2. Next to Queue, click **Add new entry**. 3. In the Queue num box, type 0. 4. In the Class name box, type DATA. 5. Click **OK**. 6. Next to Queue, click **Add new entry**. 7. In the Queue num box, type 2. 8. In the Class name box, type VOICE. 9. Click **OK**. 10. Next to Queue, click **Add new entry**. 11. In the Queue num box, type 3. 12. In the Class name box, type NC. 13. Click **OK** until you return to the Class of service page.	From the [edit class-of-service] hierarchy level, enter set forwarding-classes queue 0 DATA set forwarding-classes queue 2 VOICE set forwarding-classes queue 3 NC
Apply the behavior aggregate classifier to the incoming interface.	1. On the Class of service page, next to Interfaces, click **Configure** or **Edit**. 2. Next to Interface, click **Add new entry**. 3. In the Interface name box, type ge-0/0/1. 4. Next to Unit, click **Add new entry**. 5. In the Unit number box, type 0. 6. Next to Classifiers, click **Configure**. 7. Under Inet precedence, in the Classifier name box, type classify_input. 8. Click **OK**.	1. From the [edit class-of-service] hierarchy level, enter edit interfaces ge-0/0/1 2. Enter set unit 0 classifiers inet-precedence classify_input

Defining and Applying Scheduler Maps

By defining schedulers you configure the properties of output queues that determine the transmission service level for each queue. These properties include the amount of interface bandwidth assigned to the queue, the size of the memory buffer allocated for storing packets, and the priority of the queue. After defining schedulers you

associate them with forwarding classes by means of scheduler maps. You then associate each scheduler map with an interface, thereby configuring the hardware queues and packet schedulers that operate according to this mapping.

In this example, you define and apply scheduler maps as follows:

- Enable per-unit scheduling that allows configuration of scheduler maps on the bundle.
- Create three schedulers—DATA, VOICE, and NC. Define the VOICE and NC schedulers to have a high priority and the DATA scheduler to have the default priority (low). These priority assignments allow all voice and network control traffic to be transmitted ahead of data packets. For more information about scheduling priorities, see "Queuing with LFI on J-series Devices" on page 246.
- Create a scheduler map s_map that associates these schedulers with corresponding forwarding classes.
- Apply the scheduler map to the multilink bundle and the serial interfaces.

To define and apply scheduler maps:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 89 on page 260 on Device R0 and Device R1.
3. Go on to "Applying Shaping Rates to Interfaces" on page 263.

Table 89: Defining and Applying Scheduler Maps

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interface** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces
To configure CoS components for each multilink bundle, enable per-unit scheduling on the interface.	1. Under Interfaces, select **ls-0/0/0**. 2. From the Scheduler type list, select **Per unit scheduler**. 3. Click **OK**. 4. Under Interfaces, select **se-1/0/0**. 5. From the Scheduler type list, select **Per unit scheduler**. 6. Click **OK**. 7. Under Interfaces, select **se-1/0/1**. 8. From the Scheduler type list, select **Per unit scheduler**. 9. Click **OK** twice.	Enter set ls-0/0/0 per-unit-scheduler set se-1/0/0 per-unit-scheduler set se-1/0/1 per-unit-scheduler

Table 89: Defining and Applying Scheduler Maps *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the Class of Service configuration hierarchy and specify the link services interface to be configured.	1. On the Class of service page, next to Interfaces, click **Configure** or **Edit**. 2. Next to Interface, click **Add new entry**. 3. In the Interface name box, type ls-0/0/0.	From the [edit class-of-service] hierarchy level, enter edit interfaces ls-0/0/0
Define a scheduler map—for example, s_map.	1. Next to Unit, type **Add new entry**. 2. In the Unit number box, type 0. 3. In the Scheduler map box, type s_map. 4. Click **OK** twice.	Enter set unit 0 scheduler-map s_map
Apply the scheduler map to the constituent links of the multilink bundle—for example, se-1/0/0 and se-1/0/1.	1. On the Class of service page, next to Interfaces, click **Configure** or **Edit**. 2. Next to Interface, click **Add new entry**. 3. In the Interface name box, type the name of the interface on which scheduler map s_map is to be applied—for example, se-1/0/0 or se-1/0/1. 4. Next to Unit, type **Add new entry**. 5. In the Unit number box, type 0. 6. In the Scheduler map box, type s_map. 7. Click **OK** twice.	1. From the [edit] hierarchy level, specify the interface to be configured. ■ To apply the scheduler map to se-1/0/0, enter edit interfaces se-1/0/0 ■ To apply the scheduler map to se-1/0/1, enter edit interfaces se-1/0/1 2. Apply the scheduler map to the logical interface. set unit 0 scheduler-map s_map

Table 89: Defining and Applying Scheduler Maps *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Associate a scheduler with each forwarding class. ■ DATA—A scheduler associated with the DATA forwarding class. ■ VOICE—A scheduler associated with the VOICE forwarding class. ■ NC—A scheduler associated with the NC forwarding class. A scheduler receives the forwarding class and loss priority settings, and queues the outgoing packet based on those settings.	1. On the Class of service page, next to Scheduler maps, click **Add new entry**. 2. In the Map name box, type s_map. 3. Next to Forwarding class, click **Add new entry**. 4. In the Class name box, type DATA. 5. In the Scheduler box, type DATA. 6. Click **OK**. 7. Next to Forwarding class, click **Add new entry**. 8. In the Class name box, type VOICE. 9. In the Scheduler box, type VOICE. 10. Click **OK**. 11. Next to Forwarding class, click **Add new entry**. 12. In the Class name box, type NC. 13. In the Scheduler box, type NC. 14. Click **OK** until you return to the Class of service page.	From the [edit class-of-service] hierarchy level, enter set scheduler-maps s_map forwarding-class DATA scheduler DATA set scheduler-maps s_map forwarding-class VOICE scheduler VOICE set scheduler-maps s_map forwarding-class NC scheduler NC
Define the properties of output queues for the DATA scheduler: ■ Transmit rate—Specify a percentage of transmission capacity—49. ■ Buffer size—Specify a percentage of total buffer—49. ■ Priority—Do not specify the transmission priority for the DATA scheduler to apply the default setting—low. For more information about transmit rate and buffer size, see “Configuring Schedulers” on page 623.	1. On the Class of service page, next to Schedulers, click **Add new entry**. 2. In the Scheduler name box, type DATA. 3. Next to Transmit rate, click **Configure**. 4. From the Transmit rate choice list, select **Percent**. 5. In the Percent box, type 49. 6. Click **OK**. 7. Next to Buffer size, click **Configure**. 8. From the Buffer size choice list, select **Percent**. 9. In the Percent box, type 49. 10. Click **OK** twice.	Enter set schedulers DATA transmit-rate percent 49 set schedulers DATA buffer-size percent 49

Table 89: Defining and Applying Scheduler Maps *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define the properties of output queues for the VOICE scheduler: ■ Transmit rate—Specify a percentage of transmission capacity—50. ■ Buffer size—Specify a percentage of total buffer—5. ■ Priority—Specify a transmission priority—high.	1. On the Class of service page, next to Schedulers, click **Add new entry**. 2. In the Scheduler name box, type VOICE. 3. Next to Transmit rate, click **Configure**. 4. From the Transmit rate choice list, select **Percent**. 5. In the Percent box, type 50. 6. Click **OK**. 7. Next to Buffer size, click **Configure**. 8. From the Buffer size choice list, select **Percent**. 9. In the Percent box, type 5. 10. Click **OK**. 11. In the Priority box, type high. 12. Click **OK**.	Enter set schedulers VOICE transmit-rate percent 50 set schedulers VOICE buffer-size percent 5 set schedulers VOICE priority high
Define the properties of output queues for the NC scheduler: ■ Transmit rate—Specify a percentage of transmission capacity—1. ■ Buffer size—Specify a percentage of total buffer—1. ■ Priority—Specify a transmission priority—high.	1. On the Class of service page, next to Schedulers, click **Add new entry**. 2. In the Scheduler name box, type NC. 3. Next to Transmit rate, click **Configure**. 4. From the Transmit rate choice list, select **Percent**. 5. In the Percent box, type 1. 6. Click **OK**. 7. Next to Buffer size, click **Configure**. 8. From the Buffer size choice list, select **Percent**. 9. In the Percent box, type 1. 10. Click **OK**. 11. In the Priority box, type high. 12. Click **OK**.	Enter set schedulers NC transmit-rate percent 1 set schedulers NC buffer-size percent 1 set schedulers NC priority high

Applying Shaping Rates to Interfaces

To control the voice traffic latency within acceptable limits, you configure the shaping rate on constituent links of the MLPPP bundle. Shaping rate at the interface level is

required only when you enable LFI. To apply shaping rates to interfaces, you have to first enable per-unit scheduling. For information about shaping rates and LFI, see "Configuring CoS Components with LFI" on page 249.

You must configure the shaping rate to be equal to the combined physical interface bandwidth for the constituent links. In this example, the combined bandwidth capacity of the two constituent links—se-1/0/0 and se-1/0/1—is 2 Mbps. Hence, configure a shaping rate of 2 Mbps on each constituent link.

To apply a shaping rate to the constituent links of the multilink bundle:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 90 on page 264 on Device R0 and Device R1.
3. Go on to "Verifying the Link Services Interface Configuration" on page 271, to verify your configuration.

Table 90: Applying Shaping Rate to Interfaces

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Class of service** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Edit**.	From the [edit] hierarchy level, enter edit class-of-service
Apply the shaping rate to the constituent links of the multilink bundle—for example, se-1/0/0 and se-1/0/1. The shaping rate specifies the amount of bandwidth to be allocated for this multilink bundle.	1. Under Interfaces, select the name of the interface on which you want to apply the shaping rate—**se-1/0/0** or **se-1/0/1**. 2. Next to Unit 0, click **Edit**. 3. Select **Shaping rate**, and click **Configure**. 4. From the Shaping rate choice list, select **Rate**. 5. In the Rate box, type 2000000. 6. Click **OK**.	1. Set the shaping rate on both the constituent links: ■ To set the shaping rate for se-1/0/0, enter edit interfaces se-1/0/0 ■ To set the shaping rate for se-1/0/1, enter edit interfaces se-1/0/1 2. Set the shaping rate: set unit 0 shaping-rate 2000000

Configuring MLFR FRF.15 Bundles

J-series devices support Multilink Frame Relay end-to-end (MLFR FRF.15) on the link services interface ls-0/0/0.

With MLFR FRF.15, multilink bundles are configured as logical units on the link services interface, such as ls-0/0/0.0. MLFR FRF.15 bundles combine multiple permanent virtual circuits (PVCs) into one aggregated virtual circuit (AVC). This process provides fragmentation over multiple PVCs on one end and reassembly of

the AVC on the other end. For more information about multilink bundles, see “Multilink Bundles Overview” on page 244.

You can configure LFI and CoS with MLFR in the same way that you configure them with MLPPP. For information about configuring LFI and CoS, see “Configuring MLPPP Bundles and LFI on Serial Links” on page 253.

In this example, you aggregate two T1 links to create an MLFR FRF.15 bundle on two J-series devices—Device R0 and Device R1.

To configure an MLFR FRF.15 bundle:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor on Device R0 and Device R1.
2. Perform the configuration tasks described in Table 91 on page 265.
3. If you are finished configuring the device, commit the configuration.
4. Go on to “Verifying the Link Services Interface Configuration” on page 271, to verify your configuration.

Table 91: Configuring MLFR FRF.15 Bundles

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy. Specify the link services interface as an interface to be configured.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**. 3. Next to Interface, click **Add new entry**. 4. In the Interface name box, type ls-0/0/0. 5. Click **OK**.	From the [edit] hierarchy level, enter edit interfaces ls-0/0/0

Table 91: Configuring MLFR FRF.15 Bundles *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure a logical unit on the ls-0/0/0 interface, and define the family type—for example, inet. Configure an IP address for the multilink bundle on the unit level of the link services interface.	1. On the Interfaces page, next to ls-0/0/0, click **Edit**. 2. Next to Unit, click **Add new entry**. 3. In the Interface unit number box, type 0. 4. Under Family, select **Inet** and click **Configure**. 5. Next to Address, click **Add new entry**. 6. In the Source box, type the appropriate source address: ■ On R0—10.0.0.4/24 ■ On R1—10.0.0.5/24 7. Click **OK** until you return to the Interfaces page.	Set the appropriate source address for the interface: ■ On R0, enter set unit 0 family inet address 10.0.0.4/24 ■ On R1, enter set unit 0 family inet address 10.0.0.5/24
Define the multilink bundle as an MLFR FRF.15 bundle by specifying the Multilink Frame Relay end-to-end encapsulation type.	1. On the Interfaces page, next to ls-0/0/0, click **Edit**. 2. Under Unit, next to 0, click **Edit**. 3. From the Encapsulation list, select **multilink-frame-relay-end-to-end**. 4. Click **OK** until you return to the Interfaces page.	From the [edit interfaces ls-0/0/0] hierarchy level, enter set unit 0 encapsulation multilink-frame-relay-end-to-end
Specify the names of the constituent links to be added to the multilink bundle—for example, t1-2/0/0 and t1-2/0/1. Define the Frame Relay encapsulation type.	1. On the Interfaces page, next to Interface, click **Add new entry**. 2. In the Interface name box, type the name of the interface: ■ To configure t1–2/0/0, type t1-2/0/0. ■ To configure t1–2/0/1, type t1-2/0/1. 3. Click **OK**. 4. Next to the interface you want to configure, click **Edit**. 5. From the Encapsulation list, select **frame-relay**.	1. From the [edit] hierarchy level, enter ■ For configuring t1-2/0/0 edit interfaces t1-2/0/0 ■ For configuring t1-2/0/1 edit interfaces t1-2/0/1 2. Enter set encapsulation frame-relay

Table 91: Configuring MLFR FRF.15 Bundles *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define R0 to be a data circuit-terminating equipment (DCE) device. R1 performs as a data terminal equipment (DTE) device, which is the default with Frame Relay encapsulation. For more information about DCE and DTE, see "Serial Interface Overview" on page 37.	On R0 only, select **Dce**.	On R0 only, enter set dce
On the logical unit level of the interface, specify the data-link connection identifier (DLCI). The DLCI field identifies which logical circuit the data travels over. DLCI is a value from 16 through 1022—for example, 100. (Numbers 1 through 15 are reserved for future use.) Specify the multilink bundle to which the interface is to be added as a constituent link—ls-0/0/0.0.	1. Next to Unit, click **Add new entry**. 2. In the Interface unit number box, type 0. 3. In the Dlci box, type 100. 4. Under Family, select **mlfr-end-to-end** and click **Configure**. 5. In the Bundle box, type ls-0/0/0.0. 6. Click **OK**.	Enter set unit 0 dlci 100 family mlfr-end-to-end bundle ls-0/0/0.0

Configuring MLFR FRF.16 Bundles

J-series devices support Multilink Frame Relay (MLFR) user-to-network interface (UNI) network-to-network interface (NNI) (MLFR FRF.16) on the link services interface ls-0/0/0.

MLFR FRF.16 configures multilink bundles as channels on the link services interface, such as ls-0/0/0:0. A multilink bundle carries Frame Relay permanent virtual circuits (PVCs), identified by their data-link connection identifiers (DLCIs). Each DLCI is configured at the logical unit level of the link services interface and is also referred as a logical interface. Packet fragmentation and reassembly occur on each virtual circuit. For more information about multilink bundles, see "Multilink Bundles Overview" on page 244.

You can configure LFI and CoS with MLFR in the same way that you configure them with MLPPP. For information about configuring LFI and CoS, see "Configuring MLPPP Bundles and LFI on Serial Links" on page 253.

In this example, you aggregate two T1 interfaces to create an MLFR FRF.16 bundle on two J-series devices—Device R0 and Device R1.

To configure an MLFR FRF.16 bundle:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor on Device R0 and Device R1.
2. Perform the configuration tasks described in Table 92 on page 268.

3. If you are finished configuring the device, commit the configuration.
4. Go on to "Verifying the Link Services Interface Configuration" on page 271, to verify your configuration.

Table 92: Configuring MLFR FRF.16 Bundles

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the Chassis level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Chassis, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit chassis
Specify the number of multilink frame relay UNI NNI (FRF.16) bundles to be created on the interface. You can specify a number from 1 through 255.	1. Next to Fpc, click **Add new entry**. 2. In the Slot box, type 0. 3. Next to Pic, click **Add new entry**. 4. In the Slot box, type 0. 5. In the Mlfr uni nni bundles box, type 1. 6. Click **OK**.	Enter set fpc 0 pic 0 mlfr-uni-nni-bundles 1
Specify the channel to be configured as a multilink bundle.	1. On the main Configuration page, next to Interfaces, click **Configure** or **Edit**. 2. Next to Interface, click **Add new entry**. 3. In the Interface name box, type ls-0/0/0:0. 4. Click **OK**.	From the [edit] hierarchy level, enter edit interfaces ls-0/0/0:0
Define the multilink bundle as an MLFR FRF.16 bundle by specifying the Multilink Frame Relay UNI NNI encapsulation type.	1. Next to ls-0/0/0:0, click **Edit**. 2. From the Encapsulation list, select **multilink-frame-relay-uni-nni**.	Enter set encapsulation multilink-frame-relay-uni-nni
Define R0 to be a data circuit-terminating equipment (DCE) device. R1 performs as a data terminal equipment (DTE) device, which is the default with Frame Relay encapsulation. For more information about DCE and DTE, see "Serial Interface Overview" on page 37	On R0 only, select **Dce**.	On R0 only, enter set dce

Table 92: Configuring MLFR FRF.16 Bundles *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure a logical unit on the multilink bundle ls-0/0/0:0, and define the family type—for example, inet. Assign a data link connection identifier (DLCI) to the multilink bundle. The DLCI field identifies which logical circuit the data travels over. DLCI is a value from 16 through 1022—for example, 400. (Numbers 1 through 15 are reserved for future use.) Assign an IP address to the multilink bundle.	1. Next to Unit, click **Add new entry**. 2. In the Interface unit number box, type 0. 3. In the Dlci box, type 400. 4. Under Family, select **Inet** and click **Configure**. 5. Next to Address, click **Add new entry**. 6. In the Source box, type the appropriate source address: ■ On R0—10.0.0.10/24 ■ On R1—10.0.0.9/24 7. Click **OK** until you return to the Interfaces page.	Set the appropriate address for the interface: ■ On R0, enter set unit 0 dlci 400 family inet address 10.0.0.10/24 ■ On R1, enter set unit 0 dlci 400 family inet address 10.0.0.9/24
Create the T1 interfaces that are to be added as constituent links to the multilink bundle—t1-2/0/0 and t1-2/0/1. Define the Frame Relay encapsulation type.	1. On the Interfaces page, next to Interface, click **Add new entry**. 2. In the Interface name box, type the name of the interface: ■ To configure t1-2/0/0, type t1-2/0/0. ■ To configure t1-2/0/1, type t1-2/0/1. 3. Click **OK**. 4. Next to the interface you want to configure, click **Edit**. 5. From the Encapsulation list, select **multilink-frame-relay-uni-nni**.	1. From the [edit] hierarchy level, enter ■ For configuring t1-2/0/0 edit interfaces t1-2/0/0 ■ For configuring t1-2/0/1 edit interfaces t1-2/0/1 2. Enter set encapsulation multilink-frame-relay-uni-nni
Specify the multilink bundle to which the interface is to be added as a constituent link—ls-0/0/0:0.	1. Next to Unit, click **Add new entry**. 2. In the Interface unit number box, type 0. 3. Under Family, select **mlfr-uni-nni** and click **Configure**. 4. In the Bundle box, type ls-0/0/0:0. 5. Click **OK**.	Enter set unit 0 family mlfr-uni-nni bundle ls-0/0/0:0

Configuring CRTP

Compressed Real-Time Transport Protocol (CRTP) is typically used for compressing voice and video packets. You can configure CRTP with LFI on the link services interface of a J-series device.

On the J-series device, CRTP can be configured as a compression device on a T1 or E1 interface with PPP encapsulation, using the link services interface.

For more information about configuring CRTP on a single link, see the *JUNOS Network Interfaces Configuration Guide* and the *JUNOS Services Interfaces Configuration Guide*.

To configure CRTP on the device:

1. Navigate to the top of the interfaces configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 93 on page 270.
3. If you are finished configuring the device, commit the configuration.

Table 93: Adding CRTP to an T1 or E1 Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces *interface-name*
Select an E1 or T1 interface—for example, t1–1/0/0. Set PPP as the type of encapsulation for the physical interface.	1. Next to a T1 or E1 interface, click **Edit**. 2. From the Encapsulation list, select **ppp** as the encapsulation type. 3. Next to Unit, click **Add new entry**. 4. In the Interface unit number box, type 0.	1. Enter set encapsulation ppp 2. Enter edit unit 0
Add the link services interface, ls-0/0/0.0, to the physical interface.	1. In the Compression device box, enter ls-0/0/0.0. 2. Click **OK** until you return to the Interfaces page.	Enter set compression-device ls-0/0/0.0
Add the link services interface, ls-0/0/0, to the device.	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type ls-0/0/0. 3. Click **OK** to return to the Interfaces page. 4. On the main Interface page, next to ls-0/0/0, click **Edit**. 5. Next to Unit, click **Add new entry**. 6. In the Interface unit number box, type 0.	From the [edit interfaces] hierarchy level, enter edit interfaces ls-0/0/0 unit 0

Table 93: Adding CRTP to an T1 or E1 Interface *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure the link services interface, ls-0/0/0, properties. **F-max period**—Maximum number of compressed packets allowed between transmission of full headers. It has a range from 1 to 65535. **Maximum** and **Minimum**—UDP port values from 1 to 65536 reserve these ports for RTP compression. CRTP is applied to network traffic on ports within this range. This feature is applicable only to voice services interfaces.	1. Next to Compression, select **yes**, and then click **Configure**. 2. Select **RTP**, and then click **Configure**. 3. In the F-Max period box, type 2500. 4. Select Port, then click **Configure**. 5. In the Minimum value box, type 2000. 6. In the Maximum value box, type 64009. 7. Click **OK**.	Enter set compression rtp f-max-period 2500 port minimum 2000 maximum 64009

Verifying the Link Services Interface Configuration

To verify a link services configuration, perform the following tasks:

- Displaying Multilink Bundle Configurations on page 271
- Displaying Link Services CoS Configurations on page 272
- Verifying Link Services Interface Statistics on page 274
- Verifying Link Services CoS on page 276

Displaying Multilink Bundle Configurations

Purpose Verify the multilink bundle configuration.

Action From the J-Web interface, select **Configuration > View and Edit > View Configuration Text**. Alternatively, from configuration mode in the CLI, enter the show interfaces command.

The sample output in this section displays the multilink bundle configurations provided in "Configuring MLPPP Bundles and LFI on Serial Links" on page 253.

NOTE: The MLFR FRF.15 and MLFR FRF.16 configurations are not displayed in this section, but you can display MLFR configurations in the same manner.

```
[edit]
user@R0# show interfaces
interfaces {
    ls-0/0/0 {
        per-unit-scheduler;
        unit 0 {
            encapsulation multilink-ppp;
            fragment-threshold 128;
            interleave-fragments;
```

```
            family inet {
               address 10.0.0.10/24;
            }
         }
      }
      ge-0/0/1 {
         unit 0 {
            family inet {
               address 192.1.1.1/24;
            }
         }
      }
      se-1/0/0 {
         per-unit-scheduler;
         dce-options {
            clocking-mode dce;
            clocking-rate 2.0mhz;
         }
         unit 0 {
            family mlppp {
               bundle ls-0/0/0.0;
            }
         }
      }
      se-1/0/1 {
         per-unit-scheduler;
         dce-options {
            clocking-mode dce;
            clocking-rate 2.0mhz;
         }
         unit 0 {
            family mlppp {
               bundle ls-0/0/0.0;
            }
         }
      }
   }
```

Meaning Verify that the output shows the intended multilink bundle configurations.

Related Topics For more information about the format of a configuration file, see the *J-Web Interface User Guide* or the *JUNOS CLI User Guide*.

Displaying Link Services CoS Configurations

Purpose Displaying the CoS configurations on the link services interface.

Action From the J-Web interface, select **Configuration > View and Edit > View Configuration Text**. Alternatively, from the configuration mode in the CLI, enter the show class-of-service command.

The sample output in this section displays the CoS configurations provided in "Configuring MLPPP Bundles and LFI on Serial Links" on page 253.

```
[edit]
user@R0# show class-of-service
classifiers {
   inet-precedence classify_input {
      forwarding-class DATA {
         loss-priority low code-points 000;
      }
      forwarding-class VOICE {
         loss-priority low code-points 010;
      }
   }
}
forwarding-classes {
   queue 0 DATA;
   queue 2 VOICE;
   queue 3 NC;
}
interfaces {
   ls-0/0/0 {
      unit 0 {
         scheduler-map s_map;
      }
   }
   ge-0/0/1 {
      unit 0 {
         classifiers {
            inet-precedence classify_input
         }
      }
   }
   se-1/0/0 {
      unit 0 {
         scheduler-map s_map;
         shaping-rate 2000000;
      }
   }
   se-1/0/1 {
      unit 0 {
         scheduler-map s_map;
         shaping-rate 2000000;
      }
   }
}
scheduler-maps {
   s_map {
      forwarding-class DATA scheduler DATA;
      forwarding-class VOICE scheduler VOICE;
      forwarding-class NC scheduler NC;
   }
}
schedulers {
   DATA {
      transmit-rate percent 49;
      buffer-size percent 49;
   }
   VOICE {
```

```
        transmit-rate percent 50;
        buffer-size percent 5;
        priority high;
    }
    NC {
        transmit-rate percent 1;
        buffer-size percent 1;
        priority high;
    }
}
```

Meaning Verify that the output shows the intended CoS configurations.

Related Topics For more information about the format of a configuration file, see the *J-Web Interface User Guide* or the *JUNOS CLI User Guide*.

Verifying Link Services Interface Statistics

Purpose Verify the link services interface statistics.

Action The sample output provided in this section is based on the configurations provided in "Configuring MLPPP Bundles and LFI on Serial Links" on page 253. To verify that the constituent links are added to the bundle correctly and the packets are fragmented and transmitted correctly, take the following actions:

1. On Device R0 and Device R1, the two J-series devices used in this example, configure MLPPP and LFI as described in "Configuring MLPPP Bundles and LFI on Serial Links" on page 253.
2. From the CLI, enter the ping command to verify that a connection is established between R0 and R1.
3. Transmit 10 data packets, 200 bytes each, from R0 to R1.
4. On R0, from the CLI, enter the show interfaces *interface-name* statistics command.

Sample Output

```
user@R0> show interfaces ls-0/0/0 statistics detail
Physical interface: ls-0/0/0, Enabled, Physical link is Up
  Interface index: 134, SNMP ifIndex: 29, Generation: 135
  Link-level type: LinkService, MTU: 1504
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Last flapped   : 2006-06-23 11:36:23 PDT (03:38:43 ago)
  Statistics last cleared: 2006-06-23 15:13:12 PDT (00:01:54 ago)
  Traffic statistics:
   Input  bytes  :                    0                    0 bps
   Output bytes  :                 1820                    0 bps
   Input  packets:                    0                    0 pps
   Output packets:                   10                    0 pps
  ...
  Egress queues: 8 supported, 8 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 DATA                          10                   10                    0

    1 expedited-fo                   0                    0                    0

    2 VOICE                          0                    0                    0
```

```
    3 NC                              0                    0                    0


  Logical interface ls-0/0/0.0 (Index 67) (SNMP ifIndex 41) (Generation 133)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: Multilink-PPP
    Bandwidth: 16mbps
    Bundle options:
      ....
      Drop timer period               0
      Sequence number format          long (24 bits)
      Fragmentation threshold         128
      Links needed to sustain bundle  1
      Interleave fragments            Enabled
    Bundle errors:
      Packet drops                    0 (0 bytes)
      Fragment drops                  0 (0 bytes)
     ...
    Statistics         Frames        fps         Bytes          bps
    Bundle:
      Fragments:
        Input :             0          0             0            0
        Output:            20          0          1920            0
      Packets:
        Input :             0          0             0            0
        Output:            10          0          1820            0
    Link:
      se-1/0/0.0
        Input :             0          0             0            0
        Output:            10          0          1320            0
      se-1/0/1.0
        Input :             0          0             0            0
        Output:            10          0           600            0
  ...
      Destination: 10.0.0.9/24, Local: 10.0.0.10, Broadcast: Unspecified,
Generation:144
```

Meaning This output shows a summary of interface information. Verify the following information:

- Physical interface—The physical interface is Enabled. If the interface is shown as Disabled, do either of the following:
  - In the CLI configuration editor, delete the disable statement at the [edit interfaces *interface-name*] level of the configuration hierarchy.
  - In the J-Web configuration editor, clear the **Disable** check box on the **Interfaces > *interface-name*** page.
- Physical link—The physical link is Up. A link state of Down indicates a problem with the interface module, interface port, or physical connection (link-layer errors).
- Last flapped—The Last Flapped time is an expected value. The Last Flapped time indicates the last time the physical interface became unavailable and then available again. Unexpected flapping indicates likely link-layer errors.
- Traffic statistics—Number and rate of bytes and packets received and transmitted on the interface. Verify that the number of inbound and outbound bytes and

packets match the expected throughput for the physical interface. To clear the statistics and see only new changes, use the clear interfaces statistics *interface-name* command.

- Queue counters—Name and number of queues are as configured. This sample output shows that 10 data packets were transmitted and no packets were dropped.
- Logical interface—Name of the multilink bundle you configured—ls-0/0/0.0.
- Bundle options—Fragmentation threshold is correctly configured, and fragment interleaving is enabled.
- Bundle errors—Any packets and fragments dropped by the bundle.
- Statistics—The fragments and packets are received and transmitted correctly by the device. All references to traffic direction (input or output) are defined with respect to the device. Input fragments received by the device are assembled into input packets. Output packets are segmented into output fragments for transmission out of the device.

  In this example, 10 data packets of 200 bytes were transmitted. Because the fragmentation threshold is set to 128 bytes, all data packets were fragmented into two fragments. The sample output shows that 10 packets and 20 fragments were transmitted correctly.
- Link—The constituent links are added to this bundle and are receiving and transmitting fragments and packets correctly. The combined number of fragments transmitted on the constituent links must be equal to the number of fragments transmitted from the bundle. This sample output shows that the bundle transmitted 20 fragments and the two constituent links se-1/0/0.0 and se-1/0/1.0.0 correctly transmitted 10+10=20 fragments.
- Destination and Local—IP address of the remote side of the multilink bundle and the local side of the multilink bundle. This sample output shows that the destination address is the address on R1 and the local address is the address on R0.

Related Topics For a complete description of show interfaces output, see the *JUNOS Interfaces Command Reference*.

Verifying Link Services CoS

Purpose Verify CoS configurations on the link services interface.

Action From the CLI, enter the following commands:

- show class-of-service interface *interface-name*
- show class-of-service classifier name *classifier-name*
- show class-of-service scheduler-map *scheduler-map-name*

The sample output provided in this section is based on the configurations provided in “Configuring MLPPP Bundles and LFI on Serial Links” on page 253.

Sample Output

```
user@R0> show class-of-service interface ls-0/0/0
```

```
Physical interface: ls-0/0/0, Index: 136
Queues supported: 8, Queues in use: 4
  Scheduler map: [default], Index: 2
  Input scheduler map: [default], Index: 3
  Chassis scheduler map: [default-chassis], Index: 4
 Logical interface: ls-0/0/0.0, Index: 69
    Object              Name                    Type              Index
    Scheduler-map       s_map                   Output            16206
    Classifier          ipprec-compatibility    ip                   12

user@R0> show class-of-service interface ge-0/0/1
Physical interface: ge-0/0/1, Index: 140
          Queues supported: 8, Queues in use: 4
          Scheduler map: [default], Index: 2
          Input scheduler map: [default], Index: 3

         Logical interface: ge-0/0/1.0, Index: 68
         Object              Name                  Type         Index
         Classifier         classfy_input           ip           4330

user@R0> show class-of-service classifier name classify_input
Classifier: classfy_input, Code point type: inet-precedence, Index: 4330

  Code point       Forwarding class            Loss priority
    000                 DATA                       low
    010                 VOICE                      low

user@R0> show class-of-service scheduler-map s_map
Scheduler map: s_map, Index: 16206

  Scheduler: DATA, Forwarding class: DATA, Index: 3810
  Transmit rate: 49 percent, Rate Limit: none, Buffer size: 49 percent,
Priority:low
    Drop profiles:
      Loss priority        Protocol         Index         Name
      Low                    any             1         [default-drop-profile]

      Medium low             any             1         [default-drop-profile]

      Medium high            any             1         [default-drop-profile]

      High                   any             1         [default-drop-profile]


  Scheduler: VOICE, Forwarding class: VOICE, Index: 43363
  Transmit rate: 50 percent, Rate Limit: none, Buffer size: 5 percent,
Priority:high
    Drop profiles:
      Loss priority        Protocol          Index        Name
      Low                     any             1        [default-drop-profile]

      Medium low              any             1        [default-drop-profile]

      Medium high             any             1        [default-drop-profile]

      High                    any             1        [default-drop-profile]


  Scheduler: NC, Forwarding class: NC, Index: 2435
  Transmit rate: 1 percent, Rate Limit: none, Buffer size: 1 percent, Priority:high
```

```
Drop profiles:
  Loss priority        Protocol         Index      Name
  Low                     any             1       [default-drop-profile]

  Medium low              any             1       [default-drop-profile]

  Medium high             any             1       [default-drop-profile]

  High                    any             1       [default-drop-profile]
```

Meaning These output examples show a summary of configured CoS components. Verify the following information:

- Logical Interface—Name of the multilink bundle and the CoS components applied to the bundle. The sample output shows that the multilink bundle is ls-0/0/0.0, and the CoS scheduler-map s_map is applied to it.
- Classifier—Code points, forwarding classes, and loss priorities assigned to the classifier. The sample output shows that a default classifier, ipprec-compatibility, was applied to the ls-0/0/0 interface and the classifier classify_input was applied to the ge-0/0/1 interface.
- Scheduler—Transmit rate, buffer size, priority, and loss priority assigned to each scheduler. The sample output displays the data, voice, and network control schedulers with all the configured values.

Related Topics For complete descriptions of show class-of-service commands and output, see the *JUNOS System Basics and Services Command Reference*.

Frequently Asked Questions About the Link Services Interface

Use answers to the following questions to solve configuration problems on a link services interface:

Which CoS Components Are Applied to the Constituent Links?

Problem—I have configured a multilink bundle, but I also have traffic without MLPPP encapsulation passing through constituent links of the multilink bundle. Do I apply all CoS components to the constituent links, or is applying them to the multilink bundle enough?

Solution—On a J-series device you can apply a scheduler map to the multilink bundle and its constituent links. Although you can apply several CoS components with the scheduler map, configure only the ones that are required. We recommend that you keep the configuration on the constituent links simple to avoid unnecessary delay in transmission.

Table 94 on page 279 shows the CoS components to be applied on a multilink bundle and its constituent links. For more information, see the *JUNOS Class of Service Configuration Guide*.

Table 94: CoS Components Applied on Multilink Bundles and Constituent Links

Cos Component	Multilink Bundle	Constituent Links	Explanation
Classifier	Yes	No	CoS classification takes place on the incoming side of the interface, not on the transmitting side, so no classifiers are needed on constituent links.
Forwarding class	Yes	No	Forwarding class is associated with a queue, and the queue is applied to the interface by a scheduler map. The queue assignment is predetermined on the constituent links. All packets from Q2 of the multilink bundle are assigned to Q2 of the constituent link, and packets from all the other queues are queued to Q0 of the constituent link.
Scheduler map	Yes	Yes	Apply scheduler maps on the multilink bundle and the constituent links, as follows: ■ Transmit rate—Make sure that the relative order of the transmit rate configured on Q0 and Q2 is the same on the constituent links as on the multilink bundle. ■ Scheduler priority—Make sure that the relative order of the scheduler priority configured on Q0 and Q2 is the same on the constituent links as on the multilink bundle. ■ Buffer size—Because all non-LFI packets from the multilink bundle transit Q0 of constituent links, make sure that the buffer size on Q0 of the constituent links is large enough. ■ RED drop profile—Configure a RED drop profile on the multilink bundle only. Configuring the RED drop profile on the constituent links applies a back pressure mechanism that changes the buffer size and introduces variation. Because this behavior might cause fragment drops on the constituent links, make sure to leave the RED drop profile at the default settings on the constituent links.
Shaping rate for a per-unit scheduler or an interface-level scheduler	No	Yes	Because per-unit scheduling is applied only at the end point, apply this shaping rate to the constituent links only. Any configuration applied earlier is overwritten by the constituent link configuration.
Transmit-rate exact or queue-level shaping	Yes	No	The interface-level shaping applied on the constituent links overrides any shaping on the queue. Thus apply transmit-rate exact shaping on the multilink bundle only.
Rewrite rules	Yes	No	Rewrite bits are copied from the packet into the fragments automatically during fragmentation. Thus what you configure on the multilink bundle is carried on the fragments to the constituent links.

Table 94: CoS Components Applied on Multilink Bundles and Constituent Links *(continued)*

Cos Component	Multilink Bundle	Constituent Links	Explanation
Virtual channel group	Yes	No	Virtual channel groups are identified through firewall filter rules that are applied on packets only before the multilink bundle. Thus you do not need to apply the virtual channel group configuration to the constituent links.

What Causes Jitter and Latency on the Multilink Bundle?

Problem—To test jitter and latency on a J-series device, I sent three streams of IP packets. All packets have the same IP precedence settings. After I configued LFI and CRTP, the latency increased even over a non-congested link. How can I reduce jitter and latency?

Solution—To reduce jitter and latency do the following:

1. Make sure that you have configured a shaping rate on each constituent link.
2. Make sure that you have not configured a shaping rate on the link services interface.
3. Make sure that the configured shaping rate value is equal to the physical interface bandwidth. For more information, see "Applying Shaping Rates to Interfaces" on page 263.
4. If shaping rates are configured correctly, and jitter still persists, contact the Juniper Networks Technical Assistance Center (JTAC). (See "Requesting Technical Support" on page xxxv.)

Are LFI and Load Balancing Working Correctly?

Problem—I have a single network that supports multiple services. My network transmits data and delay-sensitive voice traffic. I configured MLPPP and LFI to make sure that voice packets are transmitted across the network with very little delay and jitter. How can I find out if voice packets are being treated as LFI packets and load balancing is performed correctly?

Solution—When LFI is enabled, data (non-LFI) packets are encapsulated with an MLPPP header and fragmented to packets of a specified size. The delay-sensitive, voice (LFI) packets are PPP-encapsulated and interleaved between data packet fragments. Queuing and load balancing are performed differently for LFI and non-LFI packets. For more information, see "Load Balancing with LFI" on page 248.

To verify that LFI is performed correctly, determine that packets are fragmented and encapsulated as configured. After you know whether a packet is treated as an LFI packet or a non-LFI packet, you can confirm whether the load balancing is performed correctly.

Solution Scenario—Suppose two J-series devices R0 and R1 are connected by a multilink bundle ls-0/0/0.0 that aggregates two serial links, se-1/0/0 and se-1/0/1.

On R0 and R1, MLPPP and LFI are enabled on the link services interface and the fragmentation threshold is set to 128 bytes.

In this example, we used a packet generator to generate voice and data streams. You can use the packet capture feature to capture and analyze the packets on the incoming interface. For more information, see the *JUNOS Software Administration Guide*.

The following two data streams were sent on the multilink bundle:

- 100 data packets of 200 bytes (larger than the fragmentation threshold)
- 500 data packets of 60 bytes (smaller than the fragmentation threshold)

The following two voice streams were sent on the multilink bundle:

- 100 voice packets of 200 bytes from source port 100
- 300 voice packets of 200 bytes from source port 200

To confirm that LFI and load balancing are performed correctly, first verify that the link services interface is performing packet fragmentation as configured. Second, verify that the interface is encapsulating packets as configured. Finally, use the results to verify load balancing.

NOTE: Only the significant portions of command output are displayed and described in this example. For more information, see "Verifying the Link Services Interface Configuration" on page 271.

Step 1: Verifying Packet Fragmentation

From the CLI, enter the show interfaces ls-0/0/0 command, to check that large packets are fragmented correctly.

```
user@R0#> show interfaces ls-0/0/0
Physical interface: ls-0/0/0, Enabled, Physical link is Up
  Interface index: 136, SNMP ifIndex: 29
  Link-level type: LinkService, MTU: 1504
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Last flapped   : 2006-08-01 10:45:13 PDT (2w0d 06:06 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface ls-0/0/0.0 (Index 69) (SNMP ifIndex 42)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: Multilink-PPP
    Bandwidth: 16mbps
    Statistics         Frames      fps       Bytes        bps
    Bundle:
      Fragments:
        Input :             0        0           0          0
        Output:          1100        0      118800          0
      Packets:
        Input :             0        0           0          0
        Output:          1000        0      112000          0
```

```
...
 Protocol inet, MTU: 1500
   Flags: None
   Addresses, Flags: Is-Preferred Is-Primary
     Destination: 9.9.9/24, Local: 9.9.9.10
```

What It Means—The output shows a summary of packets transiting the device on the multilink bundle. Verify the following information on the multilink bundle:

- The total number of transiting packets = 1000
- The total number of transiting fragments = 1100
- The number of data packets that were fragmented = 100

The total number of packets sent (600 + 400) on the multilink bundle match the number of transiting packets (1000), indicating that no packets were dropped.

The number of transiting fragments exceeds the number of transiting packets by 100, indicating that 100 large data packets were correctly fragmented.

Corrective Action—If the packets are not fragmented correctly, check your fragmentation threshold configuration. Packets smaller than the specified fragmentation threshold are not fragmented. For information about configuring the fragmentation threshold, see "Configuring the Link Services Interface with a Configuration Editor" on page 253.

Step 2: Verifying Packet Encapsulation

To find out whether a packet is treated as an LFI or non-LFI packet, determine its encapsulation type. LFI packets are PPP encapsulated, and non-LFI packets are encapsulated with both PPP and MLPPP. PPP and MLPPP encapsulations have different overheads resulting in different-sized packets. You can compare packet sizes to determine the encapsulation type.

A small unfragmented data packet contains a PPP header and a single MLPPP header. In a large fragmented data packet, the first fragment contains a PPP header and an MLPPP header, but the consecutive fragments contain only an MLPPP header.

PPP and MLPPP encapsulations add the following number of bytes to a packet:

- PPP encapsulation adds 7 bytes:

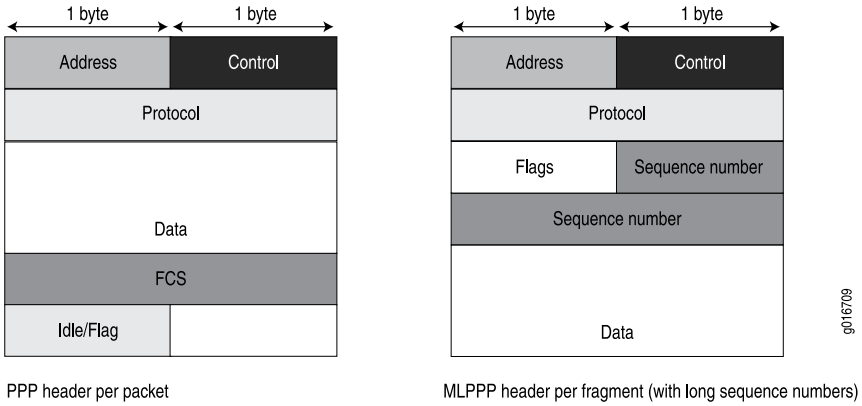
  4 bytes of header + 2 bytes of frame check sequence (FCS) + 1 byte that is idle or contains a flag

- MLPPP encapsulation adds between 6 and 8 bytes:

  4 bytes of PPP header + 2 to 4 bytes of multilink header

Figure 35 on page 283 shows the overhead added to PPP and MLPPP headers.

Figure 35: PPP and MLPPP Headers



For CRTP packets, the encapsulation overhead and packet size are even smaller than for an LFI packet. For more information, see “Configuring CRTP” on page 269.

Table 95 on page 283 shows the encapsulation overhead for a data packet and a voice packet of 70 bytes each. After encapsulation, the size of the data packet is larger than the size of the voice packet.

Table 95: PPP and MLPPP Encapsulation Overhead

Packet Type	Encapsulation	Initial Packet Size	Encapsulation Overhead	Packet Size after Encapsulation
Voice packet (LFI)	PPP	70 bytes	4 + 2 + 1 = 7 bytes	77 bytes
Data fragment (non-LFI) with short sequence	MLPPP	70 bytes	4 + 2 + 1 + 4 + 2 = 13 bytes	83 bytes
Data fragment (non-LFI) with long sequence	MLPPP	70 bytes	4 + 2 + 1 + 4 + 4 = 15 bytes	85 bytes

From the CLI, enter the show interfaces queue command to display the size of transmitted packet on each queue. Divide the number of bytes transmitted by the number of packets to obtain the size of the packets and determine the encapsulation type.

Step 3: Verifying Load Balancing

From the CLI, enter the show interfaces queue command on the multilink bundle and its constituent links to confirm whether load balancing is performed accordingly on the packets.

```
user@R0> show interfaces queue ls-0/0/0
Physical interface: ls-0/0/0, Enabled, Physical link is Up
  Interface index: 136, SNMP ifIndex: 29
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
```

```
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                600                 0 pps
    Bytes                :              44800                 0 bps
  Transmitted:
    Packets              :                600                 0 pps
    Bytes                :              44800                 0 bps
    Tail-dropped packets :                  0                 0 pps
    RED-dropped packets  :                  0                 0 pps
    …
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                  0                 0 pps
    Bytes                :                  0                 0 bps
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :                400                 0 pps
    Bytes                :              61344                 0 bps
  Transmitted:
    Packets              :                400                 0 pps
    Bytes                :              61344                 0 bps
    …
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                  0                 0 pps
    Bytes                :                  0                 0 bps
 …

user@RO> show interfaces queue se-1/0/0
Physical interface: se-1/0/0, Enabled, Physical link is Up
  Interface index: 141, SNMP ifIndex: 35
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                350                 0 pps
    Bytes                :              24350                 0 bps
  Transmitted:
    Packets              :                350                 0 pps
    Bytes                :              24350                 0 bps
   ..
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                  0                 0 pps
    Bytes                :                  0                 0 bps
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :                100                 0 pps
    Bytes                :              15272                 0 bps
  Transmitted:
    Packets              :                100                 0 pps
    Bytes                :              15272                 0 bps
    …
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                 19                 0 pps
    Bytes                :                247                 0 bps
  Transmitted:
    Packets              :                 19                 0 pps
```

```
    Bytes              :              247                0 bps
    …

user@R0> show interfaces queue se-1/0/1
Physical interface: se-1/0/1, Enabled, Physical link is Up
  Interface index: 142, SNMP ifIndex: 38
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :              350                0 pps
    Bytes                :            24350                0 bps
  Transmitted:
    Packets              :              350                0 pps
    Bytes                :            24350                0 bps
    …
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                0                0 pps
    Bytes                :                0                0 bps
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :              300                0 pps
    Bytes                :            45672                0 bps
  Transmitted:
    Packets              :              300                0 pps
    Bytes                :            45672                0 bps
    …
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :               18                0 pps
    Bytes                :              234                0 bps
  Transmitted:
    Packets              :               18                0 pps
    Bytes                :              234                0 bps
```

What It Means—The output from these commands shows the packets transmitted and queued on each queue of the link services interface and its constituent links. Table 96 on page 285 shows a summary of these values. (Because the number of transmitted packets equaled the number of queued packets on all the links, this table shows only the queued packets.)

Table 96: Number of Packets Transmitted on a Queue

Packets Queued	Bundle ls-0/0/0.0	Constituent Link se-1/0/0	Constituent Link se-1/0/1	Explanation
Packets on Q0	600	350	350	The total number of packets transiting the constituent links (350 + 350 = 700) exceeded the number of packets queued (600) on the multilink bundle.
Packets on Q2	400	100	300	The total number of packets transiting the constituent links equaled the number of packets on the bundle.

Table 96: Number of Packets Transmitted on a Queue *(continued)*

Packets Queued	Bundle ls-0/0/0.0	Constituent Link se-1/0/0	Constituent Link se-1/0/1	Explanation
Packets on Q3	0	19	18	The packets transiting Q3 of the constituent links are for keepalive messages exchanged between constituent links. Thus no packets were counted on Q3 of the bundle.

On the multilink bundle, verify the following:

- The number of packets queued matches the number transmitted. If the numbers match, no packets were dropped. If more packets were queued than were transmitted, packets were dropped because the buffer was too small. The buffer size on the constituent links controls congestion at the output stage. To correct this problem, increase the buffer size on the constituent links. For more information, see "Defining and Applying Scheduler Maps" on page 259.
- The number of packets transiting Q0 (600) matches the number of large and small data packets received (100 + 500) on the multilink bundle. If the numbers match, all data packets correctly transited Q0.
- The number of packets transiting Q2 on the multilink bundle (400) matches the number of voice packets received on the multilink bundle. If the numbers match, all voice LFI packets correctly transited Q2.

On the constituent links, verify the following:

- The total number of packets transiting Q0 (350 + 350) matches the number of data packets and data fragments (500 + 200). If the numbers match, all the data packets after fragmentation correctly transited Q0 of the constituent links.

  Packets transited both constituent links, indicating that load balancing was correctly performed on non-LFI packets.
- The total number of packets transiting Q2 (300 + 100) on constituent links matches the number of voice packets received (400) on the multilink bundle. If the numbers match, all voice LFI packets correctly transited Q2.

  LFI packets from source port 100 transited se-1/0/0, and LFI packets from source port 200 transited se-1/0/1. Thus all LFI (Q2) packets were hashed based on the source port and correctly transited both constituent links.

Corrective Action—If the packets transited only one link, take the following steps to resolve the problem:

1. Determine whether the physical link is up (operational) or down (unavailable). An unavailable link indicates a problem with the PIM, interface port, or physical connection (link-layer errors). If the link is operational, move to the next step.
2. Verify that the classifiers are correctly defined for non-LFI packets. Make sure that non-LFI packets are not configured to be queued to Q2. All packets queued to Q2 are treated as LFI packets.

3. Verify that at least one of the following values is different in the LFI packets: source address, destination address, IP protocol, source port, or destination port. If the same values are configured for all LFI packets, the packets are all hashed to the same flow and transit the same link.

Why Are Packets Dropped on a PVC Between a J-series Device and Another Vendor?

Problem—I configured a permanent virtual circuit (PVC) between T1, E1, T3, or E3 interfaces on my Juniper Networks device and another vendor's device, and packets are being dropped and ping fails.

Solution—If the other vendor's device does not have the same FRF.12 support as the J-series device or supports FRF.12 in a different way, the J-series interface on the PVC might discard a fragmented packet containing FRF.12 headers and count it as a "Policed Discard." As a workaround for this problem, configure multilink bundles on both peers, and configure fragmentation thresholds on the multilink bundles.

Chapter 11

Configuring uPIMs as Ethernet Switches

The 6-port, 8-port, and 16-port Gigabit Ethernet uPIMs can function as Ethernet access switches that switch traffic at Layer 2, in addition to routing traffic at Layer 3.

You use either the J-Web configuration editor or the CLI configuration editor to configure a uPIM as an Ethernet switch.

This chapter contains the following topics:

Gigabit Ethernet uPIM Switch Overview

You can deploy a J-series device with multiport uPIMs in branch offices as an access or desktop switch with integrated routing capability, thus eliminating intermediate access switch devices from your topology. The Gigabit Ethernet uPIM provides Ethernet switching while the Routing Engine provides routing functionality, enabling you to use a single chassis to provide routing, access switching, and WAN interfaces.

You can set a multiport uPIM to three modes of operation: routing (the default), switching, or enhanced switching. Routed traffic is forwarded from any port of the Gigabit Ethernet uPIM to the WAN interface. Switched traffic is forwarded from one port of the Gigabit Ethernet uPIM to another port on the same Gigabit Ethernet uPIM. Switched traffic is not forwarded from a port on one uPIM to a port on a different uPIM.

In routing mode, the multiport uPIM has the same configuration options as any other Gigabit Ethernet interface. To configure uPIM Gigabit Ethernet interfaces in routing mode, see “Configuring Gigabit Ethernet Interfaces—Quick Configuration” on page 86 and “Configuring Network Interfaces with a Configuration Editor” on page 102.

Switching mode

In switching mode, the uPIM appears in the list of interfaces as a single interface, which is the first interface on the uPIM—for example, ge-2/0/0. You can optionally configure each uPIM port only for autonegotiation, speed, and duplex mode. A uPIM in switching mode can perform the following functions:

- Layer 3 forwarding—Routes traffic destined for WAN interfaces and other PIMs present on the chassis.
- Layer 2 forwarding—Switches intra-LAN traffic from one host on the LAN to another LAN host (one port of uPIM to another port of same uPIM).

Connecting uPIMs in a Daisy-Chain

You cannot combine multiple uPIMs to act as a single integrated switch. However, you can connect uPIMs on the same chassis externally by physically connecting a port on one uPIM to a port on another uPIM in a daisy-chain fashion.

Two or more uPIMs daisy-chained together create a single switch with a higher port count than either individual uPIM. One port on each uPIM is used solely for the connection. For example, if you daisy-chain a 6-port uPIM and an 8-port uPIM, the result operates as a 12-port uPIM. Any port of a uPIM can be used for daisy-chaining.

Configure the IP address for only one of the daisy-chained uPIMs, making it the primary uPIM. The secondary uPIM routes traffic to the primary uPIM, which forwards it to the Routing Engine. This results in some increase in latency and packet drops due to oversubscription of the external link.

Only one link between the two uPIMs is supported. Connecting more than one link between uPIMs creates a loop topology, which is not supported.

Enhanced Switching Mode

In enhanced switching mode, each port can be configured for switching or routing mode. This usage differs from the routing and switching modes, in which all ports must be in either switching or routing mode. The uPIM in enhanced switching mode provides the following features:

- Supports configuration of different types of VLANs and inter-VLAN routing
- Supports Layer 2 control plane protocols such as Spanning Tree Protocol (STP) and Link Aggregation Control Protocol (LACP)
- Supports Port-based Network Access Control (PNAC) by means of authentication servers

NOTE: You can configure uPIM in enhanced switching mode only in 9.2 Release or later.

Link Aggregation

You can combine multiple physical Ethernet ports to form a logical point-to-point link, known as a link aggregation group (LAG) or bundle. A LAG provides more bandwidth than a single Ethernet link can provide. Additionally, link aggregation provides network redundancy by load-balancing traffic across all available links. If one of the links should fail, the system automatically load-balances traffic across all remaining links. You can select up to eight Ethernet interfaces and include them within a link aggregation group.

Link aggregation can be used for point-to-point connections. It balances traffic across the member links within an aggregated Ethernet bundle and effectively increases the uplink bandwidth. Another advantage of link aggregation is increased availability, because the LAG is composed of multiple member links. If one member link fails, the LAG continues to carry traffic over the remaining links.

This section contains the following topics:

Link Aggregation Group (LAG)

You can configure a LAG by specifying the link number as a physical device and then associating a set of ports with the link. All the ports must have the same speed and be in full-duplex mode. JUNOS software for Gigabit Ethernet uPIM switches assigns a unique ID and port priority to each port.

The ID and priority are not configurable. When configuring a LAG, consider the following guidelines:

- Up to 8 Ethernet ports can be created in each bundle.
- Up to 128 LAGs are supported in a virtual chassis configuration.
- Each LAG must be configured on both sides of the link.
- The ports on either side of the link must be set to the same speed.

A typical deployment for a LAG would be to aggregate trunk links between an access switch and a distribution switch or customer edge (CE) device. LAGs are not supported on virtual chassis port links. LAGs can only be used for a point-to-point connection. Atleast one end of the LAG should be configured as Active

Link Aggregation Control Protocol (LACP)

LACP, a subcomponent of IEEE 802.3ad, provides additional functionality for LAGs.

About enabling LACP:

- When LACP is not enabled, a local LAG might attempt to transmit packets to a remote single interface, which causes the communication to fail.
- When LACP is enabled, a local LAG cannot transmit packets unless a LAG with LACP is also configured on the remote end of the link.

By default, Ethernet links do not exchange protocol data units (PDUs), which contain information about the state of the link. You can configure Ethernet links to actively transmit PDUs, or you can configure the links to passively transmit them, sending out LACP PDUs only when they receive them from another link. The transmitting link is known as the actor and the receiving link is known as the partner.

NOTE: Presently, LACP can be configured only for Ethernet Switching family.

IGMP Snooping

Internet Group Management Protocol (IGMP) snooping regulates multicast traffic in a switched network. With IGMP snooping enabled, a LAN switch monitors the IGMP transmissions between a host (a network device) and a multicast router, keeping track of the multicast groups and associated member ports. The switch uses that information to make intelligent multicast-forwarding decisions and forward traffic to the intended destination interfaces. J-series devices support IGMPv1 and IGMPv2.

This section contains the following topics:

How IGMP Snooping Works

A J-series device usually learns *unicast* MAC addresses by checking the source address field of the frames it receives. However, a *multicast* MAC address can never be the source address for a packet. As a result, the switch floods multicast traffic on the VLAN, consuming significant amounts of bandwidth.

IGMP snooping regulates multicast traffic on a VLAN to avoid flooding. When IGMP snooping is enabled, the switch intercepts IGMP packets and uses the content of the packets to build a multicast cache table. The cache table is a database of multicast groups and their corresponding member ports. The cache table is then used to regulate multicast traffic on the VLAN.

When the router receives multicast packets, it uses the cache table to selectively forward the packets only to the ports that are members of the destination multicast group.

How Hosts Join and Leave Multicast Groups

Hosts can join multicast groups in either of two ways:

- By sending an unsolicited IGMP join message to a multicast router that specifies the IP multicast that the host is attempting to join.
- By sending an IGMP join message in response to a general query from a multicast router.

A multicast router continues to forward multicast traffic to a VLAN provided that at least one host on that VLAN responds to the periodic general IGMP queries. For a

host to remain a member of a multicast group, therefore, it must continue to respond to the periodic general IGMP queries.

To leave a multicast group, a host can either not respond to the periodic general IGMP queries, which results in a "silent leave" (the only leave option for hosts connected to switches running IGMPv1), or send a group-specific IGMPv2 leave message.

Configuring Gigabit Ethernet uPIM Switches

When you set a multiport uPIM to switching mode, the uPIM appears as a single entity for monitoring purposes. The only physical port settings that you can configure are autonegotiation, speed, and duplex mode on each uPIM port, and these settings are optional.

NOTE: You cannot configure switch ports from J-Web Quick Configuration pages. You must use the J-Web or CLI configuration editor.

To configure a multiport Gigabit Ethernet uPIM as a switch:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 97 on page 293.
3. If you are finished configuring the device, commit the configuration.
4. To verify the configuration, see "Verifying Gigabit Ethernet uPIM Switch Configuration" on page 294.

Table 97: Configuring uPIMs as Switches

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Chassis** level of the configuration hierarchy	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Chassis, click **Configure**.	From the [edit] hierarchy level, enter edit chassis
Set the uPIM mode of operation to switching. **NOTE:** Routing mode is the default setting.	1. Next to Fpc, click **Add new entry**. 2. In the Slot field, enter the number of the slot of the chassis in which the uPIM is inserted, and click **OK**. 3. Next to Pic, click **Add new Entry**. 4. Enter 0 in the Slot field. (This number is always 0 on a J-series device.) 5. Next to Ethernet, click **Configure**. 6. From the Pic mode list, choose **switching** and click **OK**.	Enter set pim *pim-number* pic 0 ethernet pic-mode switching

Table 97: Configuring uPIMs as Switches *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
(Optional) Set the uPIM mode of operation to enhanced switching. **NOTE:** Routing mode is the default setting.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Chassis, click **Configure**. 3. Next to Fpc, click **Add new entry**. 4. In the Slot field, enter the number of the slot of the chassis in which the uPIM is inserted, and click **OK**. 5. Next to Pic, click **Add new Entry**. 6. Enter 0 in the Slot field. (This number is always 0 on a J-series device.) 7. Next to Ethernet, click **Configure**. 8. From the Pic mode list, choose **enhanced-switching** and click **OK**.	Enter set fpc *fpc-number* pic 0 ethernet pic-mode enhanced-switching
(Optional) Set the physical port parameters for each port on the uPIM.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Edit**. 3. Click the name of the uPIM interface—for example ge-2/0/0. 4. Next to Switch options , click **Configure**. 5. Next to Switch port, click **Add new entry**. 6. In the Port field, enter the number of the port you want to configure. 7. Choose the settings for Autonegotiation, Link mode, and Speed, and click **OK**.	1. From the [edit] hierarchy level, enter edit interfaces 2. Configure parameters for each uPIM port that you want to specify: ge-*pim*/0/0 switch-port *port-numer* (auto-negotiation | no-auto-negotiation) speed (10m | 100m | 1g) link-mode (full-duplex | half-duplex) For example: set ge-2/0/0 switch-port 1 autonegotiation

Verifying Gigabit Ethernet uPIM Switch Configuration

The operational mode command for checking the status and statistics for multiport uPIMs switching mode is different from that of routing mode. For uPIMs in routing mode, the operational commands are the same as for other Gigabit Ethernet interfaces, such as the 1-port Gigabit Ethernet ePIM and built-in Gigabit Ethernet ports.

Not all operational mode commands are supported for ports of a uPIM in switching mode. For example, the operational mode command for monitoring port statistics is not supported.

NOTE: To clear the statistics for the individual switch ports using the clear interfaces statistics ge-*pim*/0/0 switch-port *port-number* command.

Verifying Status of uPIM Switch Ports

Purpose To verify the status and view statistics for a port on a uPIM in switching mode.

Action From the CLI, enter the show interfaces ge-*pim*/0/0 switch-port *port-number* command.

Sample Output

```
user@host show interfaces ge-pim/0/0 switch-port port-number
Port 0, Physical link is Up
        Speed: 100mbps, Auto-negotiation: Enabled
   Statistics:                          Receive         Transmit
          Total bytes                    28437086         21792250
          Total packets                    409145            88008
          Unicast packets                    9987            83817
          Multicast packets                145002                0
          Broadcast packets                254156             4191
          Multiple collisions                  23               10
          FIFO/CRC/Align errors                 0                0
          MAC pause frames                      0                0
          Oversized frames                      0
          Runt frames                           0
          Jabber frames                         0
          Fragment frames                       0
          Discarded frames                      0
        Autonegotiation information:
          Negotiation status: Complete
          Link partner:
              Link mode: Full-duplex, Flow control: None, Remote fault: OK, Link
 partner Speed: 100 Mbps
          Local resolution:
              Flow control: None, Remote fault: Link OK
```

Part 3
Configuring Routing Protocols

Chapter 12

Routing Overview

Routing is the process of delivering a message across a network or networks. This process has two primary components: the exchange of routing information to forward packets accurately from source to destination and the packet-forwarding procedure.

NOTE: Before configuring routing protocols, you must first configure security filters. For more information, see the *JUNOS Software Security Configuration Guide*.

To use the routing capabilities of a Juniper Networks device, you must understand the fundamentals of IP routing and the routing protocols that are primarily responsible for the transmission of unicast traffic. To read this chapter, you need a basic understanding of IP addressing and TCP/IP.

NOTE: When configuring IPv6 addressing and routing on a J-series Service Router, you must enable IPv6 in secure context. For information about IPv6, see the *JUNOS Routing Protocols Configuration Guide*.

This chapter includes the following topics. For more information, see the *JUNOS Routing Protocols Configuration Guide*.

Routing Terms

To understand routing, become familiar with the terms defined in Table 98 on page 300.

Table 98: Routing Terms

Term	Definition
adjacency	Portion of the local routing information that pertains to the reachability of a single neighbor over a single circuit or interface.
area	Administrative group of OSPF networks within an autonomous system (AS) that operates independently from other areas in the AS. Multiple areas within an AS reduce the amount of link-state advertisement (LSA) traffic on the network and the size of topology databases.
area border router (ABR)	In OSPF, a router having interfaces in multiple areas of an autonomous system (AS) so that it can link the areas to each other. An area border router maintains a separate topological database for each area it is connected to and shares topology information between areas.
AS path	In BGP, the list of autonomous system (ASs) that a packet must traverse to reach a given set of destinations within a single AS.
autonomous system (AS)	Network, collection of routers, or portion of a large internetwork under a single administrative authority.
backbone area	In OSPF, the central area in an autonomous system (AS) to which all other areas are connected by area border routers (ABRs). The backbone area always has the area ID 0.0.0.0.
bidirectional connectivity	Ability of directly connected devices to communicate with each other over the same link.
Border Gateway Protocol (BGP)	Exterior gateway protocol used to exchange routing information among devices in different autonomous systems.
broadcast	Operation of sending network traffic from one network node to all other network nodes.
cluster	In BGP, a set of devices that have been grouped together. A cluster consists of one system that acts as a route reflector, along with any number of client peers. The client peers receive their route information only from the route reflector system. Devices in a cluster do not need to be fully meshed.
confederation	In BGP, a group of autonomous systems (ASs) that appears to external ASs to be a single AS.
confederation sequence	Ordered set of autonomous systems (ASs) for a confederation. The closest AS in the path is first in the sequence.
convergence	After a topology change, the time all the routers in a network take to receive the information and update their routing tables.
cost	Unitless number assigned to a path between neighbors, based on throughput, round-trip time, and reliability. The sum of path costs between source and destination hosts determines the overall path cost. OSPF uses the lowest cost to determine the best path.
designated router (DR)	In OSPF, a node designated to process link-state advertisements (LSAs) and distribute topology updates for an autonomous system (AS).
distance vector	Number of hops to a routing destination.
dynamic routing	Routing method that enables the route of a message through a network to change as network conditions change. Compare *static routing*.
end systems	Network entities that send and receive packets.

Table 98: Routing Terms *(continued)*

Term	Definition
exterior gateway protocol (EGP)	Protocol that exchanges routing information between autonomous systems (ASs). BGP is an EGP. Compare *interior gateway protocol (IGP)*.
external BGP (EBGP)	BGP configuration in which sessions are established between devices in different autonomous systems (ASs).
external peer	In BGP, a peer that resides in a different autonomous system (AS) from the Juniper Networks device.
external route	Route to an area outside the network.
flooding	Technique by which a router forwards traffic to every node attached to the router, except the node from which the traffic arrived. Flooding is a simple but sometimes inefficient way to distribute routing information quickly to every node in a network. RIP and OSPF are flooding protocols, but BGP is not.
forwarding table	JUNOS software forwarding information base (FIB). The JUNOS routing protocol process installs active routes from its routing tables into the Routing Engine forwarding table. The kernel copies this forwarding table into the Packet Forwarding Engine, which is responsible for determining which interface transmits the packets.
full mesh	Network in which devices are organized in a mesh topology, with each node connected to every other network node.
gateway router	Node on a network that serves as an entrance to another network.
global AS	Global autonomous system (AS). An AS consisting of multiple subautonomous systems (sub-ASs).
handshake	Process of exchanging signaling information between two communications devices to establish the method and transmission speed of a connection.
hello packet	In OSPF, a packet sent periodically by a router to first establish and then maintain network adjacency, and to discover neighbor routers.
hold time	Maximum number of seconds allowed to elapse between the time a BGP system receives successive keepalive or update messages from a peer.
hop	Trip a data packet takes from one router to another in the network. The number of routers through which a packet passes to get from its source to its destination is known as the hop count. In general, the best route is the one with the shortest hop count.
intermediate systems	Network entities that relay (forward) packets as well as send and receive them on the network. Intermediate systems are also known as routers.
Intermediate System-to-Intermediate System (IS-IS)	Link-state, interior gateway routing protocol for IP networks that also uses the shortest-path-first (SPF) algorithm to determine routes.
interior gateway protocol (IGP)	Protocol that exchanges routing information within autonomous systems (ASs). IS-IS, OSPF, and RIP are IGPs. Compare *exterior gateway protocol (EGP)*.
Internal BGP (IBGP)	BGP configuration in which sessions are established between routers in the same autonomous systems (ASs).
internal peer	In BGP, a peer that resides in the same autonomous system (AS) as the Juniper Networks device.

Table 98: Routing Terms *(continued)*

Term	Definition
keepalive message	Periodic message sent by one BGP peer to another to verify that the session between them is still active.
latency	Delay that occurs when a packet or signal is transmitted over a communications system.
link-state advertisement (LSA)	Messages that announce the presence of OSPF-enabled interfaces to adjacent OSPF interfaces (neighbors). The exchange of LSAs establishes bidirectional connectivity between neighbors.
local preference	Optional BGP path attribute carried in internal BGP update packets that indicates the degree of preference for an external route.
mesh	Network topology in which devices are organized in a manageable, segmented manner with many, often redundant, interconnections between network nodes. See also *full mesh*.
metric	Numerical value that determines how quickly a packet can reach its destination. See also *cost*.
multiple exit discriminator (MED)	Optional BGP path attribute consisting of a metric value that is used to determine the exit point to a destination when all other factors in determining the exit point are equal.
neighbor	Adjacent router interface. A node can directly route packets to its neighbors only. See also *peer*.
network	Series of nodes interconnected by communication paths.
network diameter	Maximum hop count in a network.
network topology	Arrangement of nodes and connections in a network.
node	Connection point that operates as a redistribution point or an end point in a network, recognizing data transmissions and either forwarding or processing them.
notification message	Message sent between BGP peers to inform the receiving peer that the sending peer is terminating the session because an error occurred, and explaining the error.
not-so-stubby area (NSSA)	In OSPF, a type of stub area in which external route advertisements can be flooded.
open message	Message sent between BGP peers to establish communication.
Open Shortest Path First protocol (OSPF)	A link-state interior gateway protocol (IGP) that makes routing decisions based on the shortest-path-first (SPF) algorithm (also referred to as the Dijkstra algorithm).
origin	Value assigned to a BGP route to indicate whether the first router to advertise the route learned it from an external, internal, or unknown source.
path-vector protocol	Protocol that uses the path between autonomous systems (ASs) to select the best route, rather than the shortest distance or the characteristics of the route (link state). BGP is a path-vector protocol. In contrast, RIP is a distance-vector protocol, and OSPF and IS-IS are link-state protocols.
peer	Immediately adjacent router with which a protocol relationship has been established. See also *neighbor*.
peering	The practice of exchanging Internet traffic with directly connected peers according to commercial and contractual agreements.
point of presence (POP)	Access point to the Internet, having a unique IP address, where telecommunications equipment is located. POPs usually belong to Internet service providers (ISPs) or telephone companies.

Table 98: Routing Terms *(continued)*

Term	Definition
poison reverse	An efficiency technique in a RIP network. By setting the number of hops to an unavailable router to 16 hops or more, a router informs all the other routers in the network. Because RIP allows only up to 15 hops to another router, this technique reduces RIP updates and helps defeat large routing loops. See also *split horizon*.
propagation	Process of translating and forwarding route information discovered by one routing protocol in the update messages of another routing protocol. Route propagation is also called route redistribution.
reachability	In BGP, the feasibility of a route.
round-robin	Scheduling algorithm in which items have the same priority and are handled in a fixed cyclic order.
route advertisement	Distribution of routing information at specified intervals throughout a network, to establish adjacencies with neighbors and communicate usable routes to active destinations. See also *link-state advertisement (LSA)*.
route aggregation	Combining groups of routes with common addresses into a single entry in the routing table, to decrease routing table size and the number of route advertisements sent by a router.
route reflection	In BGP, configuring a group of routers into a cluster and having one system act as a route reflector, redistributing routes from outside the cluster to all routers in the cluster. Routers in a cluster do not need to be fully meshed.
Routing Information Protocol (RIP)	Distance-vector routing protocol that keeps a database of routing information gathered from periodic broadcasts by each router in a network.
Routing Information Protocol next generation (RIPng)	Distance-vector routing protocol that exchanges routing information used to compute routes and is intended for Internet Protocol version 6 (IPv6)-based networks.
routing table	Table stored on a router that keeps track of all possible paths (routes) between sources and destinations in a network and, in some cases, metrics associated with the routes.
split horizon	An efficiency technique in a RIP network. A router reduces the number of RIP updates in the network by not retransmitting a route advertisement out the interface through which it was received. Split-horizon updates also help prevent routing loops. See also *poison reverse*.
static routing	Routing method in which routes are manually entered in the routing table and do not change unless you explicitly update them. Unlike dynamic routes, which must be imported into the routing table each time a host comes online, static routes are available immediately. Static routes are generally preferred over other types of routes. Compare *dynamic routing*.
stub area	In OSPF, an area through which or into which autonomous system (AS) external route advertisements are not flooded.
subautonomous system (sub-AS)	Autonomous system (AS) members of a BGP confederation.
subnetwork	Subdivision of a network, which functions exactly like a network except that it has a more specific address and subnet mask (destination prefix).
three-way handshake	Process by which two routers synchronize protocols and establish a bidirectional connection.

Table 98: Routing Terms *(continued)*

Term	Definition
topology database	Map of connections between the nodes in a network. The topology database is stored in each node.
triggered update	In a network that uses RIP, a routing update that is automatically sent whenever routing information changes.
virtual link	In OSPF, a link you create between two area border routers (ABRs) that have an interface to a common nonbackbone area, to connect a third area to the backbone area. One of the area border routers must be directly connected to the backbone area.

Routing Overview

Routing is the transmission of data packets from a source to a destination address. For packets to be correctly forwarded to the appropriate host address, the host must have a unique numeric identifier or IP address. The unique IP address of the destination host forms entries in the routing table. These entries are primarily responsible for determining the path that a packet traverses when transmitted from source to destination.

This overview contains the following topics:

Networks and Subnetworks

Large groups of machines that are interconnected and can communicate with one another form networks. Typically, networks identify large systems of computers and devices that are owned or operated by a single entity. Traffic is routed between or through the networks as data is passed from host to host.

As networks grow large, the ability to maintain the network and effectively route traffic between hosts within the network becomes increasingly difficult. To accommodate growth, networks are divided into subnetworks. Fundamentally, subnetworks behave exactly like networks, except that they are identified by a more specific network address and subnet mask (destination prefix). Subnetworks have routing gateways and share routing information in exactly the same way as large networks.

Autonomous Systems

A large network or collection of routers under a single administrative authority is termed an autonomous system (AS). Autonomous systems are identified by a unique numeric identifier that is assigned by the Internet Assigned Numbers Authority (IANA). Typically, the hosts within an AS are treated as internal peers, and hosts in a peer AS are treated as external peers. The status of the relationship between hosts—internal or external—governs the protocol used to exchange routing information.

Interior and Exterior Gateway Protocols

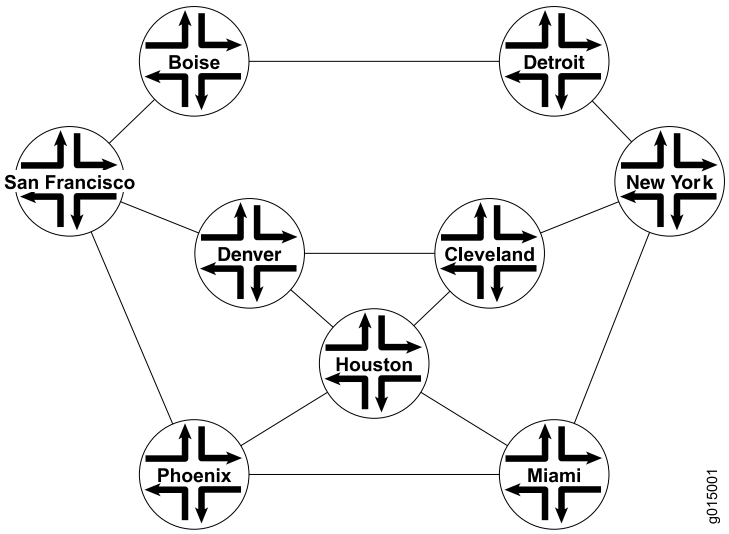
Routing information that is shared within an AS is transmitted by an interior gateway protocol (IGP). Of the different IGPs, the most common are RIP, OSPF, and IS-IS. IGPs are designed to be fast acting and light duty. They typically incorporate only a moderate security system, because trusted internal peers do not require the stringent security measures that untrusted peers require. As a result, you can usually begin routing within an AS by enabling the IGP on all internal interfaces and performing minimal additional configuration. You do not need to establish individual adjacencies.

Routing information that is shared with a peer AS is transmitted by an exterior gateway protocol (EGP). The primary EGP in use in almost all networks is the Border Gateway Protocol (BGP). BGP is designed to be very secure. Individual connections must be explicitly configured on each side of the link. As a result, although large numbers of connections are difficult to configure and maintain, each connection is secure.

Routing Tables

To route traffic from a source host to a destination host, the devices through which the traffic will pass must learn the path that the packet is to take. Once learned, the information is stored in routing tables. The routing table maintains a list of all the possible paths from point A to point B. Figure 36 on page 306 shows a simple network of routers.

Figure 36: Simple Network Topology



This simple network provides multiple ways to get from Host San Francisco to Host Miami. The packet can follow the path through Denver and Cleveland. Alternatively, the packet can be routed through Phoenix and directly to Miami. The routing table includes all the possible paths and combinations—an exhaustive list of all the ways to get from the source to the destination.

The routing table must include every possible path from a source to a destination. Routing tables for the network in Figure 36 on page 306 must include entries for San Francisco-Denver, San Francisco-Cleveland, San Francisco-Miami, Denver-Cleveland, and so on. As the number of sources and destinations increases, the routing table quickly becomes large. The unwieldy size of routing tables in the primary reason for the division of networks into subnetworks.

Forwarding Tables

If the routing table is a list of all the possible paths a packet can take, the forwarding table is a list of only the best routes to a particular destination. The best path is determined according to the particular routing protocol being used, but generally the number of hops between the source and destination determines the best possible route.

In the network shown in Figure 36 on page 306, because the path with the fewest number of hops from San Francisco to Miami is through Phoenix, the forwarding table distills all the possible San Francisco-Miami routes into the single route through Phoenix. All traffic with a destination address of Miami is sent directly to the next hop, Phoenix.

After it receives a packet, the Phoenix router performs another route lookup, using the same destination address. The Phoenix router then routes the packet appropriately. Although it considers the entire path, the router at any individual hop along the way is responsible only for transmitting the packet to the next hop in the path. If the Phoenix router is managing its traffic in a particular way, it might send the packet through Houston on its route to Miami. This scenario is likely if specific

customer traffic is treated as priority traffic and routed through a faster or more direct route, while all other traffic is treated as nonpriority traffic.

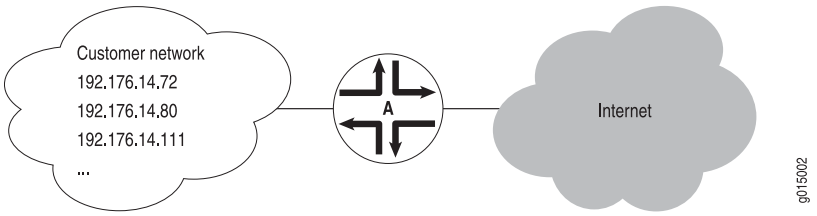
Dynamic and Static Routing

Entries are imported into a router's routing table from dynamic routing protocols or by manual inclusion as static routes. Dynamic routing protocols allow routers to learn the network topology from the network. The routers within the network send out routing information in the form of route advertisements. These advertisements establish and communicate active destinations, which are then shared with other routers in the network.

Although dynamic routing protocols are extremely useful, they have associated costs. Because they use the network to advertise routes, dynamic routing protocols consume bandwidth. Additionally, because they rely on the transmission and receipt of route advertisements to build a routing table, dynamic routing protocols create a delay (latency) between the time a router is powered on and the time during which routes are imported into the routing table. Some routes are therefore effectively unavailable until the routing table is completely updated, when the router first comes online or when routes change within the network (due to a host going offline, for example).

Static routing avoids the bandwidth cost and route import latency of dynamic routing. Static routes are manually included in the routing table, and never change unless you explicitly update them. Static routes are automatically imported into the routing table when a router first comes online. Additionally, all traffic destined for a static address is routed through the same router. This feature is particularly useful for networks with customers whose traffic must always flow through the same routers. Figure 37 on page 307 shows a network that uses static routes.

Figure 37: Static Routing Example



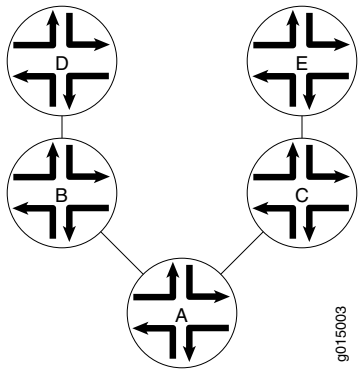
In Figure 37 on page 307, the customer routes in the 192.176.14/24 subnetwork are static routes. These are hard links to specific customer hosts that never change. Because all traffic destined for any of these routes is forwarded through Router A, these routes are included as static routes in Router A's routing table. Router A then advertises these routes to other hosts so that traffic can be routed to and from them.

Route Advertisements

The routing table and forwarding table contain the routes for the routers within a network. These routes are learned through the exchange of route advertisements. Route advertisements are exchanged according to the particular protocol being employed within the network.

Generally, a router transmits hello packets out each of its interfaces. Neighboring routers detect these packets and establish adjacencies with the router. The adjacencies are then shared with other neighboring routers, which allows the routers to build up the entire network topology in a topology database, as shown in Figure 38 on page 308.

Figure 38: Route Advertisement



In Figure 38 on page 308, Router A sends out hello packets to each of its neighbors. Routers B and C detect these packets and establish an adjacent relationship with Router A. Router B and C then share this information with their neighbors, Routers D and E, respectively. By sharing information throughout the network, the routers create a network topology, which they use to determine the paths to all possible destinations within the network. The routes are then distilled into the forwarding table of best routes according to the route selection criteria of the protocol in use.

Route Aggregation

As the number of hosts in a network increases, the routing and forwarding tables must establish and maintain more routes. As these tables become larger, the time routers require to look up particular routes so that packets can be forwarded becomes prohibitive. The solution to the problem of growing routing tables is to group (aggregate) the routers by subnetwork, as shown in Figure 39 on page 309.

Figure 39: Route Aggregation

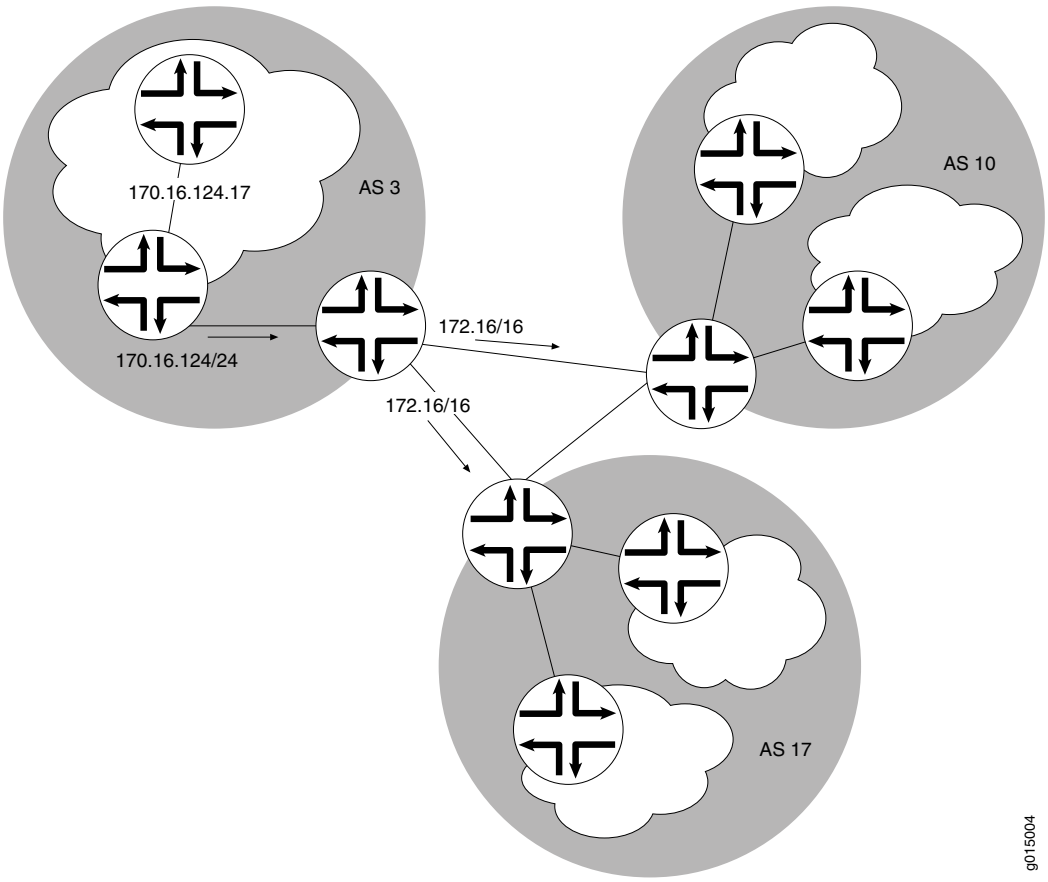


Figure 39 on page 309 shows three different ASs. Each AS contains multiple subnetworks with thousands of host addresses. To allow traffic to be sent from any host to any host, the routing tables for each host must include a route for each destination. For the routing tables to include every combination of hosts, the flooding of route advertisements for each possible route becomes prohibitive. In a network of hosts numbering in the thousands or even millions, simple route advertisement is not only impractical but impossible.

By employing route aggregation, instead of advertising a route for each host in AS 3, the gateway router advertises only a single route that includes all the routes to all the hosts within the AS. For example, instead of advertising the particular route 170.16.124.17, the AS 3 gateway router advertises only 170.16/16. This single route advertisement encompasses all the hosts within the 170.16/16 subnetwork, which reduces the number of routes in the routing table from 2^{16} (one for every possible IP address within the subnetwork) to 1. Any traffic destined for a host within the AS is forwarded to the gateway router, which is then responsible for forwarding the packet to the appropriate host.

Similarly, in this example, the gateway router is responsible for maintaining 2^{16} routes within the AS (in addition to any external routes). The division of this AS into subnetworks allows for further route aggregation to reduce this number. In the

subnetwork in the example, the subnetwork gateway router advertises only a single route (170.16.124/24), which reduces the number of routes from 2^8 to 1.

RIP Overview

In a Routing Information Protocol (RIP) network, each router's forwarding table is distributed among the nodes through the flooding of routing table information. Because topology changes are flooded throughout the network, every node maintains the same list of destinations. Packets are then routed to these destinations based on path-cost calculations done at each node in the network.

For an overview of RIPng, see "RIPng Overview" on page 314. For configuration instructions, see the *JUNOS Routing Protocols Configuration Guide*.

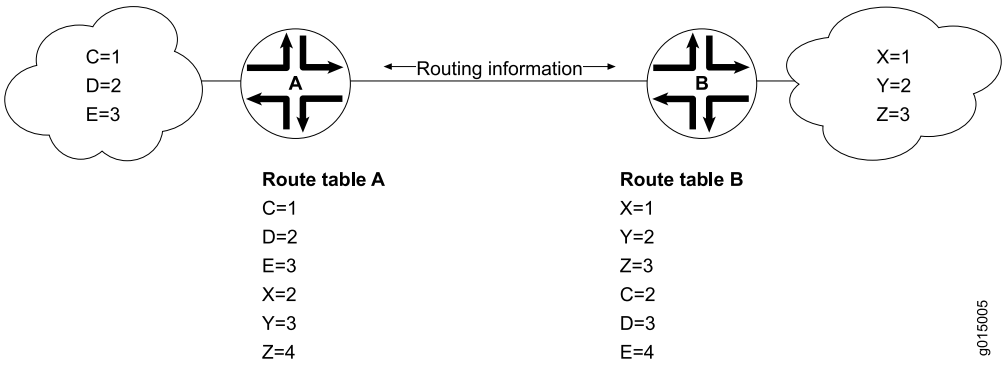
This overview contains the following topics:

NOTE: In general, in this guide, the term *RIP* refers to RIP version 1 and RIP version 2.

Distance-Vector Routing Protocols

Distance-vector routing protocols transmit routing information that includes a distance vector, typically expressed as the number of hops to the destination. This information is flooded out all protocol-enabled interfaces at regular intervals (every 30 seconds in the case of RIP) to create a network map that is stored in each node's local topology database. Figure 40 on page 310 shows how distance-vector routing works.

Figure 40: Distance-Vector Protocol



In Figure 40 on page 310, Routers A and B have RIP enabled on adjacent interfaces. Router A has known RIP neighbors Routers C, D, and E, which are 1, 2, and 3 hops away, respectively. Router B has known RIP neighbors Routers X, Y, and Z, which are 1, 2, and 3 hops away, respectively. Every 30 seconds, each router floods its entire routing table information out all RIP-enabled interfaces. In this case, flooding exchanges routing table information across the RIP link.

When Router A receives routing information from Router B, it adds 1 to the hop count to determine the new hop count. For example, Router X has a hop count of 1, but when Router A imports the route to X, the new hop count is 2. The imported route also includes information about where the route was learned, so that the original route is imported as a route to Router X through Router B with a hop count of 2.

When multiple routes to the same host are received, RIP uses the distance-vector algorithm to determine which path to import into the forwarding table. The route with the smallest hop count is imported. If there are multiple routes with the same hop count, all are imported into the forwarding table, and traffic is sent along the paths in round-robin fashion.

Maximizing Hop Count

The successful routing of traffic across a RIP network requires that every node in the network maintain the same view of the topology. Topology information is broadcast between RIP neighbors every 30 seconds. If Router A is many hops away from a new host, Router B, the route to B might take significant time to propagate through the network and be imported into Router A's routing table. If the two routers are 5 hops away from each other, Router A cannot import the route to Router B until 2.5 minutes after Router B is online. For large numbers of hops, the delay becomes prohibitive. To help prevent this delay from growing arbitrarily large, RIP enforces a maximum hop count of 15 hops. Any prefix that is more than 15 hops away is treated as unreachable and assigned a hop count equal to infinity. This maximum hop count is called the network diameter.

RIP Packets

Routing information is exchanged in a RIP network by RIP request and RIP response packets. A router that has just booted can broadcast a RIP request on all RIP-enabled interfaces. Any routers running RIP on those links receive the request and respond by sending a RIP response packet immediately to the router. The response packet contains the routing table information required to build the local copy of the network topology map.

In the absence of RIP request packets, all RIP routers broadcast a RIP response packet every 30 seconds on all RIP-enabled interfaces. The RIP broadcast is the primary way in which topology information is flooded throughout the network.

Once a router learns about a particular destination through RIP, it starts a timer. Every time it receives a new response packet with information about the destination, the router resets the timer to zero. However, if the router receives no updates about a particular destination for 180 seconds, it removes the destination from its RIP routing table.

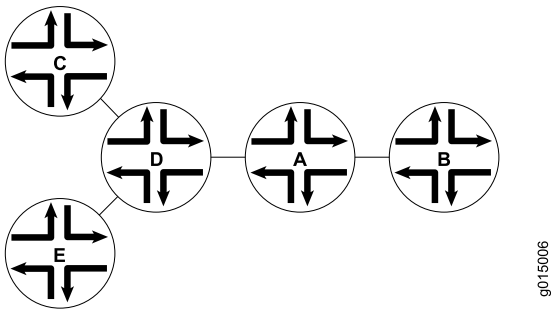
In addition to the regular transmission of RIP packets every 30 seconds, if a router detects a new neighbor or detects that an interface is unavailable, it generates a triggered update. The new routing information is immediately broadcast out all RIP-enabled interfaces, and the change is reflected in all subsequent RIP response packets.

Split Horizon and Poison Reverse Efficiency Techniques

Because RIP functions by periodically flooding the entire routing table out to the network, it generates a lot of traffic. The split horizon and poison reverse techniques can help reduce the amount of network traffic originated by RIP hosts and make the transmission of routing information more efficient.

If a router receives a set of route advertisements on a particular interface, RIP determines that those advertisements do not need to be retransmitted out the same interface. This technique, known as split horizon, helps limit the amount of RIP routing traffic by eliminating information that other neighbors on that interface have already learned. Figure 41 on page 312 shows an example of the split horizon technique.

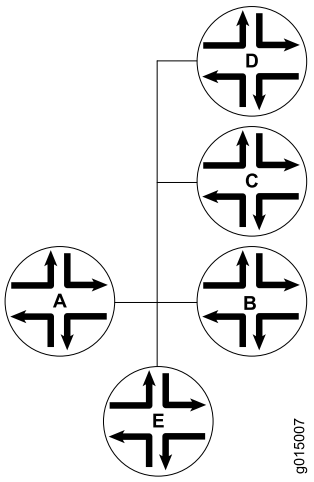
Figure 41: Split Horizon Example



In Figure 41 on page 312, Router A advertises routes to Routers C, D, and E to Router B. In this example, Router A can reach Router C in 2 hops. When Router A advertises the route to Router B, B imports it as a route to Router C through Router A in 3 hops. If Router B then readvertised this route to Router A, A would import it as a route to Router C through Router B in 4 hops. However, the advertisement from Router B to Router A is unnecessary, because Router A can already reach the route in 2 hops. The split horizon technique helps reduce extra traffic by eliminating this type of route advertisement.

Similarly, the poison reverse technique helps to optimize the transmission of routing information and improve the time to reach network convergence. If Router A learns about unreachable routes through one of its interfaces, it advertises those routes as unreachable (hop count of 16) out the same interface. Figure 42 on page 313 shows an example of the poison reverse technique.

Figure 42: Poison Reverse Example

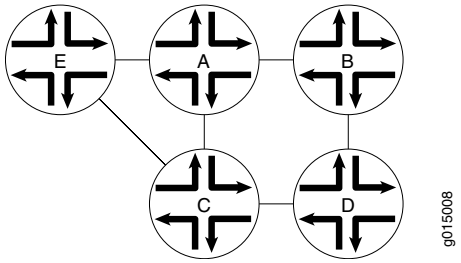


In Figure 42 on page 313, Router A learns through one of its interfaces that routes to Routers C, D, and E are unreachable. Router A readvertises those routes out the same interface as unreachable. The advertisement informs Router B that Hosts C, D, and E are definitely not reachable through Router A.

Limitations of Unidirectional Connectivity

Because RIP processes routing information based solely on the receipt of routing table updates, it cannot ensure bidirectional connectivity. As Figure 43 on page 313 shows, RIP networks are limited by their unidirectional connectivity.

Figure 43: Limitations of Unidirectional Connectivity



In Figure 43 on page 313, Routers A and D flood their routing table information to Router B. Because the path to Router E has the fewest hops when routed through Router A, that route is imported into Router B's forwarding table. However, suppose that Router A can transmit traffic but is not receiving traffic from Router B due to an unavailable link or invalid routing policy. If the only route to Router E is through Router A, any traffic destined for Router A is lost, because bidirectional connectivity was never established.

OSPF establishes bidirectional connectivity with a three-way handshake. For more information, see “Link-State Advertisements” on page 316.

RIPng Overview

The Routing Information Protocol next generation (RIPng) is an interior gateway protocol (IGP) that uses a distance-vector algorithm to determine the best route to a destination, using hop count as the metric. RIPng is a routing protocol that exchanges routing information used to compute routes and is intended for Internet Protocol version 6 (IPv6)-based networks.

On devices in secure context, IPv6 is disabled. You must enable IPv6 to use RIPng. For instructions, see "Enabling IPv6 in Secure Context" on page 65.

RIPng is disabled by default. For configuration instructions, see the *JUNOS Routing Protocols Configuration Guide*.

This overview contains the following topics:

RIPng Protocol Overview

The RIPng IGP uses the Bellman-Ford distance-vector algorithm to determine the best route to a destination, using hop count as the metric. RIPng allows hosts and routers to exchange information for computing routes through an IP-based network. RIPng is intended to act as an IGP for moderately- sized autonomous systems.

RIPng is a distinct routing protocol from RIPv2. The JUNOS software implementation of RIPng is similar to RIPv2, but has the following differences:

- RIPng does not need to implement authentication on packets.
- The JUNOS software does not support multiple instances of RIPng.
- The JUNOS software does not support RIPng routing table groups.

RIPng is a UDP-based protocol and uses UDP port 521.

RIPng has the following architectural limitations:

- The longest network path cannot exceed 15 hops (assuming that each network, or hop, has a cost of 1).
- RIPng is prone to routing loops when the routing tables are reconstructed. Especially when RIPng is implemented in large networks that consist of several hundred routers, RIPng might take extremely long time to resolve routing loops.
- RIPng uses only a fixed metric to select a route. Other IGPs use additional parameters, such as measured delay, reliability, and load.

RIPng Standards

RIPng is defined in the following documents:

- RFC 2080, *RIPng for IPv6*
- RFC 2081, *RIPng Protocol Applicability Statement*

To access Internet Requests for Comments (RFCs) and drafts, go to the Internet Engineering Task Force (IETF) Web site at http://www.ietf.org.

RIPng Packets

A RIPng packet header contains the following fields:

- Command—Indicates whether the packet is a request or response message. Request messages seek information for the router's routing table. Response messages are sent periodically or when a request message is received. Periodic response messages are called update messages. Update messages contain the command and version fields and a set of destinations and metrics.
- Version number—Specifies the version of RIPng that the originating router is running. This is currently set to Version 1.

The rest of the RIPng packet contains a list of routing table entries consisting of the following fields:

- Destination prefix—128-bit IPv6 address prefix for the destination.
- Prefix length—Number of significant bits in the prefix.
- Metric—Value of the metric advertised for the address.
- Route tag—A route attribute that must be advertised and redistributed with the route. Primarily, the route tag distinguishes external RIPng routes from internal RIPng routes in cases where routes must be redistributed across an exterior gateway protocol (EGP).

To configure RIPng, see the *JUNOS Routing Protocols Configuration Guide*.

OSPF Overview

In an Open Shortest Path First (OSPF) network, the network topology is distributed among the nodes of the autonomous system (AS) and is regularly updated through the exchange of link-state advertisements (LSAs). As a result, OSPF is known as a link-state protocol. Because topology changes are flooded throughout the network, every node maintains the same copy of the network map in its local topological database. Packets are then routed based on the shared topology using the shortest path first (SPF) algorithm.

This overview contains the following topics:

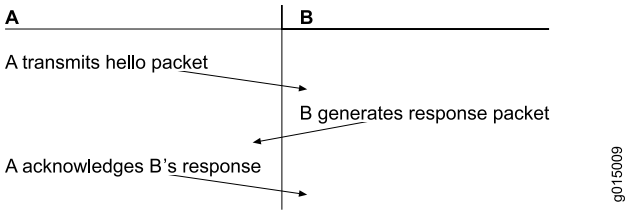
- Role of the Backbone Area on page 318
- Stub Areas and Not-So-Stubby Areas on page 319

NOTE: In this guide, the term *OSPF* refers to OSPF version 2 and OSPF version 3.

Link-State Advertisements

OSPF creates a topology map by flooding link-state advertisements (LSAs) across OSPF-enabled links. LSAs announce the presence of OSPF-enabled interfaces to adjacent OSPF interfaces. The exchange of LSAs establishes bidirectional connectivity between all adjacent OSPF interfaces (neighbors) using a three-way handshake, as shown in Figure 44 on page 316.

Figure 44: OSPF Three-Way Handshake



In Figure 44 on page 316, Router A sends hello packets out all its OSPF-enabled interfaces when it comes online. Router B receives the packet, which establishes that Router B can receive traffic from Router A. Router B generates a response to Router A to acknowledge receipt of the hello packet. When Router A receives the response, it establishes that Router B can receive traffic from Router A. Router A then generates a final response packet to inform Router B that Router A can receive traffic from Router B. This three-way handshake ensures bidirectional connectivity.

As new neighbors are added to the network or existing neighbors lose connectivity, the adjacencies in the topology map are modified accordingly through the exchange (or absence) of LSAs. These LSAs advertise only the incremental changes in the network, which helps minimize the amount of OSPF traffic on the network. The adjacencies are shared and used to create the network topology in the topological database.

Role of the Designated Router

Large local area networks (LANs) that have many routers and therefore many OSPF adjacencies can produce heavy control-packet traffic as LSAs are flooded across the network. To alleviate the potential traffic problem, OSPF uses designated routers (DRs). Rather than broadcasting LSAs to all their OSPF neighbors, the routers send their LSAs to the designated router, which processes the LSAs, generates responses, and multicasts topology updates to all OSPF routers.

In LANs, the election of the designated router takes place when the OSPF network is initially established. When the first OSPF links are active, the router with the highest router identifier (defined by the router-id configuration value or the loopback address)

is elected designated router. The router with the second highest router identifier is elected the backup designated router (BDR). If the designated router fails or loses connectivity, the BDR assumes its role and a new BDR election takes place between all the routers in the OSPF network.

Path Cost Metrics

Once the topology is shared across the network, OSPF uses it to route packets between network nodes. Each path between neighbors is assigned a cost based on the throughput, round-trip time, and reliability of the link. The sum of the costs across a particular path between hosts determines the overall cost of the path. Packets are then routed along the shortest path using the shortest path first (SPF) algorithm. If multiple equal-cost paths exist between a source and destination address, OSPF routes packets along each path alternately, in round-robin fashion.

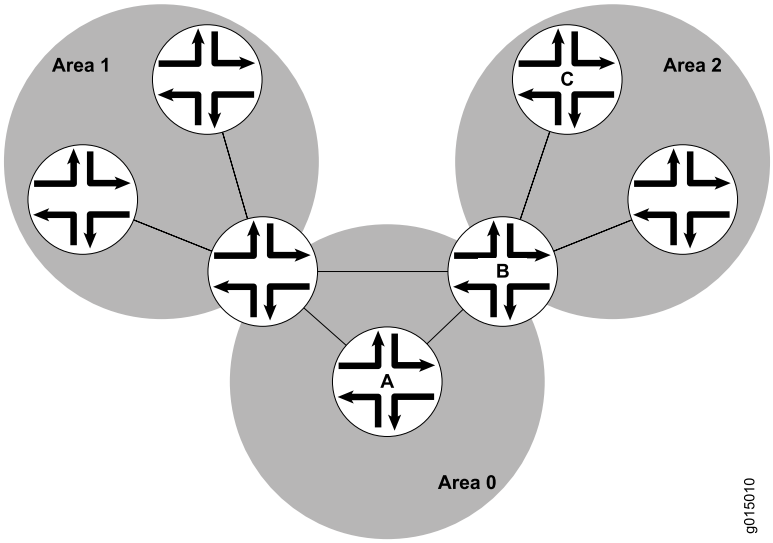
OSPF allows you to manually assign a cost (or metric) to a particular path segment to control the flow of packets across the network.

Areas and Area Border Routers

The OSPF networks in an AS are administratively grouped into areas. Each area within an AS operates like an independent network and has a unique 32-bit area ID, which functions like a network address. Within an area, the topology database contains only information about the area, LSAs are flooded only to nodes within the area, and routes are computed only within the area. Subnetworks are divided into other areas, which are connected to form the whole of the main network.

The central area of an AS, called the backbone area, has a special function and is always assigned the area ID 0.0.0.0. Area IDs are unique numeric identifiers, in dotted decimal notation, but they are not IP addresses. Area IDs need only be unique within an AS. All other networks or areas in the AS must be directly connected to the backbone area by a router that has interfaces in more than one area. These connecting routers are called area border routers (ABRs). Figure 45 on page 318 shows an OSPF topology of three areas connected by two area border routers.

Figure 45: Multiarea OSPF Topology



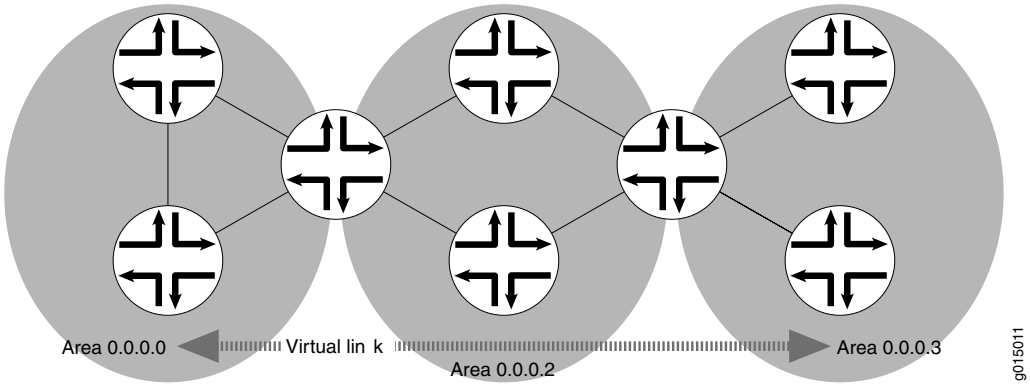
Area border routers are responsible for sharing topology information between areas. They summarize the link-state records of each area and advertise destination address summaries to neighboring areas. The advertisements contain the ID of the area in which each destination lies, so that packets are routed to the appropriate area border router. For example, in the OSPF areas shown in Figure 45 on page 318, packets sent from Router A to Router C are automatically routed through Area Border Router B.

Role of the Backbone Area

An OSPF restriction requires all areas to be directly connected to the backbone area so that packets can be properly routed. All packets are routed first to the backbone area by default. Packets that are destined for an area other than the backbone area are then routed to the appropriate area border router and on to the remote host within the destination area.

In large networks with many areas, in which direct connectivity between all areas and the backbone area is physically difficult or impossible, you can configure virtual links to connect noncontiguous areas. For example, Figure 46 on page 319 shows a virtual link between a noncontiguous area and the backbone area through an area connected to both.

Figure 46: OSPF Topology with a Virtual Link

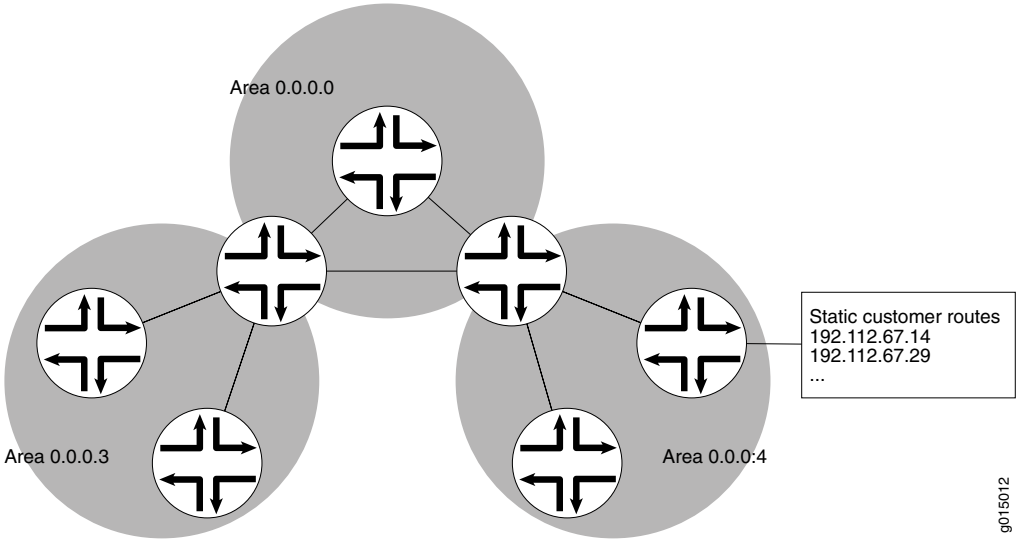


In the topology shown in Figure 46 on page 319, a virtual link is established between area 0.0.0.3 and the backbone area through area 0.0.0.2. All outbound traffic destined for other areas is routed through area 0.0.0.2 to the backbone area and then to the appropriate area border router. All inbound traffic destined for area 0.0.0.3 is routed to the backbone area and then through area 0.0.0.2.

Stub Areas and Not-So-Stubby Areas

Figure 47 on page 319 shows an AS across which many external routes are advertised. If external routes make up a significant portion of a topology database, you can suppress the advertisements in areas that do not have links outside the network. By doing so, you can reduce the amount of memory the nodes use to maintain the topology database and free it for other uses.

Figure 47: OSPF AS Network with Stub Areas and NSSAs



To control the advertisement of external routes into an area, OSPF uses stub areas. By designating an area border router interface to the area as a stub interface, you

suppress external route advertisements through the area border router. Instead, the area border router automatically advertises a default route (through itself) in place of the external routes. Packets destined for external routes are automatically sent to the area border router, which acts as a gateway for outbound traffic and routes them appropriately.

For example, area 0.0.0.3 in Figure 47 on page 319 is not directly connected to the outside network. All outbound traffic is routed through the area border router to the backbone and then to the destination addresses. By designating area 0.0.0.3 a stub area, you reduce the size of the topology database for that area by limiting the route entries to only those routes internal to the area.

Like area 0.0.0.3 in Figure 47 on page 319, area 0.0.0.4 has no external connections. However, area 0.0.0.4 has static customer routes that are not internal OSPF routes. You can limit the external route advertisements to the area and advertise the static customer routes by designating it a not-so-stubby area (NSSA). External routes are flooded into the NSSA and then leaked to the other areas, but external routes from other areas are not advertised within the NSSA.

IS-IS Overview

The Intermediate System-to-Intermediate System (IS-IS) protocol is a classless interior routing protocol developed by the International Organization for Standardization (ISO) as part of the development of the Open Systems Interconnection (OSI) protocol suite. Like OSPF routing, IS-IS uses hello packets that allow network convergence to occur quickly when network changes are detected.

This overview contains the following topics:

IS-IS Areas

An IS-IS network is a single autonomous system (AS), also called a routing domain, that consists of end systems and intermediate systems. End systems are network entities that send and receive packets. Intermediate systems (routers) send, receive, and relay (forward) packets.

IS-IS does not force the network to use a hierarchical physical topology. Instead, a single AS can be divided into two types of areas: Level 1 areas and Level 2 areas. A Level 1 area is similar to an OSPF stub area, and a Level 2 area interconnects all Level 1 areas. The router and its interfaces reside within one area, and Level 2 routers share link-state information. No IS-IS area functions strictly as a backbone.

Level 1 routers share intra-area routing information, and Level 2 routers share interarea information about IP addresses available within each area. Uniquely, IS-IS routers can act as both Level 1 and Level 2 routers, sharing intra-area routes with other Level 1 routers and interarea routes with other Level 2 routers.

The propagation of link-state updates is determined by the level boundaries. All routers within a level maintain a complete link-state database of all other routers in the same level. Each router then uses the Dijkstra algorithm to determine the shortest path from the local router to other routers in the link-state database.

Network Entity Titles and System Identifiers

In IS-IS, special network addresses are called network entity titles (NETs) and take several forms, depending on your network requirements. NET addresses are hexadecimal and range from 8 octets to 20 octets in length. Generally, the format consists of an authority and format Identifier (AFI), a domain ID, an area ID, a system identifier, and a selector. The simplest format omits the domain ID and is 10 octets long. For example, the NET address 49.0001.1921.6800.1001.00 consists of the following parts:

- 49—AFI
- 0001—Area ID
- 1921.6800.1001—System identifier
- 00—Selector

The system identifier must be unique within the network. For an IP-only network, we recommend using the IP address of an interface on the router. Configuring a loopback NET address with the IP address is helpful when troubleshooting is required on the network.

IS-IS Path Selection

Level 1 routers store information about all the subnets within an area, and choose intranetwork paths over internetwork paths. Using the area ID portion of the NET address, Level 1 routers determine which neighboring routers are Level 1 routers within the same area.

If the destination address is not within the area, Level 1 routers forward the packet to the nearest router configured as both a Level 1 and Level 2 router within the area. The Level 1 and Level 2 router forwards the packet, using the Level 2 topology, to the proper area. The destination router, which is configured as a Level 1 and Level 2 router, then determines the best path through the destination area.

Protocol Data Units

IS-IS routers use protocol data units (PDUs) to exchange information. Each protocol data unit (PDU) shares a common header.

IS-IS Hello PDU

IS-IS hello PDUs establish adjacencies with other routers and have three different formats: one for point-to-point hello packets, one for Level 1 broadcast links, and one for Level 2 broadcast links. Level 1 routers must share the same area address to form an adjacency, while Level 2 routers do not have this limitation. The request for adjacency is encoded in the Circuit type field of the PDU.

Hello PDUs have a preset length assigned to them. The IS-IS router does not resize any PDU to match the maximum transmission unit (MTU) on a router interface. Each interface supports the maximum IS-IS PDU of 1492 bytes, and hello PDUs are padded to meet the maximum value. When the hello is sent to a neighboring router, the connecting interface supports the maximum PDU size.

Link-State PDU

A link-state PDU (LSP) contains information about each router in the network and the connected interfaces. Also included is metric and IS-IS neighbor information. Each LSP must be refreshed periodically on the network and is acknowledged by information within a sequence number packet.

On point-to-point links, each LSP is acknowledged by a partial sequence number PDU (PSNP), but on broadcast links, a complete sequence number PDU (CSNP) is sent out over the network. Any router that finds newer LSP information in the CSNP then purges the out-of-date entry and updates the link-state database.

LSPs support variable-length subnet mask addressing.

Complete Sequence Number PDU

The complete sequence number PDU (CSNP) lists all the link-state PDUs (LSPs) in the link-state database of the local router. Contained within the CSNP is an LSP identifier, a lifetime, a sequence number, and a checksum for each entry in the database. Periodically, a CSNP is sent on both broadcast and point-to-point links to maintain a correct database. Also, the advertisement of CSNPs occurs when an adjacency is formed with another router. Like IS-IS hello PDUs, CSNPs come in two types: Level 1 and Level 2.

When a device receives a CSNP, it checks the database entries again its own local link-state database. If it detects missing information, the device requests specific LSP details using a partial sequence number PDU (PSNP).

Partial Sequence Number PDU

A partial sequence number PDU (PSNP) is used by an IS-IS router to request LSP information from a neighboring router. A PSNP can also explicitly acknowledge the receipt of an LSP on a point-to-point link. On a broadcast link, a CSNP is used as implicit knowledge. Like hello PDUs and CSNPs, the PSNP also has two types: Level 1 and Level 2.

When a device compares a CSNP to its local database and determines that an LSP is missing, the router issues a PSNP for the missing LSP, which is returned in a link-state PDU from the router sending the CSNP. The received LSP is then stored in the local database, and an acknowledgement is sent back to the originating router.

BGP Overview

The Border Gateway Protocol (BGP) is an exterior gateway protocol (EGP) used primarily to establish point-to-point connections and transmit data between peer ASs. Unlike the IGPs RIP, OSPF and IS-IS, BGP must explicitly advertise the routes

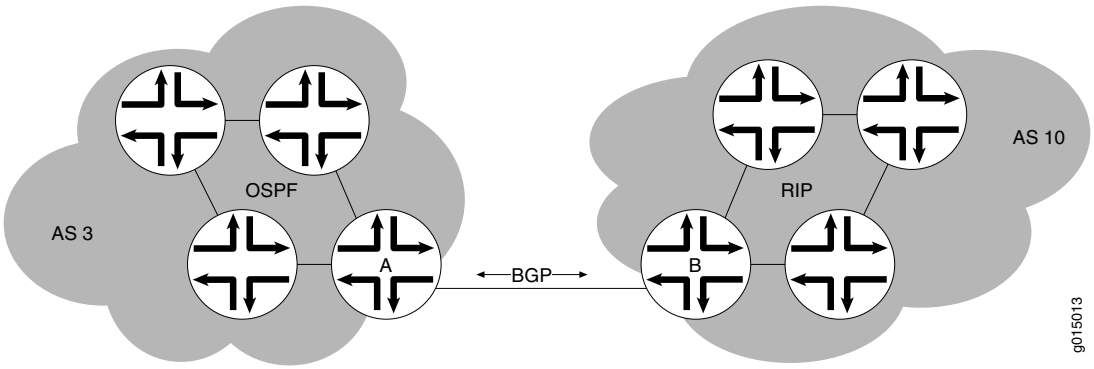
between its peers. The route advertisements determine prefix reachability and the way packets are routed between BGP neighbors. Because BGP uses the packet path to determine route selection, it is considered a path-vector protocol.

This overview contains the following topics:

Point-to-Point Connections

To establish point-to-point connections between peer ASs, you configure a BGP session on each interface of a point-to-point link. Figure 48 on page 323 shows an example of a BGP peering session.

Figure 48: BGP Peering Session



In Figure 48 on page 323, Router A is a gateway router for AS 3, and Router B is a gateway router for AS 10. For traffic internal to either AS, an IGP (OSPF, for instance) is used. To route traffic between peer ASs, a BGP session is used.

BGP Messages for Session Establishment

When the routers on either end of a BGP session first boot, the session between them is in the Idle state. The BGP session remains idle until a start event is detected. Typically, the start event is the configuration of a new BGP session or the resetting of an existing BGP session. At boot time, the start event is generated by the router as the BGP session is initiated.

After it detects a start event, the BGP host sends TCP request packets to its configured BGP neighbors. These packets are directed only to neighboring interfaces that have been explicitly configured as BGP neighbors. Upon receipt of the TCP request packet, the neighboring host generates a TCP response to complete the three-way handshake and establish a TCP connection between the peers. While this handshake is taking place, the BGP state for the connection is Connect. If a TCP timeout occurs while the originating host is waiting for a TCP response packet, the BGP state for the connection is Active. The Active state indicates that the router is actively listening for a TCP response and the TCP retry timer has been initiated.

Once a TCP connection has been established between both ends of a BGP session, the BGP session state is OpenSent, indicating that the originating router has generated an open message. The open message is an initial BGP handshake that must occur before any route advertisement can take place. Upon receipt of the open message, the neighboring router generates a keepalive message. Receipt of the keepalive message establishes a point-to-point connection, and the BGP session state transitions to Established. While the originating host waits for the keepalive response packet, the BGP session state is OpenConfirm.

BGP Messages for Session Maintenance

Once a BGP session has been established, the BGP peers exchange route advertisements by means of update messages. Update messages contain a one or more route advertisements, and they can contain one or more prefixes that are to be removed from the BGP routing table. If the peers need to advertise multiple routes, they generate and send multiple update messages as they detect changes to the network. In the absence of changes to the routing table, no update messages are generated.

While a BGP session is active, each router on the BGP session generates keepalive messages periodically. The timing of these messages is determined by the hold time on the session. The hold time is a negotiated value specifying the number of seconds that can elapse without keepalive messages before BGP designates the link inactive. Three messages are sent during every hold time interval.

When a peer connection is closed (either by error or if the BGP session is closed), a notification message is generated and sent to the peer router that did not experience the error or did not terminate the BGP session.

IBGP and EBGP

BGP uses two primary modes of information exchange, internal BGP (IBGP) and external BGP (EBGP), to communicate with internal and external peers, respectively.

Peer ASs establish links through an external peer BGP session. As a result, all route advertisement between the external peers takes place by means of the EBGP mode of information exchange. To propagate the routes through the AS and advertise them to internal peers, BGP uses IBGP. To advertise the routes to a different peer AS, BGP again uses EBGP.

To avoid routing loops, IBGP does not advertise routes learned from an internal BGP peer to other internal BGP peers. For this reason, BGP cannot propagate routes

throughout an AS by passing them from one router to another. Instead, BGP requires that all internal peers be fully meshed so that any route advertised by one router is advertised to all peers within the AS.

As a network grows, the full mesh requirement becomes increasingly difficult to manage. In a network with 1000 routers, the addition of a single router requires that all the routers in the network be modified to account for the new addition. To combat these scaling problems, BGP uses route reflection and BGP confederations.

For information about route reflection, see "Scaling BGP for Large Networks" on page 330. For information about routing confederations, see "Scaling BGP for Large Networks" on page 330.

Route Selection

The BGP route selection process compares BGP attributes to select a single best path or active route for each prefix in the routing table. The attributes are compared in a particular order. A local BGP router uses the following criteria, in the order presented, to select a route from the routing table for the forwarding table:

1. Next-hop accessibility—If the next hop is inaccessible, the local router does not consider the route. The router must verify that it has a route to the BGP next-hop address. If a local route to the next hop does not exist, the local route does not include the router in its forwarding table. If such a route exists, route selection continues.
2. Highest local preference—The local router selects the route with the highest local preference value. If multiple routes have the same preference, route selection continues. (For more information, see "Local Preference" on page 326.)
3. Shortest AS path—The local router selects the route with the fewest entries in the AS path. If multiple routes have the same AS path length, route selection continues. (For more information, see "AS Path" on page 327.)
4. Lowest origin—The local router selects the route with the lowest origin value. If multiple routes have the same origin value, route selection continues. (For more information, see "Origin" on page 327.)
5. Lowest MED value—The local router selects the route with the lowest multiple exit discriminator (MED) value, comparing the routes from the same AS only. If multiple routes have the same MED value, route selection continues. For more information, see "Multiple Exit Discriminator" on page 328.
6. Strictly external paths—The local router prefers strictly external (EBGP) paths over external paths learned through interior sessions (IBGP). If multiple routes have the same strictly external paths, route selection continues.
7. Lowest IGP route metric— The local router selects the path for which the next hop is resolved through the IGP route with the lowest metric. If multiple routes have the same IGP route metric, route selection continues.
8. Maximum IGP next hops—The local router selects the path for which the BGP next hop is resolved through the IGP route with the largest number of next hops. If multiple routes have the same number of next hops, route selection continues.
9. Shortest route reflection cluster list—The local router selects the path with the shortest route reflection cluster list. Routes without a cluster list are considered

to have a cluster list of length 0. If multiple routes have the same route reflection cluster list, route selection continues.

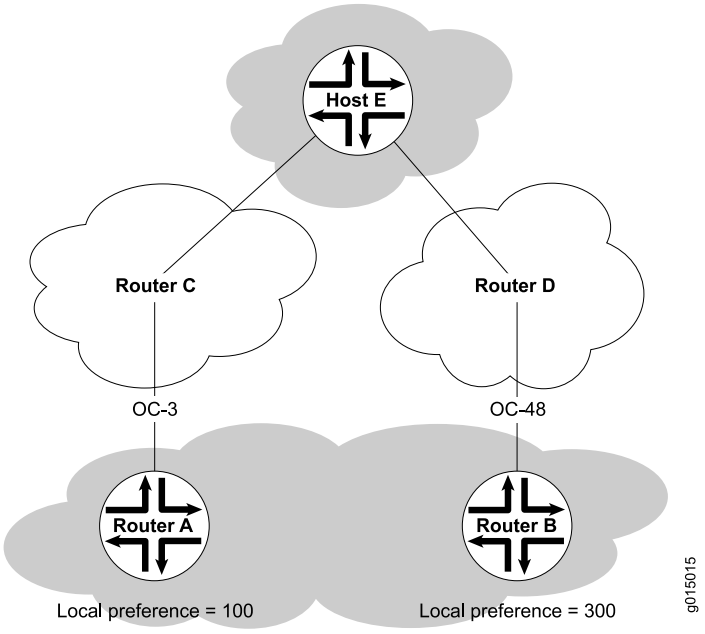
10. Lowest router ID—The local router selects the route with the lowest IP address value for the BGP router ID. By default, the router IDs of routes received from different ASs are not compared. You can change this default behavior. For more information, see the *JUNOS Routing Protocols Configuration Guide*.
11. Lowest peer IP address—The local router selects the path that was learned from the neighbor with the lowest peer IP address.

You can change the default behavior of some attributes (such as MED and router ID) used in the route selection process. For more information, see the *JUNOS Routing Protocols Configuration Guide*.

Local Preference

The local preference is typically used to direct all outbound AS traffic to a certain peer. When you configure a local preference, all routes that are advertised through that peer are assigned the preference value. The preference is a numeric value, and higher values are preferred during BGP route selection. Figure 49 on page 326 illustrates how to use local preference to determine BGP route selection.

Figure 49: Local Preference



The network in Figure 49 on page 326 shows two possible routes to the prefixes accessible through Host E. The first route, through Router A, uses an OC3 link to Router C and is then forwarded to Host E. The second route, through Router B, uses an OC48 link to Router D and is then forwarded to Host E. Although the number of hops to Host E is identical regardless of the route selected, the route through Router B is more desirable because of the increased bandwidth. To force traffic through

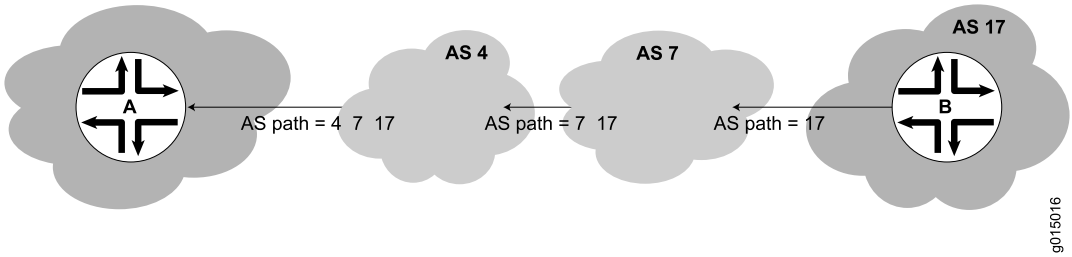
Router B, you can set the local preference on Router A to 100 and the local preference on Router B to 300. During BGP route selection, the route with the higher local preference is selected.

NOTE: In contrast to almost every other metric associated with dynamic routing protocols, the local preference gives higher precedence to the larger value.

AS Path

Routes advertised by BGP maintain a list of the ASs through which the route travels. This information is stored in the route advertisement as the AS path, and it is one of the primary criteria that a local router uses to evaluate BGP routes for inclusion in its forwarding table. Figure 50 on page 327 shows how BGP creates an AS path.

Figure 50: BGP AS Path



In the network shown in Figure 50 on page 327, the route from Host A to Host B travels through two intermediate ASs. As the route advertisement is propagated through the BGP network, it accumulates an AS path number each time it exits one AS and enters another. Each AS number is prepended to the AS path, which is stored as part of the route advertisement. When the route advertisement first leaves Host B's AS, the AS path is 17. When the route is advertised between intermediate ASs, the AS number 7 is prepended to the AS path, which becomes 7 17. When the route advertisement exits the third AS, the AS path becomes 4 7 17. The route with the shortest AS path is preferred for inclusion into the BGP forwarding table.

Origin

The BGP router that first advertises a route assigns it of the following values to identify its origin. During route selection, the lowest origin value is preferred.

- 0—The router originally learned the route through an IGP (OSPF, IS-IS, or a static route).
- 1—The router originally learned the route through an EGP (most likely BGP).
- 2—The route's origin is unknown.

Multiple Exit Discriminator

A multiple exit discriminator (MED) is an arbitrary metric assigned to a route to determine the exit point to a destination when all other factors are equal. By default, MED metrics are compared only for routes to the same peer AS, but you can also configure routing table path selection options for different ways of comparing MEDs.

Default MED Usage

Because the AS path rather than the number of hops between hosts is the primary criterion for BGP route selection, an AS with multiple connections to a peer AS can have multiple equivalent AS paths. When the routing table contains two routes to the same host in a neighboring AS, a multiple exit discriminator (MED) metric assigned to each route can determine which to include in the forwarding table. The MED metric you assign can force traffic through a particular exit point in an AS.

Figure 51 on page 328 illustrates how MED metrics are used to determine route selection.

Figure 51: Default MED Example

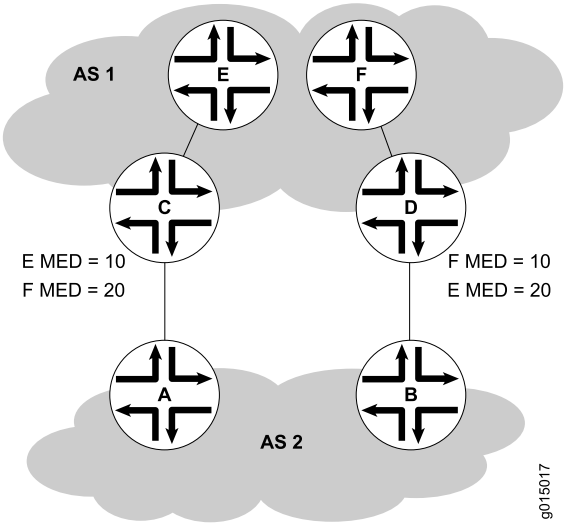


Figure 51 on page 328 shows AS 1 and AS 2 connected by two separate BGP links to Routers C and D. Host E in AS 1 is located nearer Router C. Host F, also in AS 1, is located nearer Router D. Because the AS paths are equivalent, two routes exist for each host, one through Router C and one through Router D. To force all traffic destined for Host E through Router C, network administrator for AS 2 assigns an MED metric for each router to Host E at its exit point. An MED metric of 10 is assigned to the route to Host E through Router C, and an MED metric of 20 is assigned to the route to Host E through Router D. BGP routers in AS 2 then select the route with the lower MED metric for the forwarding table.

Additional MED Options for Path Selection

By default, only the MEDs of routes that have the same peer ASs are compared. However, you can configure the routing table path selection options listed in Table 99 on page 329 to compare MEDs in different ways. The MED options are not mutually exclusive and can be configured in combination or independently. For the MED options to take effect, you must configure them uniformly all through your network. The MED option or options you configure determine the route selected. Thus we recommend that you carefully evaluate your network for preferred routes before configuring the MED options. For information about configuring the MED options, see the *JUNOS Routing Protocols Configuration Guide*.

Table 99: MED Options for Routing Table Path Selection

Option (Name)	Function	Use
Always comparing MEDs (always-compare-med)	Ensures that the MEDs for paths from peers in different ASs are always compared in the route selection process	Useful when all enterprises participating in a network agree on a uniform policy for setting MEDs. For example, in a network shared by two ISPs, both must agree that a certain path is the better path to configure the MED values correctly.
Adding IGP cost to MED (med-plus-igp)	Before comparing MED values for path selection, adds to the MED the cost of the IGP route to the BGP next-hop destination. This option replaces the MED value for the router, but does not affect the IGP metric comparison. As a result, when multiple routes have the same value after the MED-plus-IPG comparison, and route selection continues, the IGP route metric is also compared, even though it was added to the MED value and compared earlier in the selection process.	Useful when the downstream AS requires the complete cost of a certain route that is received across multiple ASs.
Applying Cisco IOS nondeterministic behavior (cisco-non-deterministic)	Specifies the nondeterministic behavior of the Cisco IOS software: ■ The active path is always first. All nonactive but eligible paths follow the active path and are maintained in the order in which they were received. Ineligible paths remain at the end of the list. ■ When a new path is added to the routing table, path comparisons are made among all routes, including those paths that must never be selected because they lose the MED tie-breaking rule.	We recommend that you do not configure this option, because the nondeterministic behavior sometimes prevents the system from properly comparing the MEDs between paths.

Scaling BGP for Large Networks

BGP is not a flooding protocol like RIP or OSPF. Instead, it is a peering protocol that exchanges routes with fully meshed peers only. However, in large networks, the full mesh requirement causes scaling problems. BGP combats scaling problems with the following methods:

- Route Reflectors—for Added Hierarchy on page 330
- Confederations—for Subdivision on page 332

Route Reflectors—for Added Hierarchy

To use route reflection in an AS, you designate one or more routers as a route reflector—typically, one per point of presence (POP). Route reflectors have the special BGP ability to readvertise routes learned from an internal peer to other internal peers. So rather than requiring all internal peers to be fully meshed with each other, route reflection requires only that the route reflector be fully meshed with all internal peers. The route reflector and all its internal peers form a cluster, as shown in Figure 52 on page 330.

NOTE: You must have an Advanced BGP Feature license installed on each device that uses a route reflector. For license details, see the *JUNOS Software Administration Guide*.

Figure 52: Simple Route Reflector Topology (One Cluster)

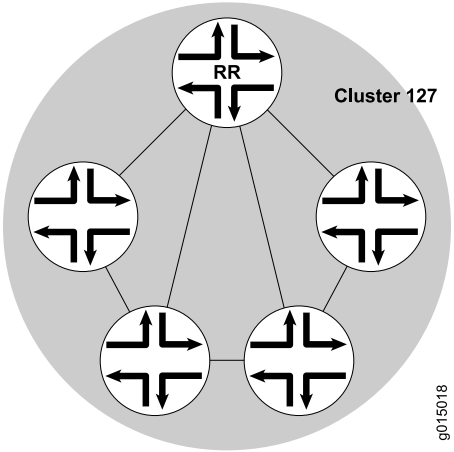


Figure 52 on page 330 shows Router RR configured as the route reflector for Cluster 127. The other routers are designated internal peers within the cluster. BGP routes are advertised to Router RR by any of the internal peers. RR then readvertises those routes to all other peers within the cluster.

You can configure multiple clusters and link them by configuring a full mesh of route reflectors (see Figure 53 on page 331).

Figure 53: Basic Route Reflection (Multiple Clusters)

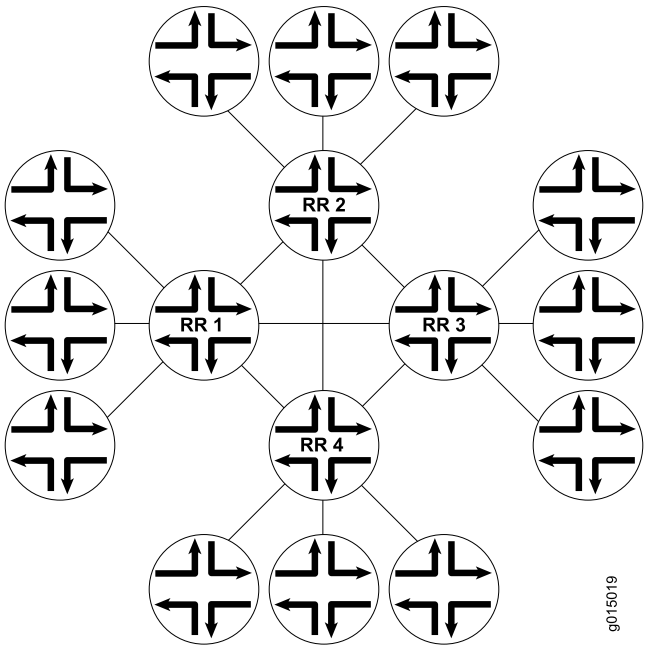


Figure 53 on page 331 shows Route Reflectors RR1, RR2, RR3, and RR4 as fully meshed internal peers. When a router advertises a route to Reflector RR1, RR1 readvertises the route to the other route reflectors, which, in turn, readvertise the route to the remaining routers within the AS. Route reflection allows the route to be propagated throughout the AS without the scaling problems created by the full mesh requirement.

However, as clusters become large, a full mesh with a route reflector becomes difficult to scale, as does a full mesh between route reflectors. To help offset this problem, you can group clusters of routers together into clusters of clusters for hierarchical route reflection (see Figure 54 on page 331).

Figure 54: Hierarchical Route Reflection (Clusters of Clusters)

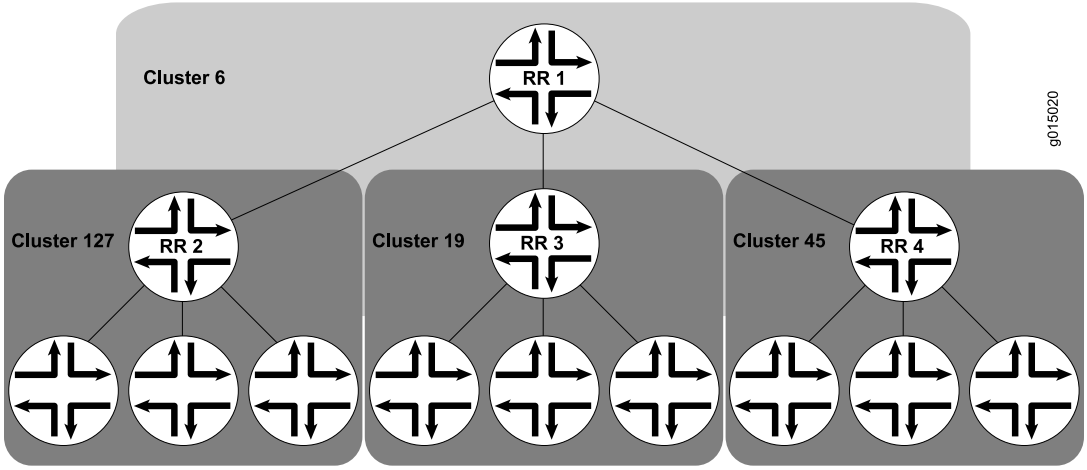


Figure 54 on page 331 shows RR2, RR3, and RR4 as the route reflectors for Clusters 127, 19, and 45, respectively. Rather than fully mesh those route reflectors, the network administrator has configured them as part of another cluster (Cluster 6) for which RR1 is the route reflector. When a router advertises a route to RR2, RR2 readvertises the route to all the routers within its own cluster, and then readvertises the route to RR1. RR1 readvertises the route to the routers in its cluster, and those routers propagate the route down through their clusters.

Confederations—for Subdivision

BGP confederations are another way to solve the scaling problems created by the BGP full mesh requirement. BGP confederations effectively break up a large AS into subautonomous systems (sub-ASs). Each sub-AS must be uniquely identified within the confederation AS by a sub-AS number. Typically, sub-AS numbers are taken from the private AS numbers between 64512 and 65535.

Within a sub-AS, the same IBGP full mesh requirement exists. Connections to other confederations are made with standard EBGP, and peers outside the sub-AS are treated as external. To avoid routing loops, a sub-AS uses a confederation sequence, which operates like an AS path but uses only the privately assigned sub-AS numbers.

The confederation AS appears whole to other confederation ASs. The AS path received by other ASs shows only the globally assigned AS number. It does not include the confederation sequence or the privately assigned sub-AS numbers. The sub-AS numbers are removed when the route is advertised out of the confederation AS. Figure 55 on page 332 shows an AS divided into four confederations.

Figure 55: BGP Confederations

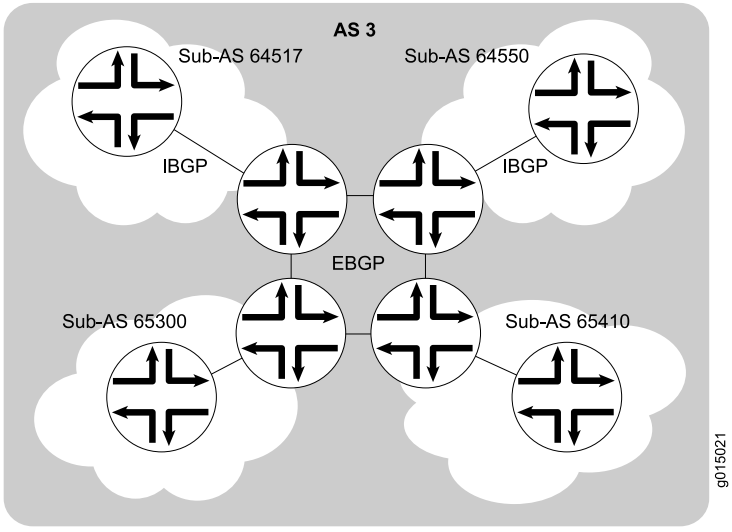


Figure 55 on page 332 shows AS 3 divided into four sub-ASs, 64517, 64550, 65300, and 65410, which are linked through EBGP sessions. Because the confederations are connected by EBGP, they do not need to be fully meshed. EBGP routes are readvertised to other sub-ASs.

Chapter 13

Configuring Static Routes

Static routes are routes that you explicitly enter into the routing table as permanent additions. Traffic through static routes is always routed the same way.

NOTE: Before configuring routing protocols on a device running JUNOS software, you must first configure security filters. For more information, see the *JUNOS Software Security Configuration Guide*.

You can use either J-Web Quick Configuration or a configuration editor to configure static routes.

This chapter contains the following topics. For more information about static routes, see the *JUNOS Routing Protocols Configuration Guide*.

Static Routing Overview

Routes that are permanent fixtures in the routing and forwarding tables are often configured as static routes. These routes generally do not change, and often include only one or very few paths to the destination.

To create a static route in the routing table, you must, at minimum, define the route as static and associate a next-hop address with it. The static route in the routing table is inserted into the forwarding table when the next-hop address is reachable. All traffic destined for the static route is transmitted to the next-hop address for transit.

This overview contains the following topics:

Static Route Preferences

A static route destination address can have multiple next hops associated with it. In this case, multiple routes are inserted into the routing table, and route selection must occur. Because the primary criterion for route selection is the route preference, you can control the routes that are used as the primary route for a particular destination by setting the route preference associated with a particular next hop. The routes with a higher preference are always used to route traffic. When you do not set a preferred route, traffic is alternated between routes in round-robin fashion.

Qualified Next Hops

In general, the default properties assigned to a static route apply to all the next-hop addresses configured for the static route. If, however, you want to configure two possible next-hop addresses for a particular route and have them treated differently, you can define one as a qualified next hop.

Qualified next hops allow you to associate one or more properties with a particular next-hop address. You can set an overall preference for a particular static route and then specify a different preference for the qualified next hop. For example, suppose two next-hop addresses (10.10.10.10 and 10.10.10.7) are associated with the static route 192.168.47.5/32. A general preference is assigned to the entire static route, and then a different preference is assigned to only the qualified next-hop address 10.10.10.7. For example:

```
route 192.168.47.5/32 {
    next-hop 10.10.10.10;
    qualified-next-hop 10.10.10.7 {
        preference 2;
    }
    preference 6;
}
```

In this example, the qualified next hop 10.10.10.7 is assigned the preference 2, and the next-hop 10.10.10.10 is assigned the preference 6.

Control of Static Routes

You can control the importation of static routes into the routing and forwarding tables in a number of ways. Primary ways include assigning one or more of the following attributes to the route:

- retain—Keeps the route in the forwarding table after the routing process shuts down or the device reboots. For more information, see “Route Retention” on page 335.
- no-readvertise—Prevents the route from being readvertised to other routing protocols. For more information, see “Readvertisement Prevention” on page 335.
- passive—Rejects traffic destined for the route. For more information, see “Forced Rejection of Passive Route Traffic” on page 335.

Route Retention

By default, static routes are not retained in the forwarding table when the routing process shuts down. When the routing process starts up again, any routes configured as static routes must be added to the forwarding table again. To avoid this latency, routes can be flagged as retain, so that they are kept in the forwarding table even after the routing process shuts down. Retention ensures that the routes are always in the forwarding table, even immediately after a system reboot.

Readvertisement Prevention

Static routes are eligible for readvertisement by other routing protocols by default. In a stub area where you might not want to readvertise these static routes under any circumstances, you can flag the static routes as no-readvertise.

Forced Rejection of Passive Route Traffic

Generally, only active routes are included in the routing and forwarding tables. If a static route's next-hop address is unreachable, the route is marked passive, and it is not included in the routing or forwarding tables. To force a route to be included in the routing tables regardless of next-hop reachability, you can flag the route as passive. If a route is flagged passive and its next-hop address is unreachable, the route is included in the routing table and all traffic destined for the route is rejected.

Default Properties

The basic configuration of static routes defines properties for a particular route. To define a set of properties to be used as defaults on all static routes, set those properties as default values. For example:

```
defaults {
    retain;
    no-readvertise;
    passive;
}
route 0.0.0.0/0 next-hop 192.168.1.1;
route 192.168.47.5/32 {
    next-hop 10.10.10.10;
    qualified-next-hop 10.10.10.7 {
        preference 6;
    }
    preference 2;
}
```

In this example, the retain, no-readvertise, and passive attributes are set as defaults for all static routes. If any local setting for a particular route conflicts with the default values, the local setting supersedes the default.

Before You Begin

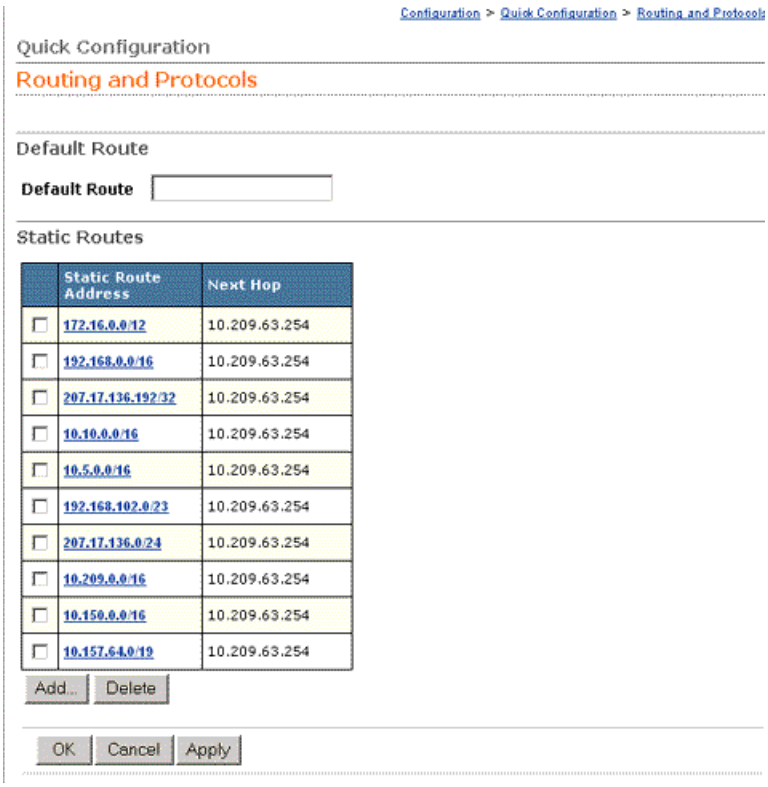
Before you begin configuring static routes, complete the following tasks:

- Establish basic connectivity. See the Getting Started Guide for your device.
- Configure network interfaces. See “Configuring Ethernet, DS1, DS3, and Serial Interfaces” on page 73.
- Configure security filters. See the *JUNOS Software Security Configuration Guide*.

Configuring Static Routes with Quick Configuration

J-Web Quick Configuration allows you to configure static routes. Figure 56 on page 336 shows the Quick Configuration Routing page for static routing.

Figure 56: Quick Configuration Routing Page for Static Routing



To configure static routes with Quick Configuration:

1. In the J-Web user interface, select **Configuration > Quick Configuration > Routing and Protocols**.
2. Enter information into the Static Routing Quick Configuration page, as described in Table 100 on page 337.
3. From the main static routing Quick Configuration page, click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration Routing page for static routing, click **Apply**.

- To apply the configuration and return to the Quick Configuration Routing page, click **OK**.
- To cancel your entries and return to the Quick Configuration Routing page, click **Cancel**.

4. To check the configuration, see "Verifying the Static Route Configuration" on page 343.

Table 100: Static Routing Quick Configuration Summary

Field	Function	Your Action
Default Route		
Default Route	Specifies the default gateway for the router.	Type the 32-bit IP address of the device's default route in dotted decimal notation.
Static Routes		
Static Route Address (required)	Specifies the static route to add to the routing table.	1. On the main static routing Quick Configuration page, click **Add**. 2. In the Static Route Address box, type the 32-bit IP address of the static route in dotted decimal notation.
Next-Hop Addresses	Specifies the next-hop address or addresses to be used when routing traffic to the static route.	1. In the Add box, type the 32-bit IP address of the next-hop host. 2. Click **Add**. 3. Add more next-hop addresses as necessary. **NOTE:** If a route has multiple next-hop addresses, traffic is routed across each address in round-robin fashion. 4. When you have finished adding next-hop addresses, click **OK**.

Configuring Static Routes with a Configuration Editor

To configure static routes on the device, you must perform the following tasks marked *(Required)*.

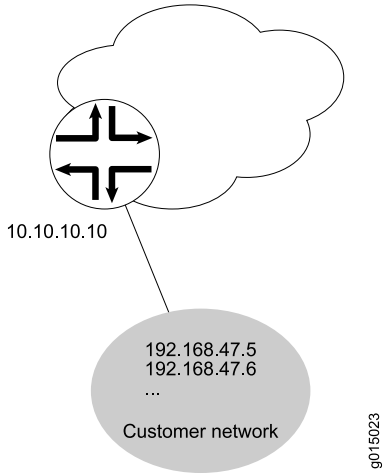
- Configuring a Basic Set of Static Routes (Required) on page 338
- Controlling Static Route Selection (Optional) on page 339
- Controlling Static Routes in the Routing and Forwarding Tables (Optional) on page 341
- Defining Default Behavior for All Static Routes (Optional) on page 342

For information about using the J-Web and CLI configuration editors, see the *J-Web Interface User Guide* and the *JUNOS CLI User Guide*.

Configuring a Basic Set of Static Routes (Required)

Customer routes that are connected to stub networks are often configured as static routes. Figure 57 on page 338 shows a sample network.

Figure 57: Customer Routes Connected to a Stub Network



To configure customer routes as static routes, like the ones in Figure 57 on page 338, follow these steps on the device to which the customer routes are connected:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 101 on page 339.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following procedures:
   - To manually control static route selection, see "Controlling Static Route Selection (Optional)" on page 339.
   - To determine how static routes are imported into the routing and forwarding tables, see "Controlling Static Routes in the Routing and Forwarding Tables (Optional)" on page 341.
   - To define default properties for static routes, see "Defining Default Behavior for All Static Routes (Optional)" on page 342.
   - To check the configuration, see "Verifying the Static Route Configuration" on page 343.

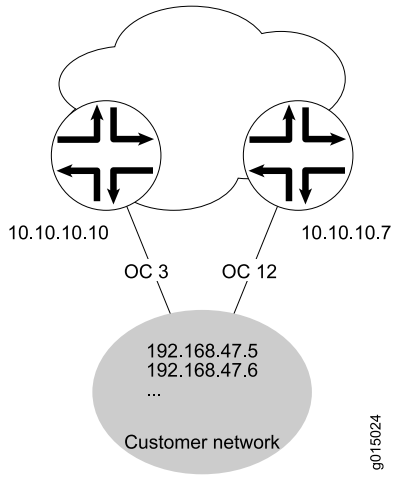
Table 101: Configuring Basic Static Routes

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Static** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing options, click **Configure** or **Edit**. 3. Next to Static, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit routing-options static
Add the static route 192.168.47.5/32, and define the next-hop address 10.10.10.10.	1. Next to Route, click **Add new entry**. 2. In the Destination box, type 192.168.47.5/32. 3. From the Next hop list, select **Next hop**. 4. Next to Next hop, click **Add new entry**. 5. In the Value box, type 10.10.10.10. 6. Click **OK**.	Define the static route and set the next-hop address: set route 192.168.47.5 next-hop 10.10.10.10

Controlling Static Route Selection (Optional)

When multiple next hops exist for a single static route (see Figure 58 on page 339), you can specify how traffic is to be routed to the destination.

Figure 58: Controlling Static Route Selection



In this example, the static route 192.168.47.5/32 has two possible next hops. Because of the links between those next-hop hosts, host 10.10.10.7 is the preferred path. To

configure the static route 192.168.47.5/32 with two next hops and give preference to host 10.10.10.7, follow these steps:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 102 on page 340.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following procedures:
   - To determine how static routes are imported into the routing and forwarding tables, see "Controlling Static Routes in the Routing and Forwarding Tables (Optional)" on page 341.
   - To define default properties for static routes, see "Defining Default Behavior for All Static Routes (Optional)" on page 342.
   - To check the configuration, see "Verifying the Static Route Configuration" on page 343.

Table 102: Controlling Static Route Selection

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Static** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing options, click **Configure** or **Edit**. 3. Next to Static, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit routing-options static
Add the static route 192.168.47.5/32, and define the next-hop address 10.10.10.10.	1. Next to Route, click **Add new entry**. 2. In the Destination box, type 192.168.47.5/32. 3. From the Next hop list, select **Next hop**. 4. In the Next hop box, click **Add new entry**. 5. In the Value box, type 10.10.10.10. 6. Click **OK**.	Define the static route and set the next-hop address: set route 192.168.47.5 next-hop 10.10.10.10
Set the preference for the 10.10.10.10 next hop to 7.	1. Next to Preference, select the **Yes** check box. 2. Click **Configure**. 3. In the Metric value box, type **7**. 4. Click **OK**.	Set the preference to 7: set route 192.168.47.5 next-hop 10.10.10.10 preference 7
Define the qualified next-hop address 10.10.10.7.	1. Next to Qualified next hop, click **Add new entry**. 2. In the Nexthop box, type 10.10.10.7.	Set the qualified-next-hop address: set route 192.168.47.5 qualified-next-hop 10.10.10.7

Table 102: Controlling Static Route Selection *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Set the preference for the 10.10.10.7 qualified next hop to 6.	1. In the Preference box, type 6. 2. Click **OK**.	Set the preference to 6: set route 192.168.47.5 qualified-next-hop 10.10.10.7 preference 6

Controlling Static Routes in the Routing and Forwarding Tables (Optional)

Static routes have a number of attributes that define how they are inserted and maintained in the routing and forwarding tables. To customize this behavior for the static route 192.168.47.5/32, perform these steps:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 103 on page 341.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following procedures:
   - To define default properties for static routes, see "Defining Default Behavior for All Static Routes (Optional)" on page 342.
   - To check the configuration, see "Verifying the Static Route Configuration" on page 343.

Table 103: Controlling Static Routes in the Routing and Forwarding Tables

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **192.168.47.5/32** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing options, click **Edit**. 3. Next to Static, click **Edit**. 4. Under Route and Destination, click **192.168.47.5/32**.	From the [edit] hierarchy level, enter edit routing-options static route 192.168.47.5/32
Specify that the route is to be retained in the forwarding table after the routing process shuts down. By default, static routes are not retained.	Next to Retain, select the **Yes** check box.	Set the retain attribute: set retain
Specify that the static route is not to be readvertised. By default, static routes are eligible to be readvertised.	Next to Readvertise, select the **No** check box.	Set the no-readvertise attribute: set no-readvertise

Table 103: Controlling Static Routes in the Routing and Forwarding Tables *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Specify that the static route is to be included in the routing table whether the route is active or not. By default, passive routes are not included in the routing table.	1. From the Passive flag list, select **Passive**. 2. Click **OK**.	Set the passive attribute: set passive

Defining Default Behavior for All Static Routes (Optional)

Attributes that define static route behavior can be configured either at the individual route level or as a default behavior that applies to all static routes. In the case of conflicting configuration, the configuration at the individual route level overrides static route defaults. To configure static route defaults, perform these steps:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 104 on page 342.
3. If you are finished configuring the router, commit the configuration.
4. To check the configuration, see "Verifying the Static Route Configuration" on page 343.

Table 104: Defining Static Route Defaults

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Defaults** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing options, click **Edit**. 3. Next to Static, click **Edit**. 4. Next to Defaults, click **Configure**.	From the [edit] hierarchy level, enter edit routing-options static defaults
Specify that the route is to be retained in the forwarding table after the routing process shuts down. By default, static routes are not retained.	1. Next to Retain, select the **Yes** check box. 2. Click **OK**.	Set the retain attribute: set retain
Specify that the static route is not to be readvertised. By default, static routes are eligible to be readvertised.	1. Next to Readvertise, select the **No** check box. 2. Click **OK**.	Set the no-readvertise attribute: set no-readvertise
Specify that the static route is to be included in the routing table whether the route is active or not. By default, passive routes are not included in the routing table.	1. From the Passive flag list, select **Passive**. 2. Click **OK**.	Set the passive attribute: set passive

Verifying the Static Route Configuration

Verify that the static routes are in the routing table and that those routes are active.

Displaying the Routing Table

Purpose Verify static route configuration as follows by displaying the routing table and checking its contents.

Action From the CLI, enter the show route terse command.

Sample Output

```
user@host> show route terse
inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 192.168.47.5/32        S   5                       Reject
* 172.16.0.0/12      S   5                      >192.168.71.254
* 192.168.0.0/18     S   5                      >192.168.71.254
* 192.168.40.0/22    S   5                      >192.168.71.254
* 192.168.64.0/18    S   5                      >192.168.71.254
* 192.168.64.0/21    D   0                      >fxp0.0
* 192.168.71.246/32  L   0                       Local
* 192.168.220.4/30   D   0                      >ge-0/0/1.0
* 192.168.220.5/32   L   0                       Local
* 192.168.220.8/30   D   0                      >ge-0/0/2.0
* 192.168.220.9/32   L   0                       Local
* 192.168.220.12/30  D   0                      >ge-0/0/3.0
* 192.168.220.13/32  L   0                       Local
* 192.168.220.17/32  L   0                       Reject
* 192.168.220.21/32  L   0                       Reject
* 192.168.220.24/30  D   0                      >at-1/0/0.0
* 192.168.220.25/32  L   0                       Local
* 192.168.220.28/30  D   0                      >at-1/0/1.0
* 192.168.220.29/32  L   0                       Local
* 224.0.0.9/32       R 100          1            MultiRecv
```

Meaning The output shows a list of the routes that are currently in the inet.0 routing table. Verify the following information:

- Each configured static route is present. Routes are listed in ascending order by IP address. Static routes are identified with an S in the protocol (P) column of the output.
- Each static route is active. Routes that are active show the next-hop IP address in the Next hop column. If a route's next-hop address is unreachable, the next-hop address is identified as Reject. These routes are not active routes, but they appear in the routing table because the passive attribute is set.
- The preference for each static route is correct. The preference for a particular route is listed in the Prf column of the output.

Related Topics For a complete description of show route terse output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Chapter 14

Configuring a RIP Network

The Routing Information Protocol (RIP) is an interior gateway protocol that routes packets within a single autonomous system (AS). To use RIP, you must understand the basic components of a RIP network and configure the Juniper Networks device to act as a node in the network.

For an overview of RIPng, see "RIPng Overview" on page 314. For configuration instructions, see the *JUNOS Routing Protocols Configuration Guide*.

NOTE: Before configuring routing protocols on a device running JUNOS software, you must first configure security filters. For more information, see the *JUNOS Software Security Configuration Guide*.

NOTE: In general, in this guide, the term *RIP* refers to RIP version 1 (RIPv1) and RIP version 2 (RIPv2).

You can use either J-Web Quick Configuration or a configuration editor to configure a RIP network.

This chapter contains the following topics. For more information about RIP, see the *JUNOS Routing Protocols Configuration Guide*.

RIP Overview

To achieve basic connectivity between all RIP hosts in a RIP network, you enable RIP on every interface that is expected to transmit and receive RIP traffic. To enable RIP on an interface, you define RIP groups, which are logical groupings of interfaces, and add interfaces to the groups. Additionally, you must configure a routing policy to export directly connected routes and routes learned through RIP routing exchanges.

RIP Traffic Control with Metrics

To tune a RIP network and control traffic flowing through the network, you increase or decrease the cost of the paths through the network. RIP provides two ways to modify the path cost: an incoming metric and an outgoing metric, which are each set to 1 by default. These metrics are attributes that manually specify the cost of any route advertised through a host. By increasing or decreasing the metrics—and thus the cost—of links throughout the network, you can control packet transmission across the network.

The incoming metric modifies the cost of an individual segment when a route across the segment is imported into the routing table. For example, if you set the incoming metric on the segment to 3, the individual segment cost along the link is changed from 1 to 3. The increased cost affects all route calculations through that link. Other routes that were previously excluded because of a high hop count might now be selected into the router's forwarding table.

The outgoing metric modifies the path cost for all the routes advertised out a particular interface. Unlike the incoming metric, the outgoing metric modifies the routes that other routers are learning and thereby controls the way they send traffic.

Authentication

RIPv2 provides authentication support so that RIP links can require authentication keys (passwords) before they become active. These authentication keys can be specified in either plain-text or MD5 form. Authentication provides an additional layer of security on the network beyond the other security features.

This type of authentication is not supported on RIPv1 networks.

Before You Begin

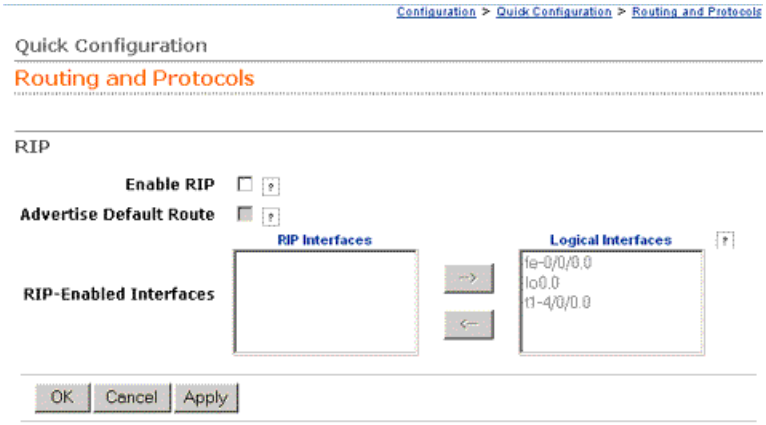
Before you begin configuring a RIP network, complete the following tasks:

- Establish basic connectivity. See the Getting Started Guide for your device.
- Configure network interfaces. See “Configuring Ethernet, DS1, DS3, and Serial Interfaces” on page 73.
- Configure security filters. See the *JUNOS Software Security Configuration Guide*.

Configuring a RIP Network with Quick Configuration

J-Web Quick Configuration allows you to create RIP networks. Figure 59 on page 347 shows the Quick Configuration Routing page for RIP.

Figure 59: Quick Configuration Routing Page for RIP



To configure a RIP network with Quick Configuration:

1. In the J-Web user interface, select **Configuration > Quick Configuration > Routing and Protocols**.
2. Enter information into the Quick Configuration page for RIP, as described in Table 105 on page 347.
3. From the main RIP routing Quick Configuration page, click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration Routing page for RIP, click **Apply**.
   - To apply the configuration and return to the Quick Configuration Routing page, click **OK**.
   - To cancel your entries and return to the Quick Configuration Routing page, click **Cancel**.
4. To check the configuration, see "Verifying the RIP Configuration" on page 356.

Table 105: RIP Routing Quick Configuration Summary

Field	Function	Your Action
RIP		
Enable RIP	Enables or disables RIP.	■ To enable RIP, select the check box. ■ To disable RIP, clear the check box.
Advertise Default Route	Advertises the default route using RIPv2.	■ To advertise the default route using RIPv2, select the check box. ■ To disable the default route advertisement, clear the check box.

Table 105: RIP Routing Quick Configuration Summary *(continued)*

Field	Function	Your Action
RIP-Enabled Interfaces	Designates one or more interfaces on which RIP is enabled. see "Network Interface Naming" on page 16.	The first time you configure RIP, the Logical Interfaces box displays a list of all the logical interfaces configured on the device. Do any of the following: ■ To enable RIP on an interface, click the interface name to highlight it, and click the left arrow to add the interface to the RIP interfaces list. ■ To enable RIP on multiple interfaces at once, press Ctrl while you click multiple interface names to highlight them. Then click the left arrow to add the interfaces to the RIP interfaces list. ■ To disable RIP on one or more interfaces, highlight the interface or interfaces in the RIP interfaces box and click the right arrow to move them back to the Logical Interfaces list.

Configuring a RIP Network with a Configuration Editor

To configure the Juniper Networks device as a node in a RIP network, you must perform the following task marked *(Required)*.

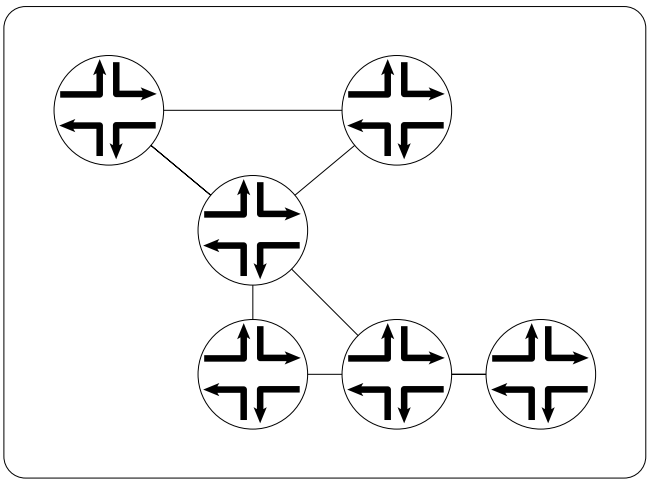
- Configuring a Basic RIP Network (Required) on page 348
- Controlling Traffic in a RIP Network (Optional) on page 351
- Enabling Authentication for RIP Exchanges (Optional) on page 354

For information about using the J-Web and CLI configuration editors, see the *J-Web Interface User Guide* and the *JUNOS CLI User Guide*.

Configuring a Basic RIP Network (Required)

To use RIP on the device, you must configure RIP on all the RIP interfaces within a network like the one shown in Figure 60 on page 349.

Figure 60: Typical RIP Network Topology



By default, RIP does not advertise the subnets that are directly connected through the device's interfaces. For traffic to pass through a RIP network, you must create a routing policy to export these routes. Advertising only the direct routes propagates the routes to the immediately adjacent RIP-enabled router only. To propagate all routes through the entire RIP network, you must configure the routing policy to export the routes learned through RIP.

To configure a RIP network like the one in Figure 60 on page 349, with a routing policy, perform these steps on each device in the network:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 106 on page 350.
3. If you are finished configuring the router, commit the configuration.

   After you add the appropriate interfaces to the RIP group, RIP begins sending routing information. No additional configuration is required to enable RIP traffic on the network.
4. Go on to one of the following procedures:
   - To control RIP traffic on the network, see "Controlling Traffic in a RIP Network (Optional)" on page 351.
   - To authenticate RIP exchanges, see "Enabling Authentication for RIP Exchanges (Optional)" on page 354.
   - To check the configuration, see "Verifying the RIP Configuration" on page 356.

Table 106: Configuring a RIP Network

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Rip** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Configure** or **Edit**. 3. Next to Rip, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter `edit protocols rip`
Create the RIP group alpha1.	1. Next to Group, click **Add new entry**. 2. In the Group name box, type alpha1.	1. Create the RIP group alpha1, and add an interface: `set group alpha1 neighbor ge-0/0/0.0` 2. Repeat Step 1 for each interface on this device that you are adding to the RIP group. Only one interface is required.
Add interfaces to the RIP group alpha1. For information about interface names, see "Network Interface Naming" on page 16.	1. Next to Neighbor, click **Add new entry**. 2. In the Neighbor name box, type the name of an interface on the device—for example, ge-0/0/0.0—and click **OK**. 3. Repeat Step 2 for each interface on this device that you are adding to the RIP group. Only one interface is required.	

Configure a routing policy to advertise directly connected routes.	1. On the main Configuration page next to Policy options, click **Configure** or **Edit**.
	2. Next to Policy statement, click **Add new entry**.
	3. In the Policy name box, type the name of the policy statement—for example, advertise-rip-routes.
	4. Next to Term, click **Add new entry**.
	5. In the Term name box, type the name of the policy statement—for example, from-direct.
	6. Next to From, click **Configure.**
	7. Next to Protocol, click **Add new entry**.
	8. From the Value list, select **Direct**.
	9. Click **OK** until you return to the Policy statement page.
	10. Next to Then, click **Configure.**
	11. From the Accept reject list, select **Accept**.
	12. Click **OK**.
		1. From the [edit] hierarchy level, enter
		`edit policy-options`
		2. Set the match condition to match on direct routes:
		`set policy-statement advertise-rip-routes term from-direct from protocol direct`
		3. Set the match action to accept these routes:
		`set policy-statement advertise-rip-routes term from-direct then accept`

Table 106: Configuring a RIP Network *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor

Configure the previous routing policy to advertise routes learned from RIP.	1. On the main Configuration page next to Policy options, click **Configure** or **Edit**.
	2. Next to Policy statement, click **advertise-rip-routes**.
	3. Next to Term, click **Add new entry**.
	4. In the Term name box, type the name of the policy statement—for example, from-rip.
	5. Next to From, click **Configure.**
	6. Next to Protocol, click **Add new entry**.
	7. From the Value list, select **rip**.
	8. Click **OK** until you return to the Policy statement page.
	9. Next to Then, click **Configure**.
	10. From the Accept reject list, select **Accept**.
	11. Click **OK**.
		1. From the [edit] hierarchy level, enter
		`edit policy-options`
		2. Set the match condition to match on direct routes:
		`set policy-statement advertise-rip-routes term from-rip from protocol rip`
		3. Set the match action to accept these routes:
		`set policy-statement advertise-rip-routes term from-rip then accept`

Controlling Traffic in a RIP Network (Optional)

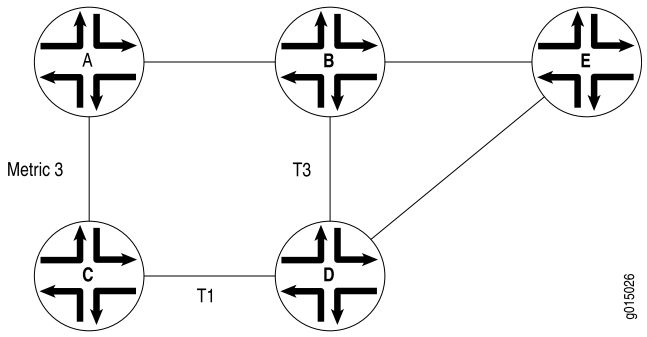
There are two primary means for controlling traffic in a RIP network: the incoming metric and the outgoing metric. To modify these attributes, see the following sections:

- Controlling Traffic with the Incoming Metric on page 351
- Controlling Traffic with the Outgoing Metric on page 353

Controlling Traffic with the Incoming Metric

Depending on the RIP network topology and the links between nodes in the network, you might want to control traffic flow through the network to maximize flow across higher-bandwidth links. Figure 61 on page 351 shows a network with alternate routes between Routers A and D.

Figure 61: Controlling Traffic in a RIP Network with the Incoming Metric



In this example, routes to Router D are received by Router A across both of its RIP-enabled interfaces. Because the route through Router B and the route through Router C have the same number of hops, both routes are imported into the forwarding table. However, because the T3 link from Router B to Router D has a higher bandwidth than the T1 link from Router C to Router D, you want traffic to flow from A through B to D.

To force this flow, you can modify the route metrics as they are imported into Router A's routing table. By setting the incoming metric on the interface from Router A to Router C, you modify the metric on all routes received through that interface. Setting the incoming route metric on Router A changes only the routes in Router A's routing table, and affects only how Router A sends traffic to Router D. Router D's route selection is based on its own routing table, which, by default, includes no adjusted metric values.

In the example, Router C receives a route advertisement from Router D and readvertises the route to Router A. When Router A receives the route, it applies the incoming metric on the interface. Instead of incrementing the metric by 1 (the default), Router A increments it by 3 (the configured incoming metric), giving the route from Router A to Router D through Router C a total path metric of 4. Because the route through Router B has a metric of 2, it becomes the preferred route for all traffic from Router A to Router D.

To modify the incoming metric on all routes learned on the link between Router A and Router C and force traffic through Router B:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 107 on page 352.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following procedures:
   - To authenticate RIP exchanges, see "Enabling Authentication for RIP Exchanges (Optional)" on page 354.
   - To check the configuration, see "Verifying the RIP Configuration" on page 356.

Table 107: Modifying the Incoming Metric

Task	J-Web Configuration Editor	CLI Configuration Editor
In the configuration hierarchy, navigate to the level of an interface in the alpha1 RIP group.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Rip, click **Edit**. 4. Under Group name, click **alpha1**. 5. Under Neighbor name, click the interface name—for example, **ge-0/0/0.0**.	From the [edit] hierarchy level, enter edit protocols rip group alpha1 neighbor ge-0/0/0

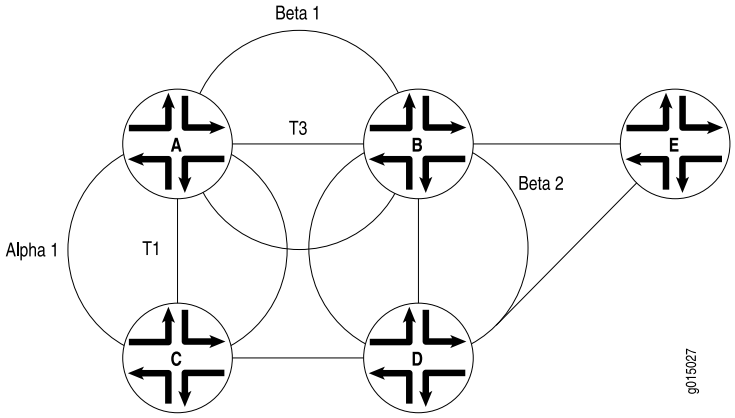
Table 107: Modifying the Incoming Metric *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Increase the incoming metric to 3.	In the Metric in box, type 3, and click **OK**.	Set the incoming metric to 3: set metric-in 3

Controlling Traffic with the Outgoing Metric

If an exported route was learned from a member of the same RIP group, the metric associated with that route is the normal RIP metric. For example, a RIP route with a metric of 5 learned from a neighbor configured with an incoming metric of 2 is advertised with a combined metric of 7 when advertised to neighbors in the same group. However, if this route was learned from a RIP neighbor in a different group or from a different protocol, the route is advertised with the metric value configured in the outgoing metric for that group. Figure 62 on page 353 shows a network with alternate routes between Routers A and D.

Figure 62: Controlling Traffic in a RIP Network with the Outgoing Metric



In this example, each route from Router A to Router D has two hops. However, because the link from Router A to Router B in RIP group Beta 1 has a higher bandwidth than the link from Router A to Router C in RIP group Alpha 1, you want traffic from Router D to Router A to flow through Router B. To control the way Router D sends traffic to Router A, you can alter the routes that Router D receives by configuring the outgoing metric on Router A's interfaces in the Alpha 1 RIP group.

If the outgoing metric for the Alpha 1 RIP group—the A-to-C link—is changed to 3, Router D calculates the total path metric from to A through C as 4. In contrast, the unchanged default total path metric to A through B in the Beta 1 RIP group is 2. The fact that Router A's interfaces belong to two different RIP groups allows you to configure two different outgoing metrics on its interfaces, because you configure path metrics at the group level.

By configuring the *incoming* metric, you control the way Router A sends traffic to Router D. By configuring the *outgoing* metric on the same router, you control the way Router D sends traffic to Router A.

To modify the outgoing metric on Router A and force traffic through Router B:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 108 on page 354.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following procedures:
   - To authenticate RIP exchanges, see "Enabling Authentication for RIP Exchanges (Optional)" on page 354.
   - To check the configuration, see "Verifying the RIP Configuration" on page 356.

Table 108: Modifying the Outgoing Metric

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **alpha1** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Rip, click **Edit**. 4. Under Group name, click **alpha1**.	From the [edit] hierarchy level, enter `edit protocols rip group alpha1`
Increase the outgoing metric to 3.	In the Metric out box, type 3, and click **OK**.	Set the outgoing metric to 3: `set metric-out 3`

Enabling Authentication for RIP Exchanges (Optional)

All RIPv2 protocol exchanges can be authenticated to guarantee that only trusted routers participate in the AS's routing. By default, this authentication is disabled. Authentication requires all routers within the RIP network or subnetwork to have the same authentication type and key (password) configured.

You can enable RIP authentication exchanges by either of the following methods:

- Enabling Authentication with Plain-Text Passwords on page 354
- Enabling Authentication with MD5 Authentication on page 355

Enabling Authentication with Plain-Text Passwords

To configure authentication that requires a plain-text password to be included in the transmitted packet, enable simple authentication by performing these steps on all RIP devices in the network:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 109 on page 355.
3. If you are finished configuring the router, commit the configuration.
4. To check the configuration, see "Verifying the RIP Configuration" on page 356.

Table 109: Configuring Simple RIP Authentication

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to **Rip** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Rip, click **Edit**.	From the [edit] hierarchy level, enter edit protocols rip
Set the authentication type to simple.	From the Authentication type list, select **simple**.	Set the authentication type to simple: set authentication-type simple
Set the authentication key to a simple-text password. The password can be from 1 through 16 contiguous characters long and can include any ASCII strings.	In the Authentication key box, type a simple-text password, and click **OK**.	Set the authentication key to a simple-text password: set authentication-key *password*

Enabling Authentication with MD5 Authentication

To configure authentication that requires an MD5 password to be included in the transmitted packet, enable MD5 authentication by performing these steps on all RIP devices in the network:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 110 on page 356.
3. If you are finished configuring the router, commit the configuration.
4. To check the configuration, see "Verifying the RIP Configuration" on page 356.

Table 110: Configuring MD5 RIP Authentication

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to **Rip** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Rip, click **Edit**.	From the [edit] hierarchy level, enter edit protocols rip
Set the authentication type to MD5.	From the Authentication type list, select **md5**.	Set the authentication type to md5: set authentication-type md5
Set the MD5 authentication key (password). The key can be from 1 through 16 contiguous characters long and can include any ASCII strings.	In the Authentication key box, type an MD5 authentication key, and click **OK**.	Set the MD5 authentication key: set authentication-key *password*

Verifying the RIP Configuration

To verify the RIP configuration, perform these tasks:

- Verifying the RIP-Enabled Interfaces on page 356
- Verifying the Exchange of RIP Messages on page 357
- Verifying Reachability of All Hosts in the RIP Network on page 358

Verifying the RIP-Enabled Interfaces

Purpose Verify that all the RIP-enabled interfaces are available and active.

Action From the CLI, enter the show rip neighbor command.

Sample Output

```
user@host> show rip neighbor
Source          Destination     Send   Receive   In
Neighbor          State  Address         Address         Mode   Mode     Met
--------          -----  -------         -----------     ----   -------  ---
ge-0/0/0.0           Dn (null)          (null)          mcast  both       1
ge-0/0/1.0           Up 192.168.220.5   224.0.0.9       mcast  both       1
```

Meaning The output shows a list of the RIP neighbors that are configured on the device. Verify the following information:

- Each configured interface is present. Interfaces are listed in alphabetical order.
- Each configured interface is up. The state of the interface is listed in the Destination State column. A state of Up indicates that the link is passing RIP traffic. A state of Dn indicates that the link is not passing RIP traffic. In a point-to-point link, this state generally means that either the end point is not configured for RIP or the link is unavailable.

Related Topics For a complete description of show rip neighbor output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying the Exchange of RIP Messages

Purpose Verify that RIP messages are being sent and received on all RIP-enabled interfaces.

Action From the CLI, enter the show rip statistics command.

Sample Output

```
user@host> show rip statistics
RIPv2 info: port 520; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              10              0              0              0

t1-0/0/2.0:  0 routes learned; 13 routes advertised; timeout 120s; update interval
 45s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                     2855           11            2
Triggered Updates Sent              5            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received             41            0            0
RIPv2 Bad Route Entries             0            0            0
RIPv2 Updates Ignored               0            0            0
Authentication Failures             0            0            0
RIP Requests Received               0            0            0
RIP Requests Ignored                0            0            0

ge-0/0/1.0:  10 routes learned; 3 routes advertised; timeout 180s; update interval
 30s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                     2855           11            2
Triggered Updates Sent              3            0            0
Responses Sent                      0            0            0
Bad Messages                        1            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received           2864           11            2
RIPv2 Bad Route Entries            14            0            0
RIPv2 Updates Ignored               0            0            0
Authentication Failures             0            0            0
RIP Requests Received               0            0            0
RIP Requests Ignored                0            0            0
```

Meaning The output shows the number of RIP routes learned. It also shows the number of RIP updates sent and received on the RIP-enabled interfaces. Verify the following information:

- The number of RIP routes learned matches the number of expected routes learned. Subnets learned by direct connectivity through an outgoing interface are not listed as RIP routes.

- RIP updates are being sent on each RIP-enabled interface. If no updates are being sent, the routing policy might not be configured to export routes.
- RIP updates are being received on each RIP-enabled interface. If no updates are being received, the routing policy might not be configured to export routes on the host connected to that subnet. The lack of updates might also might indicate an authentication error.

Related Topics For a complete description of show rip statistics output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying Reachability of All Hosts in the RIP Network

Purpose By using the traceroute tool on each loopback address in the network, verify that all hosts in the RIP network are reachable from each Juniper Networks device.

Action For each device in the RIP network:

1. In the J-Web interface, select **Diagnose > Traceroute**.
2. In the Remote Host box, type the name of a host for which you want to verify reachability from the device.
3. Click **Start**. Output appears on a separate page.

Sample Output

```
1 172.17.40.254 (172.17.40.254) 0.362 ms 0.284 ms 0.251 ms
2 routera-fxp0.englab.mycompany.net (192.168.71.246) 0.251 ms 0.235 ms 0.200 ms
```

Meaning Each numbered row in the output indicates a router (“hop”) in the path to the host. The three time increments indicate the round-trip time (RTT) between the device and the hop, for each traceroute packet.

To ensure that the RIP network is healthy, verify the following information:

- The final hop in the list is the host you want to reach.
- The number of expected hops to the host matches the number of hops in the traceroute output. The appearance of more hops than expected in the output indicates that a network segment is probably unreachable. It might also indicate that the incoming or outgoing metric on one or more hosts has been set unexpectedly.

Related Topics For information about the traceroute command and its output, see the *JUNOS System Basics and Services Command Reference*.

Chapter 15

Configuring an OSPF Network

The Open Shortest Path First protocol (OSPF) is an interior gateway protocol (IGP) that routes packets within a single autonomous system (AS). To use OSPF, you must understand the basic components of an OSPF network and configure the J-series Services Router to act as a node in the network.

NOTE: Before configuring routing protocols on a device running JUNOS software, you must first configure security filters. For more information, see the *JUNOS Software Security Configuration Guide*.

NOTE: In this chapter, the term *OSPF* refers to OSPF version 2 and OSPF version 3.

You can use either J-Web Quick Configuration or a configuration editor to configure an OSPF network.

This chapter contains the following topics. For more information about OSPF, see the *JUNOS Routing Protocols Configuration Guide*.

OSPF Overview

In an OSPF network, the network topology is distributed among the nodes of the autonomous system (AS) and is regularly updated. Because topology changes are flooded throughout the network, every node maintains the same copy of the network map in its local topological database. Packets are then routed based on the shared topology.

Enabling OSPF

To activate OSPF on a network, you must enable the protocol on all interfaces within the network on which OSPF traffic is to travel. To enable OSPF on one or more interfaces, you must configure one or more interfaces on the Services Router within an OSPF area. Once the interfaces are configured, OSPF link-state advertisements (LSAs) are transmitted on all OSPF-enabled interfaces, and the network topology is shared throughout the network.

OSPF Areas

OSPF is enabled on a per-interface basis. Those interfaces are configured as OSPF enabled, and are assigned to an area. In a simple, single-area network, the area has the numeric identifier 0.0.0.0, which designates it as the backbone area. As the network grows, it is divided into multiple subnetworks or areas that are identified by numeric identifiers unique to the AS.

In a multiarea network, all areas must be directly connected to the backbone area by area border routers (ABRs). Because all areas are adjacent to the backbone area, OSPF routers send all traffic not destined for their own area through the backbone area. The ABRs in the backbone area are then responsible for transmitting the traffic through the appropriate ABR to the destination area.

Path Cost Metrics

Once the topology is shared across the network, OSPF uses it to route packets between network nodes. Each path between neighbors is assigned a cost based on the throughput, round-trip time, and reliability of the link. The sum of the costs across a particular path between hosts determines the overall cost of the path. Packets are then routed along the shortest path using the shortest path first (SPF) algorithm. If multiple equal-cost paths exist between a source and destination address, OSPF routes packets along each path alternately, in round-robin fashion.

OSPF allows you to manually assign a cost (or metric) to a particular path segment to control the flow of packets across the network.

OSPF Dial-on-Demand Circuits

If you are configuring OSPF across a demand circuit such as an ISDN link, you must enable dial-on-demand routing backup on the OSPF-enabled interface. Because demand circuits do not pass all traffic required to maintain an OSPF adjacency (hello packets, for example), you configure dial-on-demand routing so individual nodes in an OSPF network can maintain adjacencies despite the lack of LSA exchanges.

To configure an ISDN link, see “Configuring ISDN” on page 177. For information about configuring OSPF demand circuits, see “Configuring Dial-on-Demand Routing Backup with OSPF Support (Optional)” on page 199.

Before You Begin

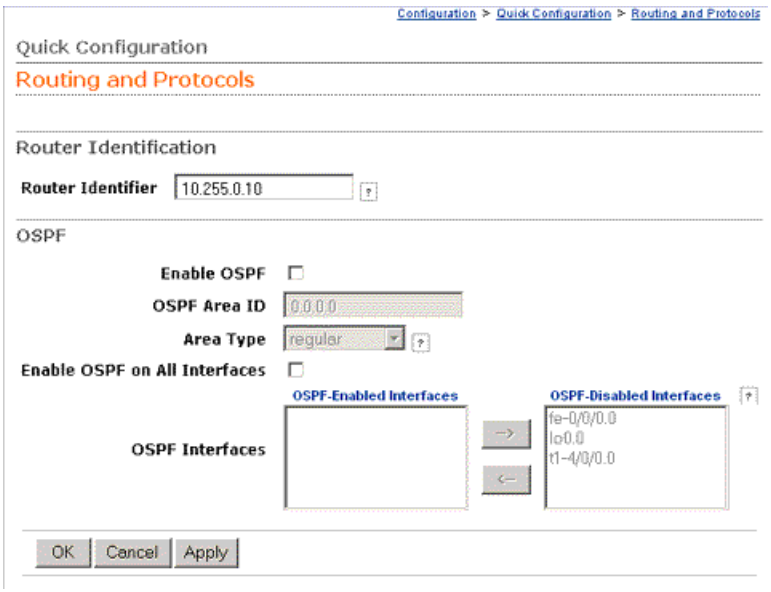
Before you begin configuring an OSPF network, complete the following tasks:

- Establish basic connectivity. See the Getting Started Guide for your device.
- Configure network interfaces. See "Configuring Ethernet, DS1, DS3, and Serial Interfaces" on page 73.
- Configure security filters. See the *JUNOS Software Security Configuration Guide*.

Configuring an OSPF Network with Quick Configuration

J-Web Quick Configuration allows you to create single-area OSPF networks. Figure 63 on page 361 shows the Quick Configuration Routing page for OSPF.

Figure 63: Quick Configuration Routing Page for OSPF



To configure a single-area OSPF network with Quick Configuration:

1. In the J-Web user interface, select **Configuration > Quick Configuration > Routing > OSPF Routing**.
2. Enter information into the Quick Configuration Routing page for OSPF, as described in Table 111 on page 362.
3. Click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration Routing page for OSPF, click **Apply**.
   - To apply the configuration and return to the Quick Configuration Routing page, click **OK**.
   - To cancel your entries and return to the Quick Configuration Routing page, click **Cancel**.
4. To check the configuration, see "Verifying an OSPF Configuration" on page 374.

Table 111: OSPF Routing Quick Configuration Summary

Field	Function	Your Action
Router Identification		
Router Identifier (required)	Uniquely identifies the router.	Type the device's 32-bit IP address, in dotted decimal notation.
OSPF		
Enable OSPF	Enables or disables OSPF.	■ To enable OSPF, select the check box. ■ To disable OSPF, clear the check box.
OSPF Area ID	Uniquely identifies the area within its AS.	Type a 32-bit numeric identifier for the area, or an integer. If you enter an integer, the value is converted to a 32-bit equivalent. For example, if you enter 3, the value assigned to the area is 0.0.0.3.
Area Type	Designates the type of OSPF area.	From the list, select the type of OSPF area you are creating: ■ **regular**—A regular OSPF area, including the backbone area ■ **stub**—A stub area ■ **nssa**—A not-so-stubby area (NSSA)
OSPF-Enabled Interfaces	Designates one or more interfaces on which OSPF is enabled. see "Network Interface Naming" on page 16.	The first time you configure OSPF, the Logical Interfaces box displays a list of all the logical interfaces configured on the device. Do any of the following: ■ To enable OSPF on an interface, click the interface name to highlight it, and click the left arrow to add the interface to the OSPF interfaces list. ■ To enable OSPF on multiple interfaces at once, press Ctrl while you click multiple interface names to highlight them. Then click the left arrow to add the interfaces to the OSPF interfaces list. ■ To enable OSPF on all logical interfaces except the special fxp0 management interface, select **All Interfaces** in the Logical Interfaces list and click the left arrow. ■ To enable OSPF on all the interfaces displayed in the Logical Interfaces list, click **All** to highlight every interface. Then click the left arrow to add the interfaces to the OSPF interfaces list. ■ To disable OSPF on one or more interfaces, highlight the interface or interfaces in the OSPF interfaces box and click the right arrow to move them back to the Logical Interfaces list.

Configuring an OSPF Network with a Configuration Editor

To configure the Services Router as a node in an OSPF network, you must perform the following tasks marked *(Required)*.

To configure an OSPF demand circuit, see “Configuring Dial-on-Demand Routing Backup with OSPF Support (Optional)” on page 199. (You must have already configured an ISDN interface as described in “Configuring ISDN” on page 177.)

For information about using the J-Web and CLI configuration editors, see the *J-Web Interface User Guide* and the *JUNOS CLI User Guide*.

Configuring the Router Identifier (Required)

The router identifier is the IP address that uniquely identifies the J-series Services Router.

OSPF uses the router identifier to elect a designated router, unless you manually specify a priority value. When the OSPF network first becomes active, by default, the router with the highest router identifier is elected the designated router.

To configure the router identifier for the Services Router:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 112 on page 363.
3. Go on to “Configuring a Single-Area OSPF Network (Required)” on page 363.

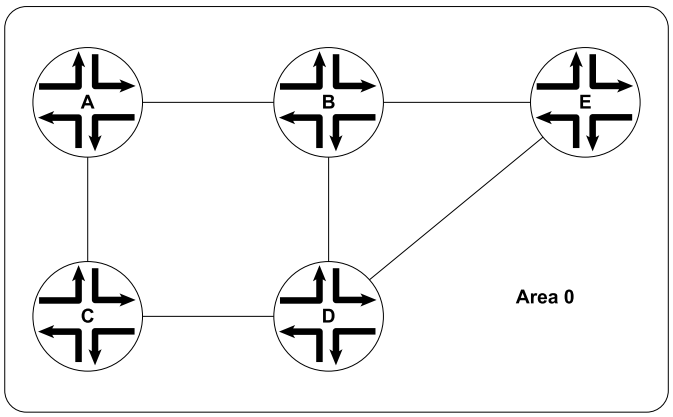
Table 112: Configuring the Router Identifier

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Routing-options** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing options, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit routing-options
Set the router ID value to the IP address of the Services Router—for example, 177.162.4.24.	1. In the Router Id box, type 177.162.4.24. 2. Click **OK**.	Enter set router-id 177.162.4.24

Configuring a Single-Area OSPF Network (Required)

To use OSPF on the Services Router, you must configure at least one OSPF area, like the one shown in Figure 64 on page 364.

Figure 64: Typical Single-Area OSPF Network Topology



To configure a single-area OSPF network with a backbone area, like the one in Figure 64 on page 364, perform these steps on each Services Router in the network:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 113 on page 365.
3. If you are finished configuring the router, commit the configuration.

   After you create the backbone area and add the appropriate interfaces to the area, OSPF begins sending LSAs. No additional configuration is required to enable OSPF traffic on the network.
4. Go on to one of the following procedures:
   - To add more areas to the AS, see "Configuring a Multiarea OSPF Network (Optional)" on page 365.
   - To control external route advertisement in the AS, see "Configuring Stub and Not-So-Stubby Areas (Optional)" on page 368.
   - To configure an OSPF demand circuit, see "Configuring Dial-on-Demand Routing Backup with OSPF Support (Optional)" on page 199. (You must have already configured an ISDN interface as described in "Configuring ISDN" on page 177.)
   - To improve network operation, see "Tuning an OSPF Network for Efficient Operation" on page 370.
   - To check the configuration, see "Verifying an OSPF Configuration" on page 374.

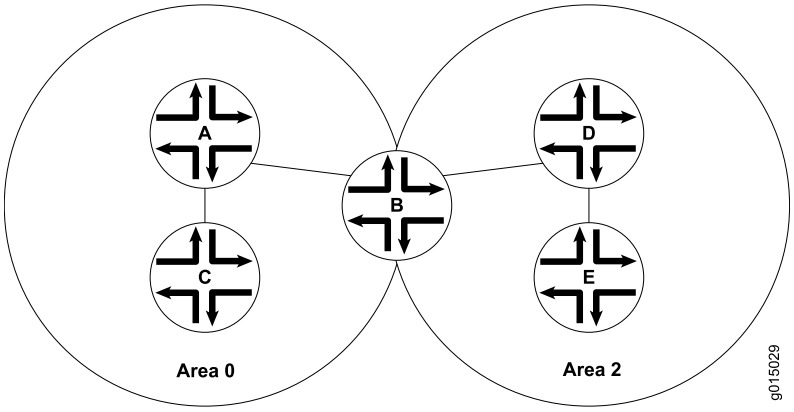
Table 113: Configuring a Single-Area OSPF Network

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Ospf** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Configure** or **Edit**. 3. Next to Ospf, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit protocols ospf
Create the backbone area with area ID 0.0.0.0.	1. In the Area box, click **Add new entry**. 2. In the Area ID box, type 0.0.0.0.	1. Set the backbone area ID to 0.0.0.0 and add an interface: set area 0.0.0.0 interface ge-0/0/0 2. Repeat Step 1 for each interface on this Services Router that you are adding to the backbone area. Only one interface is required.
Add interfaces as needed to the OSPF area—for example, ge-0/0/0. For information about interface names, see "Network Interface Naming" on page 16.	1. In the Interface box, click **Add new entry**. 2. In the Interface name box, type ge-0/0/0. 3. Click **OK**. 4. Repeat Step 1 through Step 3 for each interface on this Services Router that you are adding to the backbone area. Only one interface is required.	

Configuring a Multiarea OSPF Network (Optional)

To reduce traffic and topology maintenance for the Services Routers in an OSPF autonomous system (AS), you can group them into multiple areas, as shown in Figure 65 on page 365.

Figure 65: Typical Multiarea OSPF Network Topology



To configure a multiarea OSPF network shown in Figure 65 on page 365, perform the following tasks on the appropriate Services Routers in the network. You must create

a backbone area. To link each additional area to the backbone area, you must configure one of the Services Routers as an area border router (ABR).

Creating the Backbone Area

On each Services Router that is to operate as an ABR in the network, create backbone area 0.0.0.0 with at least one interface enabled for OSPF.

For instruction, see "Configuring a Single-Area OSPF Network (Required)" on page 363.

Creating Additional OSPF Areas

To create additional OSPF areas:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 114 on page 366.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following procedures:
   - To configure this device as an area border router, see "Configuring Area Border Routers" on page 367.
   - To control external route advertisement in the AS, see "Configuring Stub and Not-So-Stubby Areas (Optional)" on page 368.
   - To configure an OSPF demand circuit, see "Configuring Dial-on-Demand Routing Backup with OSPF Support (Optional)" on page 199. (You must have already configured an ISDN interface as described in "Configuring ISDN" on page 177.)
   - To improve network operation, see "Tuning an OSPF Network for Efficient Operation" on page 370.
   - To check the configuration, see "Verifying an OSPF Configuration" on page 374.

Table 114: Configuring a Multiarea OSPF Network

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Ospf** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Ospf, click **Edit**.	From the [edit] hierarchy level, enter edit protocols ospf

Table 114: Configuring a Multiarea OSPF Network *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Create the additional area with a unique area ID, in dotted decimal notation—for example, 0.0.0.2.	1. In the Area box, click **Add new entry**. 2. In the Area ID box, type 0.0.0.2.	1. Set the area ID to 0.0.0.2 and add an interface: set area 0.0.0.2 interface ge-0/0/0
Add interfaces as needed to the OSPF area—for example, ge-0/0/0.	1. In the Interface box, click **Add new entry**. 2. In the Interface name box, type ge-0/0/0. 3. Click **OK**. 4. Repeat Step 1 through Step 3 for each interface on this Services Router that you are adding to the backbone area. Only one interface is required.	2. Repeat Step 1 for each interface on this Services Router that you are adding to the area. Only one interface is required.

Configuring Area Border Routers

A Services Router operating as an area border router (ABR) has interfaces enabled for OSPF in the backbone area and in the area you are linking to the backbone. For example, Services Router B acts as the ABR in Figure 65 on page 365 and has interfaces in both the backbone area and area 0.0.0.3.

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 115 on page 368.
3. If you are finished configuring the router, commit the configuration.

   After you create the areas on the appropriate Services Routers and add and enable the appropriate interfaces to the areas, no additional configuration is required to enable OSPF traffic within or across the areas.
4. Go on to one of the following procedures:
   - To control external route advertisement in the AS, see "Configuring Stub and Not-So-Stubby Areas (Optional)" on page 368.
   - To configure an OSPF demand circuit, see "Configuring Dial-on-Demand Routing Backup with OSPF Support (Optional)" on page 199. (You must have already configured an ISDN interface as described in "Configuring ISDN" on page 177.)
   - To improve network operation, see "Tuning an OSPF Network for Efficient Operation" on page 370.
   - To check the configuration, see "Verifying an OSPF Configuration" on page 374.

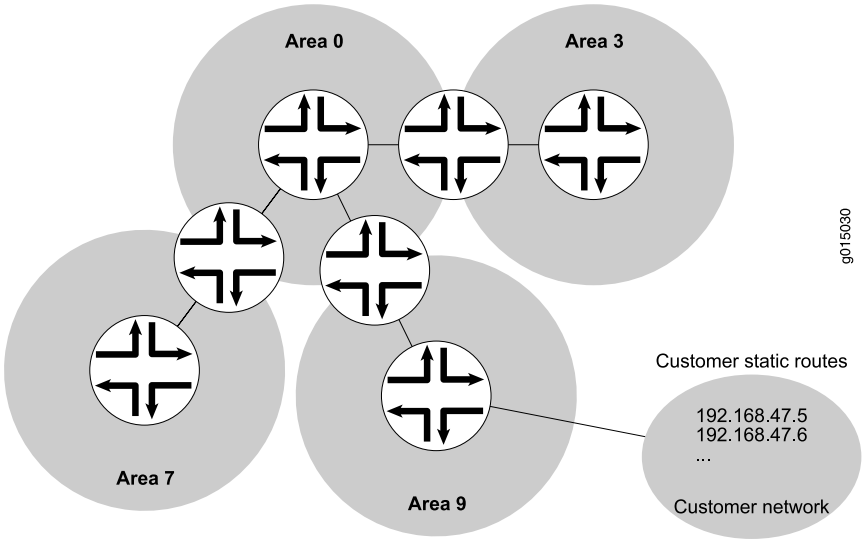
Table 115: Configuring Area Border Routers

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Ospf** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Ospf, click **Edit**.	From the [edit] hierarchy level, enter edit protocols ospf
Verify that the backbone area has at least one interface enabled for OSPF.	Click **0.0.0.0** to display the Area ID 0.0.0.0 page, and verify that the backbone area has at least one interface enabled for OSPF. For example, Services Router B in Figure 65 on page 365 has the following interfaces enabled for OSPF in the backbone area: ■ Interface ge-0/0/0.0 ■ Interface ge-0/0/1.0 To enable an interface on the backbone area, see "Configuring a Single-Area OSPF Network (Required)" on page 363.	View the configuration using the show command: show For example, Services Router B in Figure 65 on page 365 has the following interfaces enabled for OSPF in the backbone area: area 0.0.0.0 { interface ge-0/0/0.0; interface ge-0/0/1.0; } To enable an interface on the backbone area, see "Configuring a Single-Area OSPF Network (Required)" on page 363.
Create the additional area with a unique area ID—for example, 0.0.0.2.	1. In the Area box, click **Add new entry**. 2. In the Area ID box, type 0.0.0.2.	1. Set the area ID to 0.0.0.2 and add an interface: set area 0.0.0.2 interface ge-0/0/0 2. Repeat Step 1 for each interface on this Services Router that you are adding to the area. Only one interface is required.
Add interfaces as needed to the OSPF area—for example, ge-0/0/0.	1. In the Interface box, click **Add new entry**. 2. In the Interface name box, type ge-0/0/0. 3. Click **OK**. 4. Repeat Step 1 through Step 3 for each interface on this Services Router that you are adding to the backbone area. Only one interface is required.	

Configuring Stub and Not-So-Stubby Areas (Optional)

To control the advertisement of external routes into an area, you can create stub areas and not-so-stubby areas (NSSAs) in an OSPF network. In the network shown in Figure 66 on page 369, area 0.0.0.7 has no external connections and can be configured as a stub area. Area 0.0.0.9 only has external connections to static routes and can be configured as an NSSA.

Figure 66: OSPF Network Topology with Stub Areas and NSSAs



To configure stub areas and NSSAs in an OSPF network like the one shown in Figure 66 on page 369:

1. Create the area and enable OSPF on the interfaces within that area.

   For instructions, see "Creating Additional OSPF Areas" on page 366.

2. Configure an area border router to bridge the areas.

   For instructions, see "Configuring Area Border Routers" on page 367.

3. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.

4. To configure each Services Router in area 0.0.0.7 as a stub area router, perform the configuration tasks described in Table 116 on page 370.

5. If you are finished configuring the router, commit the configuration.

6. Go on to one of the following procedures:

   - To configure an OSPF demand circuit, see "Configuring Dial-on-Demand Routing Backup with OSPF Support (Optional)" on page 199. (You must have already configured an ISDN interface as described in "Configuring ISDN" on page 177.)
   - To improve network operation, see "Tuning an OSPF Network for Efficient Operation" on page 370.
   - To check the configuration, see "Verifying an OSPF Configuration" on page 374.

Table 116: Configuring Stub Area and Not-So-Stubby Area Routers

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **0.0.0.7** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Ospf, click **Edit**. 4. Under Area id, click **0.0.0.7**.	From the [edit] hierarchy level, enter edit protocols ospf area 0.0.0.7
Configure each Services Router in area 0.0.0.7 as a stub router.	1. In the Stub option list, select **Stub** and click **OK**. 2. Repeat Step 1 for every Services Router in the stub area to configure them with the stub parameter for the area.	1. Set the stub attribute: set stub 2. Repeat Step 1 for every Services Router in the stub area to configure them with the stub parameter for the area.
Navigate to the **0.0.0.9** level in the configuration hierarchy.	1. On the main Configuration page next to Protocols, click **Edit**. 2. Next to Ospf, click **Edit**. 3. Under Area id, click **0.0.0.9**.	From the [edit] hierarchy level, enter edit protocols ospf area 0.0.0.9
Configure each Services Router in area 0.0.0.9 as an NSSA router.	1. In the Stub option list, select **Nssa** and click **OK**. 2. Repeat Step 1 for every Services Router in the NSSA to configure them with the nssa parameter for the area.	1. Set the nssa attribute: set nssa 2. Repeat Step 1 for every Services Router in the NSSA to configure them with the nssa parameter for the area.

Tuning an OSPF Network for Efficient Operation

To make your OSPF network operate more efficiently, you can change some default settings on the Services Router by performing the following tasks:

- Controlling Route Selection in the Forwarding Table on page 370
- Controlling the Cost of Individual Network Segments on page 371
- Enabling Authentication for OSPF Exchanges on page 372
- Controlling Designated Router Election on page 373

Controlling Route Selection in the Forwarding Table

OSPF uses route preferences to select the route that is installed in the forwarding table when several routes have the same shortest path first (SFP) calculation. To evaluate a route, OSPF calculates the sum of the individual preferences of every router along the path and selects the route with the lowest total preference.

By default, internal OSPF routes have a preference value of 10, and external OSPF routes have a preference value of 150. Suppose all routers in your OSPF network use the default preference values. By setting the internal preference to 7 and the external preference to 130, you can ensure that the path through a particular Services Router is selected for the forwarding table any time multiple equal-cost paths to a destination exist.

To modify the default preferences on a Services Router:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 117 on page 371.

Table 117: Controlling Route Selection in the Forwarding Table by Setting Preferences

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Ospf** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Ospf, click **Edit**.	From the [edit] hierarchy level, enter edit protocols ospf
Set the external and internal route preferences.	1. In the External preference box, type 130. 2. In the Preference box, type the internal preference value of 7. 3. Click **OK**.	1. Set the external preference: set external-preference 130 2. Set the internal preference: set preference 7

Controlling the Cost of Individual Network Segments

When evaluating the cost of individual network segments, OSPF evaluates the reference bandwidth. For any link faster than 100 Mbps, the default cost metric is 1. When OSPF calculates the SPF algorithm, it sums the metrics of all interfaces along a path to determine the overall cost of the path. The path with the lowest metric is selected for the forwarding table.

To control the cost of the network segment, you can modify the metric value on an individual interface. Suppose all routers in the OSPF network use default metric values. If you increase the metric on an interface to 5, all paths through this interface have a calculated metric higher than the default and are *not* preferred.

To manually set the cost of a network segment on the stub area's Fast Ethernet interface by modifying the interface metric:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 118 on page 372.

Table 118: Controlling the Cost of Individual Network Segments by Modifying the Metric

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **ge-0/0/0.0** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Ospf, click **Edit**. 4. Under Area id, click **0.0.0.0**. 5. Under Interface name, click **ge-0/0/0.0**.	From the [edit] hierarchy level, enter edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0
Set the interface metric.	1. In the Metric box, type the interface metric value 5. 2. Click **OK**.	Set the interface metric: set metric 5

Enabling Authentication for OSPF Exchanges

All OSPFv2 protocol exchanges can be authenticated to guarantee that only trusted routers participate in the AS's routing. By default, OSPF authentication is disabled.

NOTE: OSPFv3 does not support authentication.

You can enable either of two authentication types:

- Simple authentication—Authenticates by means of a plain-text password (key) included in the transmitted packet.
- MD5 authentication—Authenticates by means of an MD5 checksum included in the transmitted packet.

Because OSPF performs authentication at the area level, all routers within the area must have the same authentication and corresponding password (key) configured. For MD5 authentication to work, both the receiving and transmitting routers must have the same MD5 key.

To enable OSPF authentication on the stub area:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 119 on page 373.

Table 119: Enabling OSPF Authentication

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **0.0.0.0** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Ospf, click **Edit**. 4. Under Area id, click **0.0.0.0**.	From the [edit] hierarchy level, enter edit protocols ospf area 0.0.0.0
Set the authentication type for the stub area to either simple or MD5—for example, MD5.	1. From the Authentication type list, select **md5**. 2. Click **OK**.	Set the authentication type: set authentication-type md5
Navigate to the *interface-name* level in the configuration hierarchy.	1. On the main Configuration page next to Protocols, click **Edit**. 2. Next to Ospf, click **Edit**. 3. Under Area id, click **0.0.0.0**. 4. Under Interface name, click an interface name.	From the [edit] hierarchy level, enter edit protocols ospf area 0.0.0.0 interface *interface-name*

Set the authentication password (key) and, for MD5 authentication only, the key identifier to associate with the MD5 password:
■ For simple authentication, set a password of from 1 through 8 ASCII characters—for example, Chey3nne.
■ For MD5 authentication:
  ■ Set a password of from 1 through 16 ASCII characters—for example, Chey3nne.
  ■ Set a key identifier between 0 (the default) and 255—for example, 2.
	1. In the Key name box, type Chey3nne.
	2. For MD5 authentication only, in the Key ID box, type 2.
	3. Click **OK**.
	4. Repeat Step 1 through Step 3 for each interface in the stub area for which you are enabling authentication.
		1. Set the authentication password and, for MD5 authentication only, set the key identifier:
		set authentication-key Chey3nne key-id 2
		2. Repeat Step 1 for each interface in the stub area for which you are enabling authentication.

Controlling Designated Router Election

At designated router election, the router priorities are evaluated first, and the router with the highest priority is elected designated router.

By default, routers have a priority of 128. A priority of 0 marks the router as ineligible to become the designated router. To configure a router so it is always the designated router, set its priority to 255.

To change the priority of a Services Router to control designated router election:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 120 on page 374.

Table 120: Controlling Designated Router Election

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the OSPF interface address for the Services Router. For example, navigate to the ge-/0/0/1 level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Ospf, click **Edit**. 4. Under Area id, click **0.0.0.3**. 5. Under Interface name, click **ge-0/0/1**.	From the [edit] hierarchy level, enter edit protocols ospf area 0.0.0.3 interface ge-0/0/1
Set the Services Router priority to a value between 0 and 255—for example, 200. The default value is 128.	1. In the Priority box, type 200. 2. Click **OK**.	Set the priority value: set priority 200

Verifying an OSPF Configuration

To verify an OSPF configuration, perform these tasks:

- Verifying OSPF-Enabled Interfaces on page 374
- Verifying OSPF Neighbors on page 375
- Verifying the Number of OSPF Routes on page 376
- Verifying Reachability of All Hosts in an OSPF Network on page 377

Verifying OSPF-Enabled Interfaces

Purpose Verify that OSPF is running on a particular interface and that the interface is in the desired area.

Action From the CLI, enter the show ospf interface command.

Sample Output

```
user@host> show ospf interface
Intf                State     Area            DR ID           BDR ID       Nbrs
at-5/1/0.0          PtToPt    0.0.0.0         0.0.0.0         0.0.0.0         1
ge-2/3/0.0          DR        0.0.0.0         192.168.4.16    192.168.4.15    1
lo0.0               DR        0.0.0.0         192.168.4.16    0.0.0.0         0
so-0/0/0.0          Down      0.0.0.0         0.0.0.0         0.0.0.0         0
so-6/0/1.0          PtToPt    0.0.0.0         0.0.0.0         0.0.0.0         1
so-6/0/2.0          Down      0.0.0.0         0.0.0.0         0.0.0.0         0
so-6/0/3.0          PtToPt    0.0.0.0         0.0.0.0         0.0.0.0         1
```

Meaning The output shows a list of the Services Router interfaces that are configured for OSPF. Verify the following information:

- Each interface on which OSPF is enabled is listed.
- Under Area, each interface shows the area for which it was configured.
- Under Intf and State, the Services Router loopback (lo0.0) interface and LAN interface that are linked to the OSPF network's designated router (DR) are identified.
- Under DR ID, the IP address of the OSPF network's designated router appears.
- Under State, each interface shows a state of PtToPt to indicate a point-to-point connection. If the state is Waiting, check the output again after several seconds. A state of Down indicates a problem.
- The designated router addresses always show a state of DR.

Related Topics For a complete description of show ospf interface output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying OSPF Neighbors

Purpose OSPF neighbors are interfaces that have an immediate adjacency. On a point-to-point connection between the Services Router and another router running OSPF, verify that each router has a single OSPF neighbor.

Action From the CLI, enter the show ospf neighbor command.

Sample Output

```
user@host> show ospf neighbor
 Address         Intf              State      ID              Pri  Dead
192.168.254.225  fxp3.0             2Way      10.250.240.32    128   36
192.168.254.230  fxp3.0             Full      10.250.240.8     128   38
192.168.254.229  fxp3.0             Full      10.250.240.35    128   33
10.1.1.129       fxp2.0             Full      10.250.240.12    128   37
10.1.1.131       fxp2.0             Full      10.250.240.11    128   38
10.1.2.1         fxp1.0             Full      10.250.240.9     128   32
10.1.2.81        fxp0.0             Full      10.250.240.10    128   33
```

Meaning The output shows a list of the Services Router's OSPF neighbors and their addresses, interfaces, states, router IDs, priorities, and number of seconds allowed for inactivity ("dead" time). Verify the following information:

- Each interface that is immediately adjacent to the Services Router is listed.
- The Services Router's own loopback address and the loopback addresses of any routers with which the Services Router has an immediate adjacency are listed.
- Under State, each neighbor shows a state of Full. Because full OSPF connectivity is established over a series of packet exchanges between clients, the OSPF link might take several seconds to establish. During that time, the state might be displayed as Attempt, Init, or 2way, depending on the stage of negotiation.

  If, after 30 seconds, the state is not Full, the OSPF configuration between the neighbors is not functioning correctly.

Related Topics For a complete description of show ospf neighbor output, see the *JUNOS Routing Protocols and Policies Command Reference*.

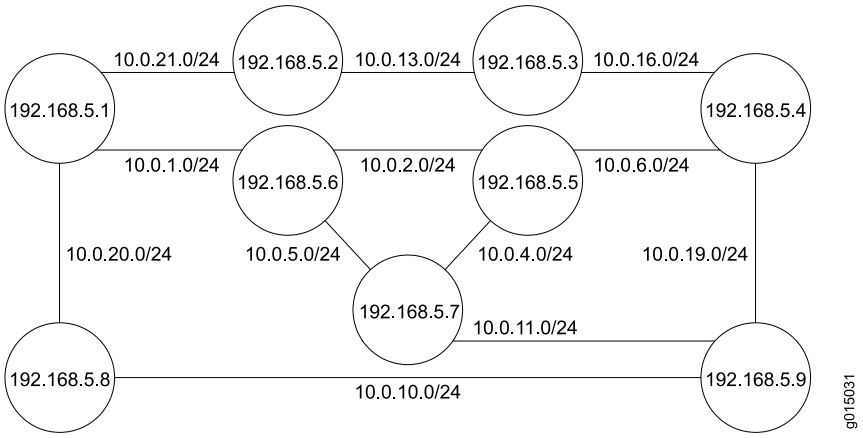
Verifying the Number of OSPF Routes

Purpose Verify that the OSPF routing table has entries for the following:

- Each subnetwork reachable through an OSPF link
- Each loopback address reachable on the network

For example, Figure 67 on page 376 shows a sample network with an OSPF topology.

Figure 67: Sample OSPF Network Topology



In this topology, OSPF is being run on all interfaces. Each segment in the network is identified by an address with a /24 prefix, with interfaces on either end of the segment being identified by unique IP addresses.

Action From the CLI, enter the show ospf route command.

Sample Output

```
user@host> show ospf route
Prefix               Path   Route       NH   Metric  NextHop       Nexthop

                     Type   Type        Type         Interface     addr/label
10.10.10.1/24        Intra  Network     IP   1       ge-0/0/2.0    10.0.21.1
10.10.10.2/24        Intra  Network     IP   1       ge-0/0/2.0    10.0.21.1
10.10.10.4/24        Intra  Network     IP   1       ge-0/0/1.0    10.0.13.1
10.10.10.5/24        Intra  Network     IP   1       ge-0/0/2.0    10.0.21.1
10.10.10.6/24        Intra  Network     IP   1       ge-0/0/1.0    10.0.13.1
10.10.10.10/24       Intra  Network     IP   1       ge-0/0/2.0    10.0.21.1
10.10.10.11/24       Intra  Network     IP   1       ge-0/0/1.0    10.0.13.1
10.10.10.13/24       Intra  Network     IP   1       ge-0/0/1.0
10.10.10.16/24       Intra  Network     IP   1       ge-0/0/1.0    10.0.13.1
10.10.10.19/24       Intra  Network     IP   1       ge-0/0/1.0    10.0.13.1
10.10.10.20/24       Intra  Network     IP   1       ge-0/0/2.0    10.0.21.1
10.10.10.21/24       Intra  Network     IP   1       ge-0/0/2.0
192.168.5.1          Intra  Router      IP   1       ge-0/0/2.0    10.0.21.1
192.168.5.2          Intra  Router      IP   1       lo0
192.168.5.3          Intra  Router      IP   1       ge-0/0/1.0    10.0.13.1
192.168.5.4          Intra  Router      IP   1       ge-0/0/1.0    10.0.13.1
192.168.5.5          Intra  Router      IP   1       ge-0/0/1.0    10.0.13.1
192.168.5.6          Intra  Router      IP   1       ge-0/0/2.0    10.0.21.1
192.168.5.7          Intra  Router      IP   1       ge-0/0/2.0    10.0.21.1
```

```
192.168.5.8          Intra  Router      IP   1       ge-0/0/2.0    10.0.21.1
192.168.5.9          Intra  Router      IP   1       ge-0/0/1.0    10.0.13.1
```

Meaning The output lists each route, sorted by IP address. Routes are shown with a route type of Network, and loopback addresses are shown with a route type of Router.

For the example shown in Figure 67 on page 376, verify that the OSPF routing table has 21 entries, one for each network segment and one for each router's loopback address.

Related Topics For a complete description of show ospf route output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying Reachability of All Hosts in an OSPF Network

Purpose By using the traceroute tool on each loopback address in the network, verify that all hosts in the network are reachable from each Services Router.

Action For each Services Router in the OSPF network:

1. In the J-Web interface, select **Diagnose > Traceroute**.
2. In the Host Name box, type the name of a host for which you want to verify reachability from the Services Router.
3. Click **Start**. Output appears on a separate page.

Sample Output

```
1 172.17.40.254 (172.17.40.254) 0.362 ms 0.284 ms 0.251 ms
2 routera-fxp0.englab.mycompany.net (192.168.71.246) 0.251 ms 0.235 ms 0.200 ms
```

Meaning Each numbered row in the output indicates a router ("hop") in the path to the host. The three time increments indicate the round-trip time (RTT) between the Services Router and the hop, for each traceroute packet. To ensure that the OSPF network is healthy, verify the following information:

- The final hop in the list is the host you want to reach.
- The number of expected hops to the host matches the number of hops in the traceroute output. The appearance of more hops than expected in the output indicates that a network segment is likely not reachable. In this case, verify the routes with the show ospf route command.

For information about show ospf route, see "Verifying the Number of OSPF Routes" on page 376.

Related Topics For information about the traceroute command and its output, see the *JUNOS System Basics and Services Command Reference*.

Chapter 16

Configuring the IS-IS Protocol

You use either the J-Web configuration editor or CLI configuration editor to configure IS-IS.

This chapter contains the following topics. For more information about IS-IS, see the *JUNOS Routing Protocols Configuration Guide*.

IS-IS Overview

On the Services Router, Intermediate System-to-Intermediate System (IS-IS) protocol is an interior gateway routing protocol (IGP) that uses link-state information for routing network traffic. IS-IS uses the shortest path first (SPF) algorithm to determine routes. Using SPF, IS-IS evaluates network topology changes and determines if a full or partial route calculation is required. The protocol was originally developed for routing International Organization for Standards (ISO) connectionless network protocol (CLNP) packets.

This overview contains the following topics:

ISO Network Addresses

IS-IS uses ISO network addresses. Each address identifies a point of connection to the network, such as a router interface, which is called a network service access point (NSAP). NSAP addresses are supported on the loopback (lo0) interface. (For information about interface names, see "Network Interface Naming" on page 16.)

An end system can have multiple NSAP addresses, which differ by the last byte called an n-selector. Each NSAP represents a service that is available at the node. In addition to multiple services, a single node can belong to multiple areas.

Each network entity also has a special address called a network entity title (NET) with an identical structure to an NSAP address but an n-selector of 00. Most end

systems and intermediate systems have one NET address, while intermediate systems participating in more than one area can have more than one NET address.

The following ISO addresses are examples of the IS-IS address format:

49.0001.00a0.c96b.c490.00

49.0001.2081.9716.9018.00

The first part of the address is the area number, which is a variable number from 1 to 13 bytes. The first byte of the area number, 49, is the authority and format indicator (AFI). The next bytes are the assigned area identifier and can be from 0 to 12 bytes. In the examples, 0001 is the area identifier.

The next 6 bytes are the system identifier and can be any 6 bytes unique throughout the entire domain. The system identifier is commonly the media access control (MAC) address, as shown in the first example, 00a0.c96b.c490. Otherwise, the system identifier is the IP address expressed in binary-coded decimal (BCD) format, as shown in the second example, 2081.9716.9018, which corresponds to 208.197.169.18. The last byte, 00, is the n-selector.

NOTE: The system identifier cannot be configured as 0000.0000.0000. Using all zeros as an identifier is not supported and does not form an adjacency.

System Identifier Mapping

To provide assistance with debugging IS-IS, the Services Router supports dynamic mapping of ISO system identifiers to the hostname. Each router can be configured with a hostname that allows the system identifier-to-hostname mapping to be sent in a dynamic hostname type length value (TLV) in the IS-IS link-state PDU (LSP). The mapping permits an intermediate system in the routing domain to learn the ISO system identifier of another intermediate system.

Before You Begin

Before you begin configuring IS-IS, complete the following tasks:

- Establish basic connectivity. See the Getting Started Guide for your device.
- If your router is in secure context, enable IS-IS. By default in secure context, the router drops IS-IS packets. Enable the router in one of the following ways to forward IS-IS packets:
  - In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. To reach the correct J-Web page, select **Configure** or **Edit** next to Security, Forwarding options, Family, and finally Iso. Next to Mode, select packet-based. Click **OK**.
  - From configuration mode in the CLI, enter the command set security forwarding-options family iso mode packet-based.

NOTE: JUNOS software security processing is not applied to IS-IS packets forwarded by the router.

- If you do not already have an understanding of IS-IS, read "IS-IS Overview" on page 320 or the *JUNOS Routing Protocols Configuration Guide*.
- Obtain ISO addresses for participating routers in the AS.
- Configure security filters. See the *JUNOS Software Security Configuration Guide*.

Configuring IS-IS with a Configuration Editor

To configure IS-IS with a configuration editor, you do the following:

- Enable IS-IS on the router.
- Configure a network entity title (NET) on one of the router interfaces, preferably the loopback interface, lo0.
- Configure the ISO family on all interfaces that are supporting the IS-IS protocol.

To configure IS-IS:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 121 on page 381.
3. Commit the configuration on the Services Router.
4. Repeat the configuration tasks on each Services Router in the IS-IS autonomous system (AS).

Table 121: Configuring the IS-IS Protocol

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Edit**.	From the [edit] hierarchy level, enter edit interfaces.
Configure the loopback interface lo0.	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type lo0. 3. Click **OK**.	Enter edit interfaces lo0

Table 121: Configuring the IS-IS Protocol *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure the logical unit on the loopback interface—for example 0. Add the NET address to the loopback interface—for example, 49.0001.00a0.c96b.c490.00.	1. Next to lo0, click **Edit** under Encapsulation. 2. Next to Unit, click **Add new entry**. 3. In the Interface unit number box, type 0. 4. Under Family, select **Iso**. 5. Next to Address, click **Add new entry**. 6. In the Source box, type 49.0001.00a0.c96b.c490.00. 7. Click **OK** until you return to the Interfaces page.	1. Enter edit unit 0 2. Enter set family iso address 49.0001.00a0.c96b.c490.00
Configure a physical interface—for example, ge-0/0/1—with the NET address, and add the Family type iso.	1. Next to ge-0/0/1, click **Edit** under Encapsulation. 2. Next to Unit, click **Add new entry**. 3. In the Interface unit number box, type 0. 4. Under Family, select **Iso**. 5. Next to Iso, click **Configure**. 6. Next to Address, click **Add new entry**. 7. In the Source box, type 49.0001.00a0.c96b.c490.00. 8. Click **OK** until you return to the Edit Configuration page.	Enter edit interfaces ge-0/0/1 Enter set unit 0 Enter set family iso address 49.0001.00a0.c96b.c490.00
Navigate to the **Protocols** level in the configuration hierarchy.	On the main Configuration page next to Protocols, click **Edit**.	From the [edit] hierarchy level, enter edit protocols
Add the IS-IS protocol to all interfaces on the Services Router.	1. Next to Isis, click **Edit**. 2. In the Interface name box, type all. 3. Click **OK**.	Enter set isis interface all

Verifying IS-IS on a Services Router

To verify IS-IS, perform these tasks:

- Displaying IS-IS Interface Configuration on page 383
- Displaying IS-IS Interface Configuration Detail on page 383

Displaying IS-IS Interface Configuration

Purpose Verify the status of IS-IS-enabled interfaces.

Action From the CLI, enter the show isis interface brief command.

Sample Output

```
user@host> show isis interface brief
IS-IS interface database:
Interface  L CirID Level 1 DR Level 2 DR
lo0.0      3  0x1  router1  router.01
ge-0/0/1.0 2  0x9  Disabled  router.03
ge-1/0/0.0 2 0x7  Disabled  router.05
```

Meaning Verify that the output shows the intended configuration of the interfaces on which IS-IS is enabled.

Related Topics For a complete description of show isis interface output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Displaying IS-IS Interface Configuration Detail

Purpose Verify the details of IS-IS-enabled interfaces.

Action From the CLI, enter the show isis interface detail command.

Sample Output

```
user@host> show isis interface detail
lo0.0
   Index:3, State:0x7, Circuit id: 0x1, Circuit type:3
   LSP interval: 100 ms, Sysid: router1
   Level Adjacencies Priority Metric Hello(s) Hold(s)
       1           0       64      0       9    27
       2           0       64      0       9    27
  ge-0/0/1.0
       Index:3, State:0x106, Circuit id: 0x9, Circuit type:2
   LSP interval: 100 ms, Sysid: router1
   Level Adjacencies Priority Metric Hello(s) Hold(s)
       1           0       64      0       9    27
       2           0       64      0       9    27
```

Meaning Check the following output fields and verify that the output shows the intended configuration of IS-IS-enabled interfaces:

- Interface—Interface configured for IS-IS
- State—Internal implementation information
- Circuit id—Circuit identifier
- Circuit type—Configured level of IS-IS:
  - 1—Level 1 only
  - 2—Level 2 only
  - 3—Level 1 and Level 2

- LSP interval—Time between IS-IS information messages
- Sysid—System identifier
- L or Level—Type of adjacency:
  - 1—Level 1 only
  - 2—Level 2 only
  - 3—Level 1 and Level 2
- Adjacencies—Adjacencies established on the interface
- Priority—Priority value established on the interface
- Metric—Metric value for the interface
- Hello(s)—Intervals between hello PDUs
- Hold(s)—Hold time on the interface

Related Topics For a complete description of show isis interface detail output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Displaying IS-IS Adjacencies

Purpose Display brief information about IS-IS neighbors.

Action From the CLI, enter the show isis adjacency brief command.

Sample Output

```
user@host> show isis adjacency brief
IS-IS adjacency database:
   Interface  System    L  State    Hold (secs) SNPA
   ge-0/0/0.0  1921.6800.5067  2  Up        13
   ge-0/0/1.0  1921.6800.5067  2 Up        25
   ge-0/0/2.0  1921.6800.5067  2  Up        19
```

Meaning Verify adjacent routers in the IS-IS database.

Related Topics For a complete description of show isis adjacency brief output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Displaying IS-IS Adjacencies in Detail

Purpose Display extensive information about IS-IS neighbors.

Action From the CLI, enter the show isis adjacency extensive command.

Sample Output

```
user@host> show isis adjacency extensive
R1
  Interface: so-0/0/0.0, Level: 2, State: Up, Expires in 25 secs
  Priority: 0, Up/Down transitions: 1, Last transition: 4w6d 19:38:52 ago
  Circuit type: 2, Speaks: IP, IPv6
  Topologies: Unicast
  Restart capable: Yes
  IP addresses: 10.1.12.1
  Transition log:
  When                  State        Reason
```

```
  Wed Jul 13 16:26:11   Up           Seenself

R3
  Interface: so-0/0/1.0, Level: 2, State: Up, Expires in 23 secs
  Priority: 0, Up/Down transitions: 1, Last transition: 6w5d 19:07:16 ago
  Circuit type: 2, Speaks: IP, IPv6
  Topologies: Unicast
  Restart capable: Yes
  IP addresses: 10.1.23.2
  Transition log:
  When                  State        Reason
  Thu Jun 30 16:57:46   Up           Seenself

R6
  Interface: so-0/0/2.0, Level: 2, State: Up, Expires in 25 secs
  Priority: 0, Up/Down transitions: 1, Last transition: 6w0d 18:01:18 ago
  Circuit type: 2, Speaks: IP, IPv6
  Topologies: Unicast
  Restart capable: Yes
  IP addresses: 10.1.26.2
  Transition log:
  When                  State        Reason
  Tue Jul  5 18:03:45   Up           Seenself
```

Meaning Check the following fields and verify adjacency information about IS-IS neighbors:

- Interface—Interface through which the neighbor is reachable
- L or Level—Configured level of IS-IS:
  - 1—Level 1 only
  - 2—Level 2 only
  - 3—Level 1 and Level 2

  An exclamation point before the level number indicates that the adjacency is missing an IP address.
- State—Status of the adjacency: Up, Down, New, One-way, Initializing, or Rejected
- Event—Message that identifies the cause of a state
- Down reason—Reason the adjacency is down
- Restart capable—Denotes a neighbor configured for graceful restart
- Transition log—List of transitions including When, State, and Reason

Related Topics For a complete description of show isis adjacency extensive output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Chapter 17

Configuring BGP Sessions

Connections between peering networks are typically made through an exterior gateway protocol, most commonly the Border Gateway Protocol (BGP).

NOTE: Before configuring routing protocols on a device running JUNOS software, you must first configure security filters. For more information, see the *JUNOS Software Security Configuration Guide*.

You can use either J-Web Quick Configuration or a configuration editor to configure BGP sessions.

This chapter contains the following topics. For more information about BGP, see the *JUNOS Routing Protocols Configuration Guide*.

BGP Overview

BGP is a heavy-duty, secure protocol that must be configured on a per-peer basis. Once a peering session has been configured, BGP uses a TCP connection to establish a session. After a BGP session is established, traffic is passed along the BGP-enabled link.

Although BGP requires a full-mesh topology to share route information, you can use route reflectors and confederations in a large autonomous system (AS) to reduce scaling problems.

BGP Peering Sessions

Unlike RIP and OSPF links, BGP peering sessions must be explicitly configured at both ends. To establish a session between BGP peers, you must manually specify the interface address to which you are establishing a connection. Once this configuration is complete on both ends of a link, a TCP negotiation takes place and a BGP session is established.

The type of the BGP peering session depends on whether the peer is outside or inside the host's autonomous system (AS):

- Peering sessions established with hosts outside the local AS are external sessions. Traffic that passes along such links uses external BGP (EBGP) as its protocol.
- Peering sessions established with hosts within the local AS are internal sessions. Traffic that passes along such links uses internal BGP (IBGP) as its protocol.

To monitor BGP neighbors, see the information about real-time performance monitoring (RPM) in the *JUNOS Software Administration Guide*.

IBGP Full Mesh Requirement

By default, BGP does not readvertise routes that are learned from BGP. To share route information throughout the network, BGP requires a full mesh of internal peering sessions within an AS. To achieve an IBGP full mesh, you configure a direct peering session every host to every other host within the network. These sessions are configured on every router within the network, as type internal.

Route Reflectors and Clusters

In larger networks, the overhead needed to implement the IBGP full-mesh requirement is prohibitive. Many networks use route reflectors to avoid having to configure an internal connection to each node for every new router.

NOTE: You must have an Advanced BGP Feature license installed on each device that uses a route reflector. For license details, see the *JUNOS Software Administration Guide*.

A route reflector can readvertise routes learned through BGP to its BGP neighbors. If you define clusters of routers and configure a single router as a route reflector within each cluster, a full mesh is required only between the route reflectors and all their internal peers within the network. The route reflector is responsible for propagating BGP routes throughout the cluster.

For more information about route reflectors, see “Route Reflectors—for Added Hierarchy” on page 330

BGP Confederations

Large ASs can be divided into smaller sub-ASs, which are groups of routers known as confederations. You configure EBGP peering sessions between confederations, and IBGP peering sessions within confederations. Within a confederation, the IBGP full mesh is required. For more information about confederations, see “Confederations—for Subdivision” on page 332

Before You Begin

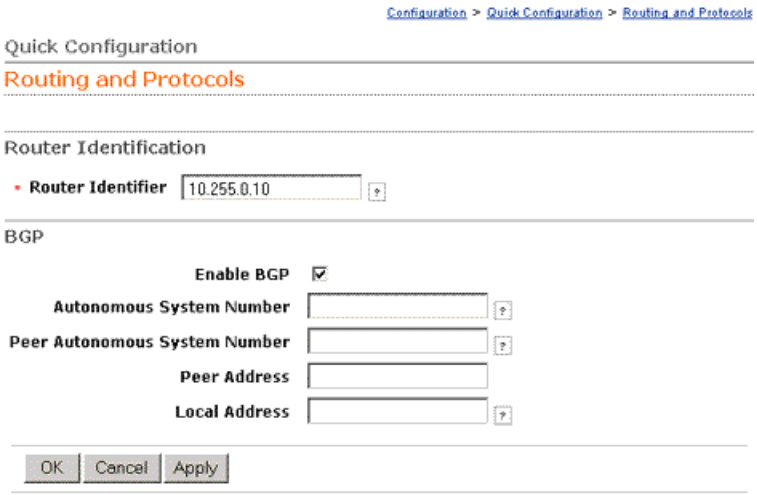
Before you begin configuring BGP sessions, complete the following tasks:

- Establish basic connectivity. See the Getting Started Guide for your device.
- Configure network interfaces. See "Configuring Ethernet, DS1, DS3, and Serial Interfaces" on page 73.
- Configure security filters. See the *JUNOS Software Security Configuration Guide*.

Configuring BGP Sessions with Quick Configuration

J-Web Quick Configuration allows you to create BGP peering sessions. Figure 68 on page 389 shows the Quick Configuration Routing page for BGP.

Figure 68: Quick Configuration Routing Page for BGP



To configure a BGP peering session with Quick Configuration:

1. In the J-Web user interface, select **Configuration > Quick Configuration > Routing and Protocols**.
2. Enter information into the Quick Configuration page for BGP, as described in Table 122 on page 390.
3. From the main BGP routing Quick Configuration page, click one of the following buttons:
   - To apply the configuration and stay on the Quick Configuration Routing page for BGP, click **Apply**.
   - To apply the configuration and return to the Quick Configuration Routing page, click **OK**.
   - To cancel your entries and return to the Quick Configuration Routing page, click **Cancel**.
4. To check the configuration, see "Verifying a BGP Configuration" on page 398.

Table 122: BGP Routing Quick Configuration Summary

Field	Function	Your Action
Router Identification		
Router Identifier (required)	Uniquely identifies the router	Type the device's 32-bit IP address, in dotted decimal notation.
BGP		
Enable BGP	Enables or disables BGP.	■ To enable BGP, select the check box. ■ To disable BGP, clear the check box.
Autonomous System Number	Sets the unique numeric identifier of the AS in which the device is configured.	Type the device's 32-bit AS number, in dotted decimal notation. If you enter an integer, the value is converted to a 32-bit equivalent. For example, if you enter 3, the value assigned to the AS is 0.0.0.3.
Peer Autonomous System Number	Sets the unique numeric identifier of the AS in which the peer host resides.	Type the peer host's 32-bit AS number, in dotted decimal notation. If you enter an integer, the value is converted to a 32-bit equivalent. For example, if you enter 3, the value assigned to the AS is 0.0.0.3.
Peer Address	Specifies the IP address of the peer host's interface to which the BGP session is being established.	Type the IP address of the peer host's adjacent interface, in dotted decimal notation.
Local Address	Specifies the IP address of the local host's interface from which the BGP session is being established.	Type the IP address of the local host's adjacent interface, in dotted decimal notation.

Configuring BGP Sessions with a Configuration Editor

To configure the device as a node in a BGP network, you must perform the following tasks marked *(Required)*.

- Configuring Point-to-Point Peering Sessions (Required) on page 390
- Configuring BGP Within a Network (Required) on page 393
- Configuring a Route Reflector (Optional) on page 394
- Configuring BGP Confederations (Optional) on page 397

For information about using the J-Web and CLI configuration editors, see the *J-Web Interface User Guide* and the *JUNOS CLI User Guide*.

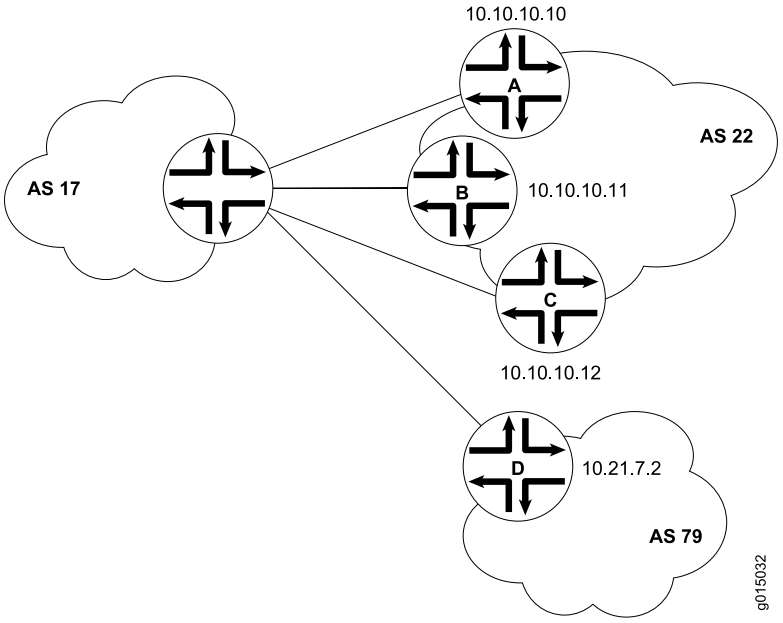
Configuring Point-to-Point Peering Sessions (Required)

To enable BGP traffic across one or more links, you must configure a BGP peering session with the adjacent host. Generally, such sessions are made at network exit

points with neighboring hosts outside the autonomous system. Figure 69 on page 391 shows a network with BGP peering sessions.

In the sample network, a device in AS 17 has BGP peering sessions to a group of peers called external-peers. Peers A, B, and C reside in AS 22 and have IP addresses 10.10.10.10, 10.10.10.11, and 10.10.10.12. Peer D resides in AS 79, at IP address 10.21.7.2.

Figure 69: Typical Network with BGP Peering Sessions



To configure the BGP peering sessions shown in Figure 69 on page 391:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 123 on page 392.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following procedures:
   - To configure IBGP sessions between peers, see "Configuring BGP Within a Network (Required)" on page 393.
   - To configure route reflector clusters, see "Configuring a Route Reflector (Optional)" on page 394.
   - To subdivide autonomous systems (ASs), see "Configuring BGP Confederations (Optional)" on page 397.
   - To check the configuration, see "Verifying a BGP Configuration" on page 398.

Table 123: Configuring BGP Peering Sessions

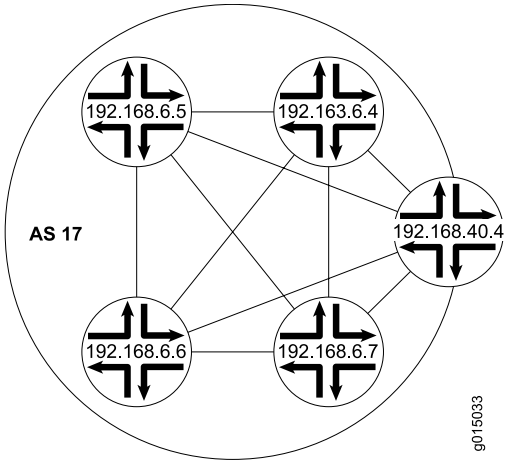
Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Routing options** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing options, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit routing-options
Set the network's AS number to 17.	1. In the AS Number box, enter 17. 2. Click **OK**.	Set the AS number to 17: set autonomous-system 17
Navigate to the **Bgp** level in the configuration hierarchy.	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Bgp, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit protocols bgp
Create the BGP group external-peers, and add the external neighbor addresses to the group.	1. In the Group box, click **Add new entry**. 2. In the Group name box, type the name of the group of external BGP peers—external-peers in this case. 3. In the Neighbor box, click **Add new entry**. 4. In the Address box, type the IP address of an external BGP peer, in dotted decimal notation, and click **OK**. 5. Repeat Step 3 and Step 4 for each BGP neighbor within the external group that you are configuring.	1. Create the group external-peers, and add the address of an external neighbor: set group external-peers neighbor 10.10.10.10 2. Repeat Step 1 for each BGP neighbor within the external peer group that you are configuring.
At the group level, set the AS number for the group external-peers to 22. Because three of the peers in this group (peers A, B, and C) reside in one AS, you can set their AS number as a group.	1. In the Peer as box, type the number of the AS in which most peers in the external-peers group reside. 2. Click **OK**.	From the [edit protocols bgp] hierarchy level: set group external-peers peer-as 22
At the individual neighbor level, set the AS number for peer D to 79. Because peer D is a member of the group external-peers, it inherits the peer AS number configured at the group level. You must override this value at the individual neighbor level.	1. Under Neighbor, in the Address column, click the IP address of peer D—10.21.7.2 in this case. 2. In the Peer as box, type the AS number of the peer. 3. Click **OK**.	From the [edit protocols bgp group external-peers] hierarchy level: set neighbor 10.21.7.2 peer-as 79
Set the group type to external.	1. From the Type list, select **external**. 2. Click **OK**.	From the [edit protocols bgp group external-peers] hierarchy level: set type external

Configuring BGP Within a Network (Required)

To configure BGP sessions between peering networks, you must configure point-to-point sessions between the external peers of the networks. Additionally, you must configure BGP internally to provide a means by which BGP route advertisements can be forwarded throughout the network. Because of the full mesh requirement of IBGP, you must configure individual peering sessions between all internal nodes of the network—unless you use route reflectors or confederations.

Figure 70 on page 393 shows a typical network with external and internal peer sessions. In the sample network, the device in AS 17 is fully meshed with its internal peers in the group internal-peers, which have IP addresses starting at 192.168.6.4.

Figure 70: Typical Network with EBGP External Sessions and IBGP Internal Sessions



To configure IBGP in the network shown in Figure 70 on page 393:

1. Configure all external peering sessions as described in "Configuring Point-to-Point Peering Sessions (Required)" on page 390.
2. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
3. Perform the configuration tasks described in Table 124 on page 394.
4. If you are finished configuring the router, commit the configuration.
5. Go on to one of the following procedures:
   - To configure route reflector clusters, see "Configuring a Route Reflector (Optional)" on page 394.
   - To subdivide autonomous systems (ASs), see "Configuring BGP Confederations (Optional)" on page 397.
   - To check the configuration, see "Verifying a BGP Configuration" on page 398.

Table 124: Configuring IBGP Peering Sessions

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Bgp** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Bgp, click **Edit**.	From the [edit] hierarchy level, enter edit protocols bgp
Create the BGP group internal-peers, and add the internal neighbor addresses to the group. You must configure a full IBGP mesh, which requires that each peer be configured with every other internal peer as a BGP neighbor.	1. In the Group box, click **Add new entry**. 2. In the Group name box, type the name of the group of internal BGP peers—internal-peers in this case. 3. In the Neighbor box, click **Add new entry**. 4. In the Address box, type the IP address of an internal BGP peer, in dotted decimal notation. 5. Click **OK**. 6. Repeat Step 3 and Step 4 for each internal BGP peer within the network.	1. Create the group internal-peers, and add the address of an internal neighbor: set group internal-peers neighbor 192.168.6.4 2. Repeat Step 1 for each internal BGP neighbor within the network.
Set the group type to internal.	1. From the Type list, select **internal**. 2. Click **OK**.	From the [edit protocols bgp group internal-peers] hierarchy level: set type internal
Configure a routing policy to advertise BGP routes.	See "Injecting OSPF Routes into the BGP Routing Table (Optional)" on page 510.	

Configuring a Route Reflector (Optional)

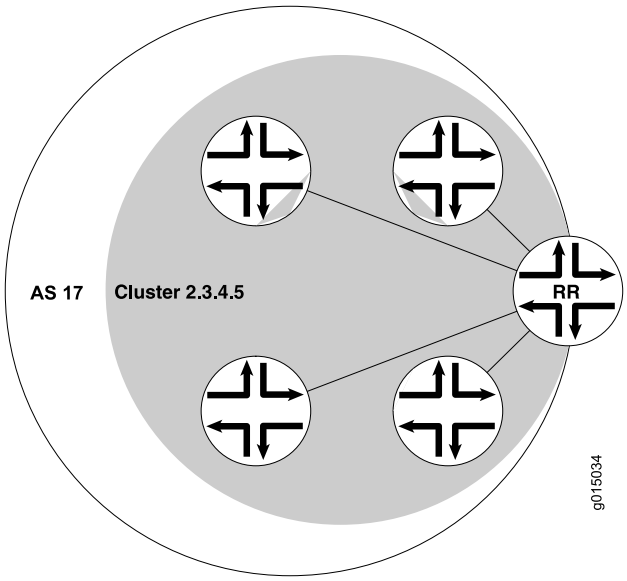
Because of the IBGP full-mesh requirement, most networks use route reflectors to simplify configuration. Using a route reflector, you group routers into clusters, which are identified by numeric identifiers unique to the AS. Within the cluster, you must configure a BGP session from a single router (the route reflector) to each internal peer. With this configuration, the IBGP full-mesh requirement is met.

NOTE: You must have an Advanced BGP Feature license installed on each device that uses a route reflector. For license details, see the *JUNOS Software Administration Guide*

Figure 71 on page 395 shows an IBGP network with a Juniper Networks device at IP address 192.168.40.4 acting as a route reflector. In the sample network, each device in Cluster 2.3.4.5 has an internal client relationship to the route reflector. To configure the cluster:

- Create an internal group on the Juniper Networks device, configure an internal peer (neighbor) relationship to every other device in the cluster, and assign a cluster identifier.
- On the other devices you are assigning to the cluster, create the cluster group and configure a client relationship to the route reflector.

Figure 71: Typical IBGP Network Using a Route Reflector



To configure IBGP in the network using the Juniper Networks device as a route reflector:

1. Configure all external peering sessions as described in "Configuring Point-to-Point Peering Sessions (Required)" on page 390.
2. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
3. Perform the configuration tasks described in Table 125 on page 396.
4. If you are finished configuring the router, commit the configuration.
5. Go on to one of the following procedures:
   - To subdivide autonomous systems (ASs), see "Configuring BGP Confederations (Optional)" on page 397.
   - To check the configuration, see "Verifying a BGP Configuration" on page 398.

Table 125: Configuring a Route Reflector

Task	J-Web Configuration Editor	CLI Configuration Editor
On the device that you are using as a route reflector, navigate to the **Bgp** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Bgp, click **Edit**.	From the [edit] hierarchy level, enter edit protocols bgp
On the device that you are using as a route reflector, create the BGP group cluster-peers, and add to the group the IP addresses of the internal neighbors that you want in the cluster.	1. In the Group box, click **Add new entry**. 2. In the Group name box, type the name of the group in which the BGP peer is configured—cluster-peers in this case. 3. In the Neighbor box, click **Add new entry**. 4. In the Address box, type the IP address of a BGP peer, in dotted decimal notation. 5. Click **OK**. 6. Repeat Step 3 and Step 4 for each BGP neighbor within the cluster that you are configuring.	1. Create the group cluster-peers, and add the address of an internal neighbor: set group cluster-peers neighbor 192.168.6.4 2. Repeat Step 1 for each BGP neighbor within the cluster that you are configuring.
On the device that you are using as a route reflector, set the group type to internal.	From the Type list, select **internal**.	From the [edit protocols bgp group internal-peers] hierarchy level: set type internal
On the device that you are using as a route reflector, configure the cluster identifier for the route reflector.	1. In the Cluster box, enter the unique numeric cluster identifier. 2. Click **OK**.	Set the cluster identifier: set cluster 2.3.4.5
On the other routers in the cluster, create the BGP group cluster-peers, and add the internal IP address of the route reflector. You do not need to include the neighbor addresses of the other internal peers, or configure the cluster identifier on these route reflector clients. They need only be configured as internal neighbors. **NOTE:** If the other routers in the network are Juniper Networks devices, follow the steps in this row. Otherwise, consult the router documentation for instructions.	On a client device in the cluster: 1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Edit**. 3. Next to Bgp, click **Edit**. 4. In the Group box, click **Add new entry**. 5. In the Group name box, type the name of the group in which the BGP peer is configured—cluster-peers in this case. 6. In the Neighbor box, click **Add new entry**. 7. In the Address box, type the IP address of the route reflector, in dotted decimal notation—in this case, 192.168.40.4. 8. Click **OK**.	On a client device in the cluster: 1. From the [edit] hierarchy level, enter edit protocols bgp 2. Create the group cluster-peers, and add only the route reflector address to the group: set group cluster-peers neighbor 192.168.40.4

Table 125: Configuring a Route Reflector *(continued)*

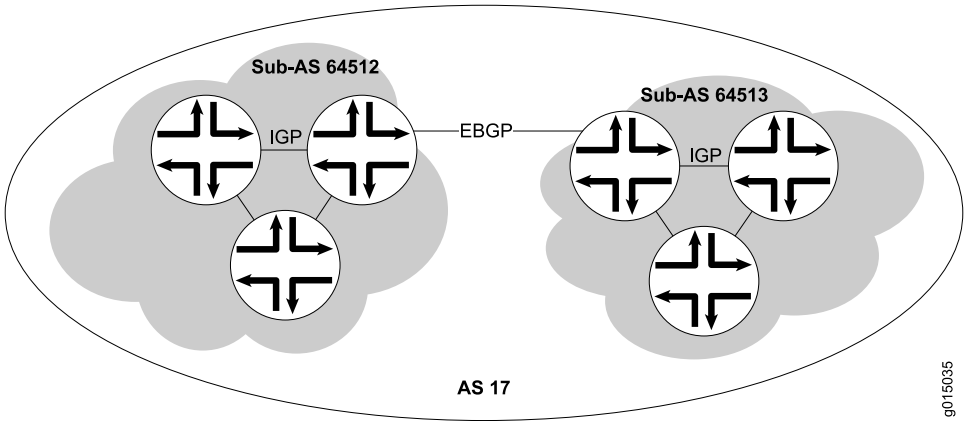
Task	J-Web Configuration Editor	CLI Configuration Editor
Configure a routing policy to advertise BGP routes.	See "Injecting OSPF Routes into the BGP Routing Table (Optional)" on page 510.	

Configuring BGP Confederations (Optional)

To help solve BGP scaling problems caused by the IBGP full-mesh requirement, you can divide your AS into sub-ASs called confederations. As Figure 72 on page 397 shows, the connections between the sub-ASs are made through EBGP sessions, and the internal connections are made through standard IBGP sessions.

In the sample network, AS 17 has two separate confederations (sub-AS 64512 and sub-AS 64513), each of which has multiple routers. Within a sub-AS, an IGP (OSPF, for example) is used to establish network connectivity with internal peers. Between sub-ASs, an external BGP peering session is established.

Figure 72: Typical Network Using BGP Confederations



To configure the BGP confederations shown in Figure 72 on page 397:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 126 on page 398.
3. If you are finished configuring the router, commit the configuration.
4. To check the configuration, see "Verifying a BGP Configuration" on page 398.

Table 126: Configuring BGP Confederations

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Routing options** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing options, click **Edit**.	From the [edit] hierarchy level, enter edit routing-options
Set the AS number to the sub-AS number 64512. The sub-AS number is a unique AS number that is usually taken from the pool of private AS numbers—64512 through 65535.	1. In the AS Number box, enter the sub-AS number. 2. Click **OK**.	Set the sub-AS number: set autonomous-system 64512
Navigate to the **Confederation** level in the configuration hierarchy.	1. On the main Configuration page next to Routing options, click **Edit**. 2. Next to Confederation, click **Configure**.	From the [edit] hierarchy level, enter edit routing-options confederation
Set the confederation number to the AS number 17.	In the Confederation as box, enter 17.	Set the confederation AS number: set 17
Add the sub-ASs as members of the confederation. Every sub-AS within the AS must be added as a confederation member.	1. Next to Members, click **Add new entry**. 2. In the Value box, enter the sub-ASs that are members of this confederation. Separate multiple sub-ASs with a space.	Add members to the confederation: set 17 members 64512 64513
Using EBGP, configure the peering session between the confederations (from Router A to Router B in this example). When setting the peer AS number for these sessions, use the sub-AS number rather than the AS number.	See "Configuring Point-to-Point Peering Sessions (Required)" on page 390.	
Using IBGP, configure internal sessions within a sub-AS. You can configure an IBGP full mesh, or you can configure a route reflector.	■ To configure an IBGP full mesh, see "Configuring BGP Within a Network (Required)" on page 393. ■ To configure a route reflector, see "Configuring a Route Reflector (Optional)" on page 394.	

Verifying a BGP Configuration

To verify a BGP configuration, perform these tasks:

- Verifying BGP Neighbors on page 399
- Verifying BGP Groups on page 400

- Verifying BGP Summary Information on page 400
- Verifying Reachability of All Peers in a BGP Network on page 401

Verifying BGP Neighbors

Purpose Verify that BGP is running on configured interfaces and that the BGP session is active for each neighbor address.

Action From the CLI, enter the show bgp neighbor command.

Sample Output

```
user@host> show bgp neighbor
Peer: 10.255.245.12+179 AS 35  Local: 10.255.245.13+2884 AS 35
  Type: Internal    State: Established  (route reflector client)Flags: Sync
  Last State: OpenConfirm   Last Event: RecvKeepAlive
  Last Error: None
  Options: Preference LocalAddress HoldTime Cluster AddressFamily Rib-group Refresh

  Address families configured: inet-vpn-unicast inet-labeled-unicast
  Local Address: 10.255.245.13 Holdtime: 90 Preference: 170
  Flags for NLRI inet-vpn-unicast: AggregateLabel
  Flags for NLRI inet-labeled-unicast: AggregateLabel
  Number of flaps: 0
  Peer ID: 10.255.245.12    Local ID: 10.255.245.13    Active Holdtime: 90
  Keepalive Interval: 30
  NLRI advertised by peer: inet-vpn-unicast inet-labeled-unicast
  NLRI for this session: inet-vpn-unicast inet-labeled-unicast
  Peer supports Refresh capability (2)
Restart time configured on the peer: 300
  Stale routes from peer are kept for: 60
  Restart time requested by this peer: 300
  NLRI that peer supports restart for: inet-unicast inet6-unicast
  NLRI that restart is negotiated for: inet-unicast inet6-unicast
  NLRI of received end-of-rib markers: inet-unicast inet6-unicast
  NLRI of all end-of-rib markers sent: inet-unicast inet6-unicast
  Table inet.0 Bit: 10000
    RIB State: restart is complete
    Send state: in sync
    Active prefixes: 4
    Received prefixes: 6
    Suppressed due to damping: 0
  Table inet6.0 Bit: 20000
    RIB State: restart is complete
    Send state: in sync
    Active prefixes: 0
    Received prefixes: 2
    Suppressed due to damping: 0
  Last traffic (seconds): Received 3    Sent 3    Checked 3
  Input messages:  Total 9      Updates 6       Refreshes 0     Octets 403
  Output messages: Total 7      Updates 3       Refreshes 0     Octets 365
  Output Queue[0]: 0
  Output Queue[1]: 0
  Trace options: detail packets
  Trace file: /var/log/bgpgr size 131072 files 10
```

Meaning The output shows a list of the BGP neighbors with detailed session information. Verify the following information:

- Each configured peering neighbor is listed.
- For State, each BGP session is Established.

- For Type, each peer is configured as the correct type (either internal or external).
- For AS, the AS number of the BGP neighbor is correct.

Related Topics For a complete description of show bgp neighbor output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying BGP Groups

Purpose Verify that the BGP groups are configured correctly.

Action From the CLI, enter the show bgp group command.

Sample Output

```
user@host> show bgp group
Group Type: Internal    AS: 10045       Local AS: 10045
  Name: pe-to-asbr2                                 Flags: Export Eval
  Export: [ match-all ]
  Total peers: 1        Established: 1
  10.0.0.4+179
  bgp.l3vpn.0: 1/1/0
  vpn-green.inet.0: 1/1/0

Groups: 1   Peers: 1    External: 0    Internal: 1    Down peers: 0   Flaps: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
bgp.l3vpn.0            1          1          0          0          0          0
```

Meaning The output shows a list of the BGP groups with detailed group information. Verify the following information:

- Each configured group is listed.
- For AS, each group's remote AS is configured correctly.
- For Local AS, each group's local AS is configured correctly.
- For Group Type, each group has the correct type (either internal or external).
- For Total peers, the expected number of peers within the group is shown.
- For Established, the expected number of peers within the group have BGP sessions in the Established state.
- The IP addresses of all the peers within the group are present.

Related Topics For a complete description of show bgp group output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying BGP Summary Information

Purpose Verify that the BGP configuration is correct.

Action From the CLI, enter the show bgp summary command.

Sample Output

```
user@host> show bgp summary
Groups: 1 Peers: 3 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
inet.0                 6          4          0          0          0          0
Peer               AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn
```

```
State|#Active/Received/Damped...
10.0.0.2          65002      88675      88652       0       2        42:38 2/4/0
                0/0/0
10.0.0.3          65002      54528      54532       0       1     2w4d22h 0/0/0
                0/0/0
10.0.0.4          65002      51597      51584       0       0     2w3d22h 2/2/0
                0/0/0
```

Meaning The output shows a summary of BGP session information. Verify the following information:

- For Groups, the total number of configured groups is shown.
- For Peers, the total number of BGP peers is shown.
- For Down Peers, the total number of unestablished peers is 0. If this value is not zero, one or more peering sessions are not yet established.
- Under Peer, the IP address for each configured peer is shown.
- Under AS, the peer AS for each configured peer is correct.
- Under Up/Dwn State, the BGP state reflects the number of paths received from the neighbor, the number of these paths that have been accepted, and the number of routes being damped (such as 0/0/0). If the field is Active, it indicates a problem in the establishment of the BGP session.

Related Topics For a complete description of show bgp summary output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying Reachability of All Peers in a BGP Network

Purpose By using the ping tool on each peer address in the network, verify that all peers in the network are reachable from each device.

Action For each device in the BGP network:

1. In the J-Web interface, select **Diagnose > Ping Host**.
2. In the Remote Host box, type the name of a host for which you want to verify reachability from the device.
3. Click **Start**. Output appears on a separate page.

Sample Output

```
PING 10.10.10.10 : 56 data bytes
64 bytes from 10.10.10.10: icmp_seq=0 ttl=255 time=0.382 ms
64 bytes from 10.10.10.10: icmp_seq=1 ttl=255 time=0.266 ms
```

Meaning If a host is active, it generates an ICMP response. If this response is received, the round-trip time is listed in the time field.

Related Topics For more information about using the J-Web interface to ping a host, see the *JUNOS Software Administration Guide*.

For information about the ping command, see the *JUNOS Software Administration Guide* or the *JUNOS System Basics and Services Command Reference*.

Part 4

Configuring Private Communications over Public Networks with MPLS

Chapter 18

Multiprotocol Label Switching Overview

Multiprotocol Label Switching (MPLS) provides a framework for controlling traffic patterns across a network. The MPLS framework allows Services Routers to pass traffic through transit networks on paths that are independent of the individual routing protocols enabled throughout the network.

The MPLS framework supports traffic engineering and the creation of virtual private networks (VPNs). Traffic is engineered (controlled) primarily by the use of signaling protocols to establish label-switched paths (LSPs). VPN support includes Layer 2 and Layer 3 VPNs and Layer 2 circuits.

When you first install JUNOS software on your Services Router, MPLS is disabled by default. After you enable your router to allow MPLS traffic, the router switches to packet-based processing and operates as described in *JUNOS Software Security Configuration Guide*

CAUTION: When MPLS is enabled on your router, all security features such as security policies, zones, NAT, ALGs, chassis clustering, screens, firewall authentication, and IPsec VPNs are unavailable. For more information on the data path for security features, see *JUNOS Software Security Configuration Guide*.

This chapter contains the following topics. For more information, see the *JUNOS MPLS Applications Configuration Guide*, and *JUNOS VPNs Configuration Guide*.

MPLS and VPN Terms

To understand MPLS and VPNs, become familiar with the terms defined in Table 127 on page 406.

Table 127: MPLS and VPN Terms

Term	Definition
color	See *link coloring*.
Constrained Shortest Path First (CSPF)	MPLS algorithm that has been modified to include specific restrictions for calculating the shortest path across the network.
customer edge (CE) router	Services Router or switch in the customer's network that is connected to a service provider's provider edge (PE) router and participates in a Layer 3 VPN.
Explicit Route Object (ERO)	Extension to the Resource Reservation Protocol (RSVP) that allows an RSVP PATH message to traverse an explicit sequence of routers independently of conventional shortest-path IP routing.
inbound router	Entry point for a label-switched path (LSP). Each LSP must have exactly one inbound router that is different from the outbound router. Inbound routers are also known as ingress routers. See also *outbound router*.
label	In Multiprotocol Label Switching (MPLS), a 20-bit unsigned integer in the range 0 through 1,048,575, used to identify a packet traveling along a label-switched path (LSP).
Label Distribution Protocol (LDP)	Protocol for distributing labels in non-traffic-engineered applications. LDP allows Services Routers to establish label-switched paths (LSPs) through a network by mapping Network layer routing information directly to Data Link Layer switched paths.
label-switched path (LSP)	Sequence of Services Routers that cooperatively perform Multiprotocol Label Switching (MPLS) operations for a packet stream. The first router in an LSP is called the inbound router, and the last router in the path is called the outbound router. An LSP is a point-to-point, half-duplex connection from the inbound router to the outbound router. (The inbound and outbound routers cannot be the same router.)
label-switching router (LSR)	Any Services Router that is part of an LSP.
Layer 2 circuit	Point-to-point Layer 2 connection transported by means of Multiprotocol Label Switching (MPLS) or another tunneling technology on a service provider's network. Multiple Layer 2 circuits can be transported over a single label-switched path (LSP) tunnel between two provider edge (PE) routers.
Layer 2 VPN	Private network service among a set of customer sites that use a service provider's existing Multiprotocol Label Switching (MPLS) and IP network. One customer's data is separated from another's by software rather than hardware. In a Layer 2 VPN, the Layer 3 routing of customer traffic occurs within the *customer* network.
Layer 3 VPN	Private network service among a set of customer sites that use a service provider's existing Multiprotocol Label Switching (MPLS) and IP network. One customer's routes and data are separated from another customer's routes and data by software rather than hardware. In a Layer 3 VPN, the Layer 3 routing of customer traffic occurs within the *service provider* network.
link coloring	In Constrained Shortest Path First (CSPF) routing, a way to group Multiprotocol Label Switching (MPLS) interfaces for CSPF path selection by assigning a color identifier and number to each administrative group.
Multiprotocol Label Switching (MPLS)	Method for engineering network traffic patterns by assigning short labels to network packets that describe how to forward the packets through the network.
multiple push	Addition by a Services Router of up to three labels to a packet as it enters a Multiprotocol Label Switching (MPLS) domain.

Table 127: MPLS and VPN Terms *(continued)*

Term	Definition
outbound router	Exit point for a label-switched path (LSP). Each LSP must have exactly one outbound router that is different from the inbound router. Outbound routers are also called egress routers. See also *inbound router*.
penultimate hop popping (PHP)	Using the penultimate router rather than the outbound router in a label-switched path (LSP) to remove the Multiprotocol Label Switching (MPLS) label from a packet.
penultimate router	Second-to-last Services Router in an LSP. The penultimate router is responsible for label popping when penultimate hop popping (PHP) is configured.
point-to-multipoint LSP	Label-switched path (LSP) that allows a network operator to use MPLS for point-to-multipoint data distribution in an efficient manner. Point-to-multipoint LSPs add IP multicast functionality to MPLS.
pop	Removal by a Services Router of the top label from a packet as it exits the Multiprotocol Label Switching (MPLS) domain.
provider edge (PE) router	Services Router in the service provider network that is connected to a customer edge (CE) router and participates in a virtual private network (VPN).
provider router	Services Router in the service provider's network that does not attach to a customer edge (CE) router.
push	Addition of a label or stack of labels by a Services Router to a packet as it enters a Multiprotocol Label Switching (MPLS) domain.
Resource Reservation Protocol (RSVP)	Resource reservation setup protocol that interacts with integrated services on the Internet.
route distinguisher	A 6-byte virtual private network (VPN) identifier that is prefixed to an IPv4 address to make it unique. The new address is part of the VPN-IPv4 address family, which is a Border Gateway Protocol (BGP) extension. A route distinguisher allows you to configure private addresses within the VPN by preventing any overlap with the private addresses in other VPNs.
routing instance	Collection of routing tables, their interfaces, and the routing protocol parameters that control the information they contain.
swap	Replacement by a Services Router of a label or stack of labels on a packet as it travels through a Multiprotocol Label Switching (MPLS) domain.
swap and push	Replacement and subsequent push by a Services Router of a label or stack of labels on a packet as it travels through a Multiprotocol Label Switching (MPLS) domain.
Traffic engineering (TE)	The techniques and processes used to cause routed traffic to travel through the network on a path other than the one that would have been chosen if standard routing methods had been used.
traffic engineering database (TED)	Database populated by label-switched path (LSP) information such as the network topology, current reservable bandwidth of links, and link colors. The traffic engineering database is used to determine Constrained Shortest Path First (CSPF) path selection.
transit router	Any label-switching router (LSR) between the inbound and outbound Services Router of a label-switched path (LSP).

Table 127: MPLS and VPN Terms *(continued)*

Term	Definition
virtual private network (VPN)	Private data network that uses a public TCP/IP network, typically the Internet, while maintaining privacy with a tunneling protocol, encryption, and security procedures.
VPN routing and forwarding (VRF) instance	Routing instance for a Layer 3 VPN implementation that consists of one or more routing tables, a derived forwarding table, the interfaces that use the forwarding table, and the policies and routing protocols that determine what goes into the forwarding table.

MPLS Overview

Multiprotocol Label Switching (MPLS) is a method for engineering traffic patterns by assigning short labels to network packets that describe how to forward them through the network. MPLS is independent of routing tables or any routing protocol and can be used for unicast packets.

This overview contains the following topics:

Label Switching

In a traditional IP network, packets are transmitted with an IP header that includes a source and destination address. When a router receives such a packet, it examines its forwarding tables for the next-hop address associated with the packet's destination address and forwards the packet to the next-hop location.

In an MPLS network, each packet is encapsulated with an MPLS header. When a router receives the packet, it copies the header as an index into a separate MPLS forwarding table. The MPLS forwarding table consists of pairs of inbound interfaces and path information. Each pair includes forwarding information that the router uses to forward the traffic and modify, when necessary, the MPLS header.

Because the MPLS forwarding table has far fewer entries than the more general forwarding table, the lookup consumes less processing time and processing power. The resultant savings in time and processing are a significant benefit for traffic that uses the network to transit between outside destinations only.

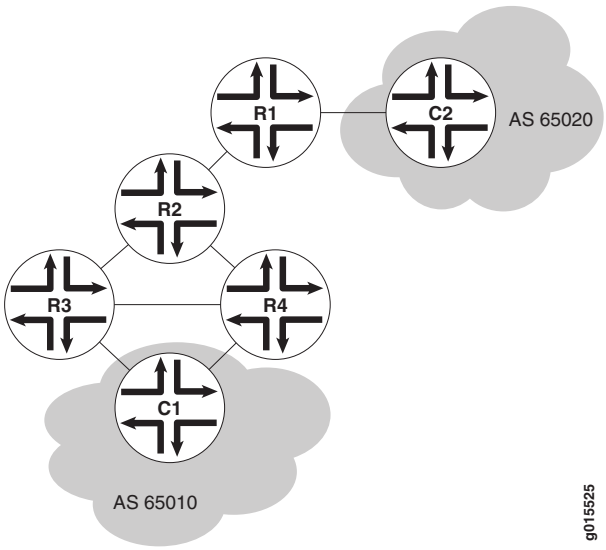
Label-Switched Paths

Label-switched paths (LSPs) are unidirectional routes through a network or autonomous system (AS). In normal IP routing, the packet has no predetermined path. Instead, each router forwards a packet to the next-hop address stored in its forwarding table, based only on the packet's destination address. Each subsequent router then forwards the packet using its own forwarding table.

In contrast, MPLS routers within an AS determine paths through a network through the exchange of MPLS traffic engineering information. Using these paths, the routers direct traffic through the network along an established route. Rather than selecting the next hop along the path as in IP routing, each router is responsible for forwarding the packet to a predetermined next-hop address.

Figure 73 on page 409 shows a typical LSP topology.

Figure 73: Typical LSP Topology



In the topology shown in Figure 73 on page 409, traffic is forwarded from Host C1 to the transit network with standard IP forwarding. When the traffic enters the transit network, it is switched across a preestablished LSP through the network. In this example, an LSP might switch the traffic from Router R4 to Router R2 to Router R1. When the traffic exits the network, it is forwarded to its destination by IP routing protocols.

Label-Switching Routers

Routers that are part of the LSP are label-switching routers (LSRs). Each LSR must be configured with MPLS so that it can interpret MPLS headers and perform the MPLS operations required to pass traffic through the network. An LSP can include four types of LSRs:

- Inbound router—The only entry point for traffic into MPLS. Native IPv4 packets are encapsulated into the MPLS protocol by the inbound router. Each LSP can have only one inbound router.
- Transit router—Any router in the middle of an LSP. An individual LSP can contain between 0 and 253 transit routers. Transit routers forward MPLS traffic along the LSP, using only the MPLS header to determine how the packet is routed.
- Penultimate router—The second-to-last router in the LSP. The penultimate router in an LSP is responsible for stripping the MPLS header from the packet before forwarding it to the outbound router.
- Outbound router—The endpoint for the LSP. The outbound router receives MPLS packets from the penultimate router and performs an IP route lookup. The router then forwards the packet to the next hop of the route. Each LSP can have only one outbound router.

Labels

To forward traffic through an MPLS network, MPLS routers encapsulate packets and assign and manage headers known as labels. The routers use the labels to index the MPLS forwarding tables that determine how packets are routed through the network.

When a network's inbound router receives traffic, it inserts an MPLS label between the IP packet and the appropriate Layer 2 header for the physical link. The label contains an index value that identifies a next-hop address for the particular LSP. When the next-hop transit router receives the packet, it uses the index in the MPLS label to determine the next-hop address for the packet and forwards the packet to the next router in the LSP.

As each packet travels through the transit network, every router along the way performs a lookup on the MPLS label and forwards the packet accordingly. When the outbound router receives a packet, it examines the header to determine that it is the final router in the LSP. The outbound router then removes the MPLS header, performs a regular IP route lookup, and forwards the packet with its IP header to the next-hop address.

Label Operations

Each LSR along an LSP is responsible for examining the MPLS label, determining the LSP next hop, and performing the required label operations. LSRs can perform five label operations:

- Push—Adds a new label to the top of the packet. For IPv4 packets arriving at the inbound router, the new label is the first label in the label stack. For MPLS packets with an existing label, this operation adds a label to the stack and sets the stacking bit to 0, indicating that more MPLS labels follow the first.

  When it receives the packet, the inbound router performs an IP route lookup on the packet. Because the route lookup yields an LSP next hop, the inbound router performs a label push on the packet, and then forwards the packet to the LSP next hop.
- Swap—Replaces the label at the top of the label stack with a new label.

When a transit router receives the packet, it performs an MPLS forwarding table lookup. The lookup yields the LSP next hop and the path index of the link between the transit router and the next router in the LSP.

- Pop—Removes the label from the top of the label stack. For IPv4 packets arriving at the penultimate router, the entire MPLS label is removed from the label stack. For MPLS packets with an existing label, this operation removes the top label from the label stack and modifies the stacking bit as necessary—sets it to 1, for example, if only a single label remains in the stack.

  If multiple LSPs terminate at the same outbound router, the router performs MPLS label operations for all outbound traffic on the LSPs. To share the operations among multiple routers, most LSPs use penultimate hop popping (PHP).

- Multiple push—Adds multiple labels to the top of the label stack. This action is equivalent to performing multiple push operations.

  The multiple push operation is used with label stacking, which is beyond the scope of this guide.

- Swap and push—Replaces the top label with a new label and then pushes a new label to the top of the stack.

  The swap and push operation is used with label stacking, which is beyond the scope of this guide.

Penultimate Hop Popping

Multiple LSPs terminating at a single outbound router load the router with MPLS label operations for all their outbound traffic. Penultimate hop popping (PHP) transfers the operation from the outbound router to penultimate routers.

With PHP, the penultimate router is responsible for popping the MPLS label and forwarding the traffic to the outbound router. The outbound router then performs an IP route lookup and forwards the traffic. For example, if four LSPs terminate at the same outbound router and each has a different penultimate router, label operations are shared across four routers.

LSP Establishment

An MPLS LSP is established by one of two methods: static LSPs and dynamic LSPs.

Static LSPs

Like a static route, a static LSP requires each router along the path to be configured explicitly. You must manually configure the path and its associated label values. Static LSPs require less processing by the LSRs because no signaling protocol is used. However, because paths are statically configured, they cannot adapt to network conditions. Topology changes and network outages can create black holes in the LSP that exist until you manually reconfigure the LSP.

Dynamic LSPs

Dynamic LSPs use signaling protocols to establish themselves and propagate LSP information to other LSRs in the network. You configure the inbound router with LSP information that is transmitted throughout the network when you enable the signaling protocols across the LSRs. Because the LSRs must exchange and process signaling packets and instructions, dynamic LSPs consume more resources than static LSPs. However, dynamic LSPs can avoid the network black holes of static LSPs by detecting topology changes and outages and propagating them throughout the network.

Traffic Engineering with MPLS

Traffic engineering facilitates efficient and reliable network operations while simultaneously optimizing network resources and traffic performance. Traffic engineering provides the ability to move traffic flow away from the shortest path selected by the interior gateway protocol (IGP) to a potentially less congested physical path across a network. To support traffic engineering, besides source routing, the network must do the following:

- Compute a path at the source by taking into account all the constraints, such as bandwidth and administrative requirements.
- Distribute the information about network topology and link attributes throughout the network once the path is computed.
- Reserve network resources and modify link attributes.

MPLS traffic engineering uses the following components:

- MPLS LSPs for packet forwarding
- IGP extensions for distributing information about the network topology and link attributes
- CSPF for path computation and path selection
- RSVP extensions to establish the forwarding state along the path and reserve resources along the path

J-series Services Routers also support traffic engineering across different OSPF regions. For more details, see the *JUNOS MPLS Applications Configuration Guide*.

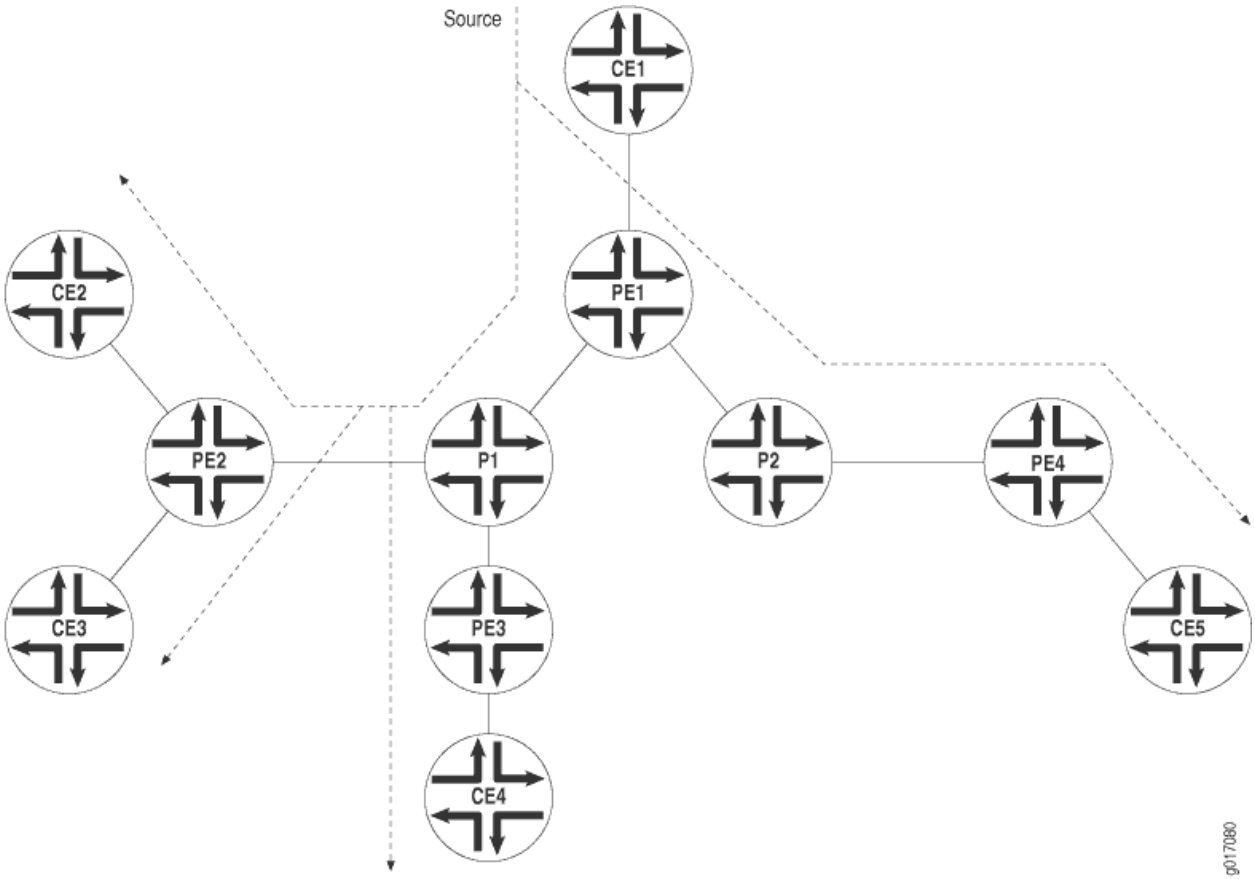
Point-to-Multipoint LSPs

A point-to-multipoint MPLS LSP is an RSVP-signaled LSP with a single source and multiple destinations. By taking advantage of the MPLS packet replication capability of the network, point-to-multipoint LSPs avoid unnecessary packet replication at the inbound (ingress) router. Packet replication takes place only when packets are forwarded to two or more different destinations requiring different network paths.

This process is illustrated in Figure 74 on page 413. Router PE1 is configured with a point-to-multipoint LSP to Routers PE2, PE3, and PE4. When Router PE1 sends a packet on the point-to-multipoint LSP to Routers P1 and P2, Router P1 replicates the

packet and forwards it to Routers PE2 and PE3. Router P2 sends the packet to Router PE4.

Figure 74: Point-to-Multipoint LSPs



Point-to-Multipoint LSP Properties

The following are some of the properties of point-to-multipoint LSPs:

- A point-to-multipoint LSP allows you to use MPLS for point-to-multipoint data distribution. This functionality is similar to that provided by IP multicast.
- You can add and remove branch LSPs from a main point-to-multipoint LSP without disrupting traffic. The unaffected parts of the point-to-multipoint LSP continue to function normally.
- You can configure a node to be both a transit and an outbound (egress) router for different branch LSPs of the same point-to-multipoint LSP.
- You can enable link protection on a point-to-multipoint LSP. Link protection can provide a bypass LSP for each of the branch LSPs that make up the point-to-multipoint LSP. If any of the primary paths fails, traffic can be quickly switched to the bypass.

- You can configure sub-paths either statically or dynamically.
- You can enable graceful restart on point-to-multipoint LSPs.

Point-to-Multipoint LSP Configuration

To set up a point-to-multipoint LSP, you configure the primary LSP from the ingress router and the branch LSPs that carry traffic to the egress routers. In addition to the conventional LSP configuration, you specify a path name on the primary LSP and this same path name on each branch LSP.

By default, the branch LSPs are dynamically signaled by means of CSPF and require no configuration. You can alternatively configure the branch LSPs as a static path.

For more information and configuration instructions, see the *JUNOS MPLS Applications Configuration Guide*.

Signaling Protocols Overview

Two MPLS signaling protocols are used to dynamically establish and maintain LSPs within a network:

- Label Distribution Protocol on page 414
- Resource Reservation Protocol on page 415

Label Distribution Protocol

LDP is a simple, fast-acting signaling protocol that automatically establishes LSP adjacencies within an MPLS network. Routers then share LSP updates such as hello packets and LSP advertisements across the adjacencies.

LDP Operation

Because LDP runs on top of an interior gateway protocol (IGP) such as IS-IS or OSPF, you must configure LDP and the IGP on the same set of interfaces. After both are configured, LDP begins transmitting and receiving LDP messages through all LDP-enabled interfaces.

Because of LDP's simplicity, it cannot perform true traffic engineering like RSVP. LDP does not support bandwidth reservation or traffic constraints.

LDP Messages

When you configure LDP on an LSR, the router begins sending LDP discovery messages out all LDP-enabled interfaces. When an adjacent LSR receives LDP discovery messages, it establishes an underlying TCP session. An LDP session is then created on top of the TCP session. The TCP three-way handshake ensures that the LDP session has bidirectional connectivity. After they establish the LDP session, the LDP neighbors maintain, and terminate, the session by exchanging messages.

LDP advertisement messages allow LSRs to exchange label information to determine the next hops within a particular LSP.

Any topology changes, such as a router failure, generate LDP notifications that can terminate the LDP session or generate additional LDP advertisements to propagate an LSP change.

Resource Reservation Protocol

Resource Reservation Protocol (RSVP) is a signaling protocol that handles bandwidth allocation and true traffic engineering across an MPLS network. Like LDP, RSVP uses discovery messages and advertisements to exchange LSP path information between all hosts. However, RSVP also includes a set of features that control the flow of traffic through an MPLS network.

This section contains the following topics:

RSVP Fundamentals

RSVP uses unidirectional and simplex flows through the network to perform its function. The inbound router initiates an RSVP path message and sends it downstream to the outbound router. The path message contains information about the resources needed for the connection. Each router along the path begins to maintain information about a reservation.

When the path message reaches the outbound router, resource reservation begins. The outbound router sends a reservation message upstream to the inbound router. Each router along the path receives the reservation message and sends it upstream, following the path of the original path message. When the inbound router receives the reservation message, the unidirectional network path is established.

The established path remains open as long as the RSVP session is active. The session is maintained by the transmission of additional path and reservation messages that report the session state every 30 seconds. If a router does not receive the maintenance messages for 3 minutes, it terminates the RSVP session and reroutes the LSP through another active router.

Bandwidth Reservation Requirement

When a bandwidth reservation is configured, reservation messages propagate the bandwidth value throughout the LSP. Routers must reserve the bandwidth specified across the link for the particular LSP. If the total bandwidth reservation exceeds the available bandwidth for a particular LSP segment, the LSP is rerouted through another LSR. If no segments can support the bandwidth reservation, LSP setup fails and the RSVP session is not established.

Explicit Route Objects

Explicit Route Objects (EROs) limit LSP routing to a specified list of LSRs. By default, RSVP messages follow a path that is determined by the network IGP's shortest path. However, in the presence of a configured ERO, the RSVP messages follow the path specified.

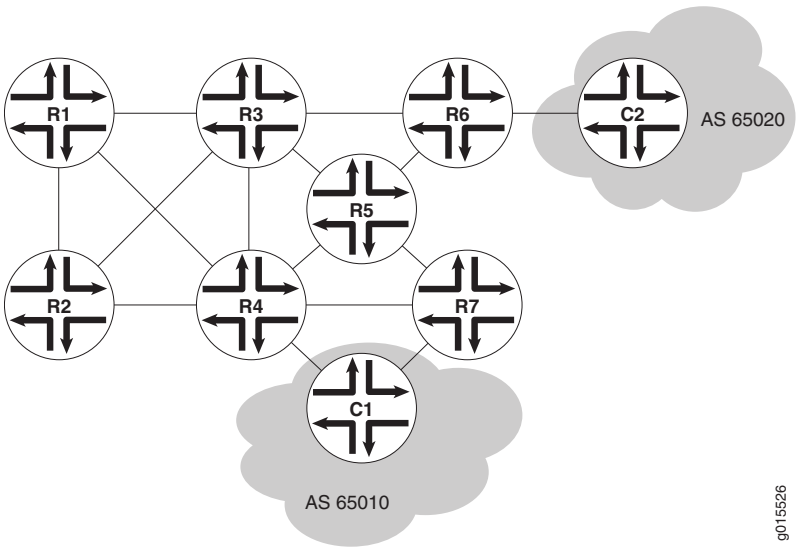
EROs consist of two types of instructions: loose hops and strict hops. When a loose hop is configured, it identifies one or more transit LSRs through which the LSP must be routed. The network IGP determines the exact route from the inbound router to the first loose hop, or from one loose hop to the next. The loose hop specifies only that a particular LSR be included in the LSP.

When a strict hop is configured, it identifies an exact path through which the LSP must be routed. Strict-hop EROs specify the exact order of routers through which the RSVP messages are sent.

You can configure loose-hop and strict-hop EROs simultaneously. In this case, the IGP determines the route between loose hops, and the strict-hop configuration specifies the exact path for particular LSP path segments.

Figure 75 on page 416 shows a typical RSVP-signaled LSP that uses EROs.

Figure 75: Typical RSVP-Signaled LSP with EROs



In the topology shown in Figure 75 on page 416, traffic is routed from Host C1 to Host C2. The LSP can pass through Router R4 or Router R7. To force the LSP to use R4, you can set up either a loose-hop or strict-hop ERO that specifies R4 as a hop in the LSP. To force a specific path through Router R4, R3, and R6, configure a strict-hop ERO through the three LSRs.

Constrained Shortest Path First

Whereas IGPs use the Shortest Path First (SPF) algorithm to determine how traffic is routed within a network, RSVP uses the Constrained Shortest Path First (CSPF) algorithm to calculate traffic paths that are subject to the following constraints:

- LSP attributes—Administrative groups such as link coloring, bandwidth requirements, and EROs
- Link attributes—Colors on a particular link and available bandwidth

These constraints are maintained in the traffic engineering database (TED). The database provides CSPF with up-to-date topology information, the current reservable bandwidth of links, and the link colors.

In determining which path to select, CSPF follows these rules:

1. Computes LSPs one at a time, beginning with the highest-priority LSP—the one with the lowest setup priority value. Among LSPs of equal priority, CSPF starts with those that have the highest bandwidth requirement.
2. Prunes the traffic engineering database of links that are not full duplex and do not have sufficient reservable bandwidth.
3. If the LSP configuration includes the include statement, prunes all links that do not share any included colors.
4. If the LSP configuration includes the exclude statement, prunes all links that contain excluded colors. If a link does not have a color, it is accepted.
5. Finds the shortest path toward the LSP's outbound router, taking into account any EROs. For example, if the path must pass through Router A, two separate SPF algorithms are computed: one from the inbound router to Router A and one from Router A to the outbound router.
6. If several paths have equal cost, chooses the one with a last-hop address the same as the LSP's destination.
7. If several equal-cost paths remain, selects the path with the fewest number of hops.
8. If several equal-cost paths remain, applies CSPF load-balancing rules configured on the LSP.

Link Coloring

RSVP allows you to configure administrative groups for CSPF path selection. An administrative group is typically named with a color, assigned a numeric value, and applied to the RSVP interface for the appropriate link. Lower numbers indicate higher priority.

After configuring the administrative group, you can either exclude, include, or ignore links of that color in the traffic engineering database:

- If you exclude a particular color, all segments with an administrative group of that color are excluded from CSPF path selection.

- If you include a particular color, only those segments with the appropriate color are selected.
- If you neither exclude nor include the color, the metrics associated with the administrative groups and applied on the particular segments determine the path cost for that segment.

The LSP with the lowest total path cost is selected into the traffic engineering database.

VPN Overview

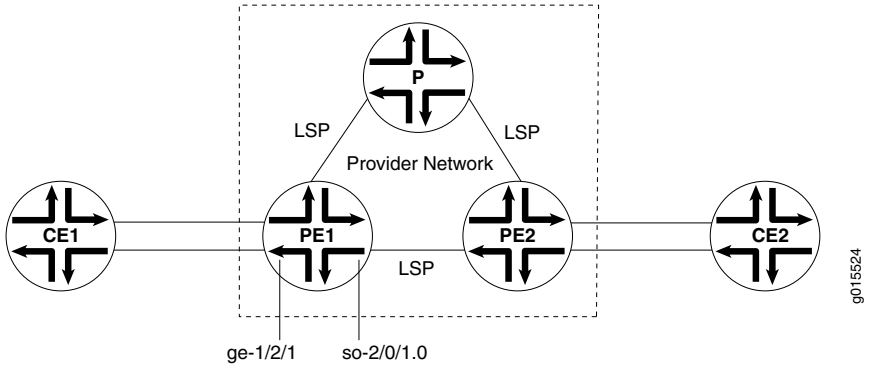
Virtual private networks (VPNs) are private networks that use a public network to connect two or more remote sites. In place of dedicated connections between networks, VPNs use virtual connections routed (tunneled) through public networks that are typically service provider networks. The type of the VPN is determined by the connections it uses and whether the customer network or the provider network performs the virtual tunneling.

This overview contains the following topics:

VPN Components

All types of VPNs share certain components. Figure 76 on page 418 shows a typical VPN topology.

Figure 76: Typical VPN Topology



The provider edge (PE) routers in the provider's network connect to the customer edge (CE) routers located at customer sites. PE routers support VPN and MPLS label functionality. Within a single VPN, pairs of PE routers are connected through a virtual tunnel, typically an LSP.

Provider routers within the core of the provider's network are not connected to any routers at a customer site but are part of the tunnel between pairs of PE routers. Provider routers support LSP functionality as part of the tunnel support, but do not support VPN functionality.

Customer edge (CE) routers are the routers or switches located at the customer site that connect to the provider's network. CE routers are typically IP routers, but they can also be Asynchronous Transfer Mode (ATM), Frame Relay, or Ethernet switches.

All VPN functions are performed by the PE routers. Neither CE routers nor provider routers are required to perform any VPN functions.

VPN Routing Requirements

VPNs tunnel traffic as follows from one customer site to another customer site, using a public network as a transit network, when certain requirements are met:

1. Traffic is forwarded by standard IP forwarding from the CE routers to the PE routers.

   The CE routers require only a BGP connection to the PE routers.

2. The PE routers establish an LSP through the provider network.

   The provider network must be running either OSPF or IS-IS as an IGP, as well as IBGP sessions through either a full mesh or route reflector. IBGP is required so that the PE routers can exchange route information for routes that originate or terminate in the VPN.

3. When the inbound PE router receives traffic, it performs a route lookup. The lookup yields an LSP next hop, and the traffic is forwarded along the LSP.

   Either LDP or RSVP must be configured to dynamically set up LSPs through the provider network.

4. When the traffic reaches the outbound PE router, the PE router pops the MPLS label and forwards the traffic with standard IP routing.

   Because the tunnel information is maintained at both PE routers, neither the provider core routers nor the CE routers need to maintain any VPN information in their configuration databases.

VPN Routing Information

Routing information, including routes, route distinguishers, and routing policies, is stored in a VPN routing and forwarding (VRF) table. Routers must maintain separate VRF tables for each VPN.

VRF Instances

A routing instance is a collection of routing tables, interfaces, and routing protocol parameters. The interfaces belong to the routing tables, and the routing protocol parameters control the information in the routing tables. In the case of VPNs, each VPN has a VPN routing and forwarding (VRF) instance.

A VRF instance consists of one or more routing tables, a derived forwarding table, the interfaces that use the forwarding table, and the policies and routing protocols that determine what goes into the forwarding table. Because each instance is configured for a particular VPN, each VPN has separate tables, rules, and policies that control its operation.

A separate VRF table is created for each VPN that has a connection to a CE router. The VRF table is populated with routes received from directly connected CE sites associated with the VRF instance, and with routes received from other PE routers in the same VPN.

Route Distinguishers

Because a typical transit network is configured to handle more than one VPN, the provider routers are likely to have multiple VRF instances configured. As a result, depending on the origin of the traffic and any filtering rules applied to the traffic, the BGP routing tables can contain multiple routes for a particular destination address. Because BGP requires that exactly one BGP route per destination be imported into the forwarding table, BGP must have a way to distinguish between potentially identical network layer reachability information (NLRI) messages received from different VPNs.

A route distinguisher is a locally unique number that identifies all route information for a particular VPN. Unique numeric identifiers allow BGP to distinguish between routes that are otherwise identical.

Route Targets to Control the VRF Table

On each PE router, you must define routing policies that specify how routes are imported into and exported from the router's VRF table. Each advertisement must have an associated route target that uniquely identifies the VPN for which the advertisement is valid. The route target allows you to keep routing and signaling information for each VPN separate.

Types of VPNs

There are three primary types of VPNs: Layer 2 VPNs, Layer 2 circuits, and Layer 3 VPNs.

Layer 2 VPNs

In a Layer 2 VPN, traffic is forwarded to the PE router in Layer 2 format, carried by MPLS through an LSP over the service provider network, and then converted back to Layer 2 format at the receiving CE router.

On a Layer 2 VPN, routing occurs on the customer routers, typically on the CE router. The CE router connected to a service provider on a Layer 2 VPN must select the appropriate circuit on which to send traffic. The PE router receiving the traffic sends it across the network to the PE router on the outbound side. The PE routers need no information about the customer's routes or routing topology, and need only to determine the virtual tunnel through which to send the traffic.

Layer 2 Circuits

A Layer 2 circuit is a point-to-point Layer 2 connection that transports traffic by MPLS or another tunneling technology on a service provider network. The Layer 2 circuit creates a virtual connection to direct traffic between two CE routers. The primary difference between a Layer 2 circuit and an Layer 2 VPN is the method of setting up the virtual connection. Like a leased line, a Layer 2 circuit forwards all packets received from the local interface to the remote interface.

Layer 3 VPNs

In a Layer 3 VPN, routing occurs on the service provider's routers. As a result, Layer 3 VPNs require information about customer routes and a more extensive VRF policy configuration to share and filter routes that originate or terminate in the VPN.

Because Layer 3 VPNs require the provider routers to route and forward VPN traffic at the entry and exit points of the transit network, the routes must be advertised and filtered throughout the provider network.

Route advertisements originate at the CE routers and are shared with the inbound PE routers through standard IP routing protocols, typically BGP. Based on the source address, the PE router filters route advertisements and imports them into the appropriate VRF table.

The PE router then exports the route in IBGP sessions to the other provider routers. Route export is governed by any routing policy that has been applied to the particular VRF table. To propagate the routes through the provider network, the PE router must also convert the route to VPN format, which includes the route distinguisher.

When the outbound PE router receives the route, it strips off the route distinguisher and advertises the route to the connected CE router, typically through standard BGP IPv4 route advertisements.

Chapter 19

Enabling MPLS

When you first install JUNOS software on your device, MPLS is disabled by default. You must explicitly configure your device to allow MPLS traffic to pass through.

After you enable your router to allow MPLS traffic, the router performs packet-based processing and functions as a standard JUNOS router. For a list of packet-based features available on the router, see the product overview section in the *JUNOS Software Security Configuration Guide*.

CAUTION: When MPLS is enabled, all flow-based security features are deactivated and the router performs packet-based processing. Flow-based services such as security policies, zones, NAT, ALGs, chassis clustering, screens, firewall authentication, and IPSec VPNs are unavailable on the router. For more information on flow-based and packet-based processing, see the *JUNOS Software Security Configuration Guide*.

This chapter contains the following topics:

Deleting Security Services

Before you enable MPLS, we recommend you delete all configured security services. To delete the configured services in the security hierarchy, complete the following tasks:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the tasks described in Table 128 on page 424.
3. Go on to "Enabling MPLS on the Router" on page 424.

CAUTION: Do not commit after deleting the security configurations. A commit without any security configurations leaves the router unreachable through the management port.

Table 128: Deleting Security Services

Task	J-Web Configuration Editor	CLI Configuration Editor
Save your current configuration in the var/tmp/ directory with an appropriate filename with the .cfg extension—for example, curfeb08.cfg.	1. In the J-Web interface, select **Configuration > History**. 2. Next to Current, in the Actions column, click **download**. 3. Select **save** and specify the path to save your current configuration—for example, curfeb08.cfg. 4. Click **OK**.	From the [edit] hierarchy level, enter save /var/tmp/curfeb08.cfg
Remove all configurations in the **security** level of the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Security, click **delete**.	From the [edit] hierarchy level, enter delete security
Remove all global group and inherited configurations.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. In the left pane, select **groups > global**. 3. Next to Security on the right panel, click **delete**. 4. Click **OK**.	From the [edit] hierarchy level, enter delete groups global security

Enabling MPLS on the Router

To include a J-series Services Router running JUNOS software in an MPLS network, you must enable the router for MPLS. Perform these tasks on all the Services Routers running JUNOS software.

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Complete the preconfiguration tasks described in "Deleting Security Services" on page 423. You will get a commit failure if you do not complete the tasks described in Table 128 on page 424.
3. Perform the configuration tasks described in Table 129 on page 425.
4. If you are finished configuring the router, commit the configuration.
5. Reboot your router.
6. Go on to "Configuring Signaling Protocols for Traffic Engineering" on page 427.

Table 129: Enabling MPLS

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Security** level of the configuration hierarchy	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Security, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit security
Enable MPLS for packet-based processing.	1. On the main Security Configuration page next to Forwarding Options, click **Configure** or **Edit**. 2. Next to Family, click **Configure** or **Edit**. 3. Next to Mpls, click **Configure** or **Edit**. 4. Next to Mode, select **packet-based**. 5. Click **OK**.	1. From the [edit security] hierarchy level, enter edit forwarding-options 2. Enter set family mpls mode packet-based
Enable the MPLS family on all transit interfaces on the router.	1. On the main Configuration page next to Interfaces, click **edit**. 2. Select the transit interface on which you want to configure MPLS—for example, **ge-1/0/0**. 3. In the Unit table, click the unit number for which you want to enable MPLS—for example, **0**. 4. In the Family area, select the **Mpls** check box. 5. Click **OK**. 6. Repeat Steps 1 through 5 for each transit interface that you want to include in the MPLS network.	1. From the [edit] hierarchy level, enter edit interfaces 2. Add the MPLS family to all transit interfaces. For example: set interfaces ge-1/0/0 unit 0 family mpls 3. Repeat Steps 1 and 2 for each transit interface that you want to include in the MPLS network.
Enable the MPLS process on all MPLS interfaces.	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Mpls, click **Configure** or **Edit**. 3. Next to Interface, click **Add new entry**. 4. In the Interface name box, type all. 5. Click **OK**. 6. Repeat Steps 1 through 5 for each transit interface on the routers in the MPLS network.	1. From the [edit] hierarchy level, enter edit protocols mpls 2. Enter set interface all 3. Repeat Steps 1 and 2 for each transit interface on the routers in the MPLS network.

CAUTION: If you disable MPLS and switch back to using the security services (flow-based processing), we recommend that you restart your router. Management sessions are reset, and transit traffic is interrupted.

Chapter 20

Configuring Signaling Protocols for Traffic Engineering

Signaling protocols are used within a Multiprotocol Label Switching (MPLS) environment to establish label-switched paths (LSPs) for traffic across a transit network.

You can use either the J-Web configuration editor or CLI configuration editor to configure signaling protocols.

This chapter contains the following topics. For more information about MPLS traffic engineering, see the *JUNOS MPLS Applications Configuration Guide*.

Signaling Protocol Overview

When transit traffic is routed through an IP network, MPLS is often used to engineer its passage. Although the exact path through the transit network is of little importance to either the sender or the receiver of the traffic, network administrators often want to route traffic more efficiently between certain source and destination address pairs. By adding a short label with specific routing instructions to each packet, MPLS switches packets from router to router through the network rather than forwarding packets based on next-hop lookups. The resulting routes are called label-switched paths (LSPs). LSPs control the passage of traffic through the network and speed traffic forwarding.

You can create LSPs manually, or through the use of signaling protocols. J-series Services Routers support two signaling protocols—the Label Distribution Protocol (LDP) and the Resource Reservation Protocol (RSVP).

LDP Signaling Protocol

The Label Distribution Protocol (LDP) is a signaling protocol that runs on a device configured for MPLS support. The LDP configuration is added to the existing interior gateway protocol (IGP) configuration and included in the MPLS configuration. To

configure a network to use LDP for LSP establishment, you first enable MPLS on all transit interfaces in the MPLS network and then enable LDP sessions on the interfaces.

The successful configuration of both MPLS and LDP initiates the exchange of TCP packets across the LDP interfaces. The packets establish TCP-based LDP sessions for the exchange of MPLS information within the network. Enabling both MPLS and LDP on the appropriate interfaces is sufficient to establish LSPs.

RSVP Signaling Protocol

The Resource Reservation Protocol (RSVP) is a more flexible and powerful way to engineer traffic through a transit network. Like LDP, RSVP establishes LSPs within an MPLS network when you enable both MPLS and RSVP on the appropriate interfaces. However, whereas LDP is restricted to using the configured IGP's shortest path as the transit path through the network, RSVP uses a combination of the Constrained Shortest Path First (CSPF) algorithm and Explicit Route Objects (EROs) to determine how traffic is routed through the network.

Basic RSVP sessions are established in exactly the same way that LDP sessions are established. By configuring both MPLS and RSVP on the appropriate transit interfaces, you enable the exchange of RSVP packets and the establishment of LSPs. However, RSVP also lets you configure link authentication, explicit LSP paths, and link coloring. For more information about these topics, see the *JUNOS MPLS Applications Configuration Guide*.

Before You Begin

Before you begin configuring signaling protocols for traffic engineering, complete the following tasks:

- Establish basic connectivity. See the Getting Started Guide for your device.
- Configure network interfaces. See “Configuring Ethernet, DS1, DS3, and Serial Interfaces” on page 73.
- Configure an interior gateway protocol (IGP) across your network. See “Configuring a RIP Network” on page 345, “Configuring an OSPF Network” on page 359 , or “Configuring the IS-IS Protocol” on page 379. For more information about the IS-IS IGP, see the *JUNOS Routing Protocols Configuration Guide*.

Configuring LDP and RSVP with a Configuration Editor

To configure either LDP or RSVP as a signaling protocol on the device to establish LSPs through an IP network, perform one of the following tasks:

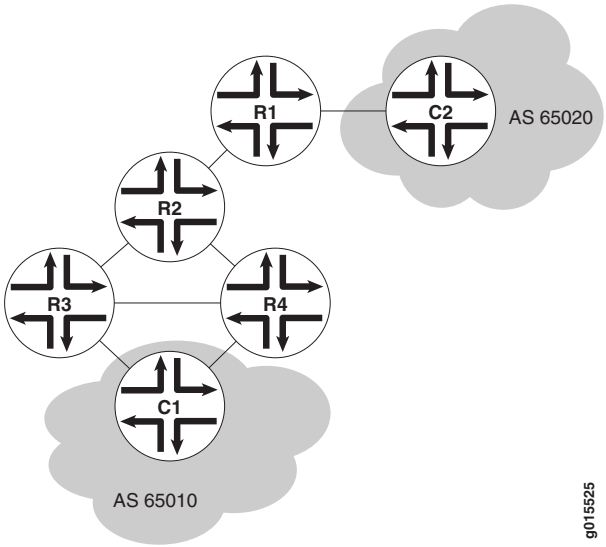
- Configuring LDP-Signaled LSPs on page 429
- Configuring RSVP-Signaled LSPs on page 431

For information about using the J-Web and CLI configuration editors, see the *J-Web Interface User Guide* and the *JUNOS CLI User Guide*.

Configuring LDP-Signaled LSPs

Using LDP as a signaling protocol, you create LSPs between routers in an IP network. A sample network is shown in Figure 77 on page 429.

Figure 77: Typical LDP-Signaled LSP



To establish an LSP between Routers R6 and R7, you must configure LDP on Routers R5, R6, and R7. This configuration ensures that Hosts C1 and C2 use the LDP-signaled LSP when the entry (ingress) router is R6 or R7.

To configure LDP to establish the LSP shown in Figure 77 on page 429, perform these steps:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 130 on page 429.
3. If you are finished configuring the router, commit the configuration.
4. Go on to "Verifying an LDP-Signaled LSP" on page 433.

Table 130: Configuring an LDP-Signaled LSP

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level of the configuration hierarchy	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces

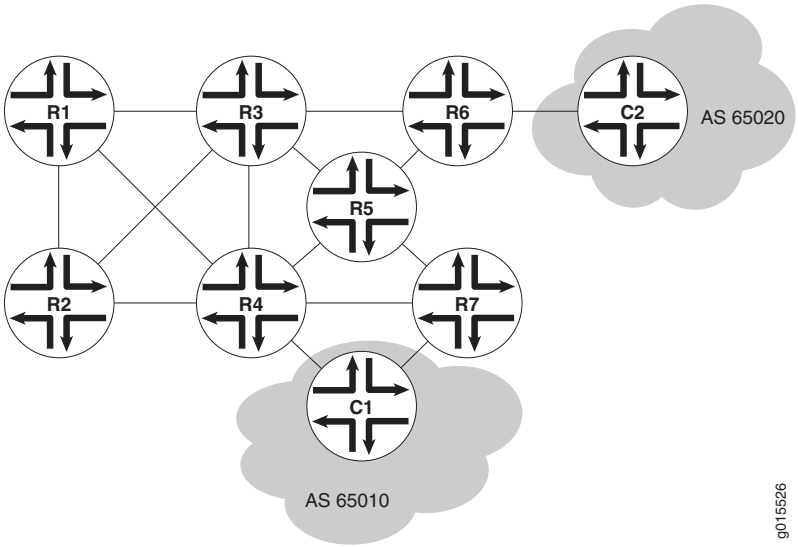
Table 130: Configuring an LDP-Signaled LSP *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Enable the MPLS family on all transit interfaces on each router in the MPLS network.	1. Click the transit interface on which you want to configure MPLS. 2. In the Unit table, click the unit number for which you want to enable MPLS. 3. In the Family area, select the **Mpls** check box. 4. Click **OK**. 5. Repeat Steps 1 through 4 for each transit interface on the routers in the MPLS network.	1. Add the MPLS family to all transit interfaces. For example: set ge-0/0/0 unit 0 family mpls 2. Repeat Step 1 for each transit interface on the routers in the MPLS network.
Enable the MPLS process on all MPLS interfaces for each router in the MPLS network. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Mpls, click **Configure** or **Edit**. 3. Next to Interface, click **Add new entry**. 4. In the Interface name box, type all. 5. Click **OK**. 6. Repeat Steps 1 through 5 for each transit interface on the routers in the MPLS network.	1. From the [edit] hierarchy level, enter edit protocols mpls 2. Enter set interface all 3. Repeat Steps 1 and 2 for each transit interface on the routers in the MPLS network.
Create the LDP instance on each router in the MPLS network.	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Ldp, click **Configure** or **Edit**. 3. Next to Interface, click **Add new entry**. 4. In the Interface name box, type the name of a transit interface—for example, ge-0/0/0. 5. Click **OK**. 6. Repeat Steps 1 through 5 for each transit interface on the routers in the MPLS network.	1. From the [edit] hierarchy level, enter edit protocols ldp 2. Enable LDP on a transit interface. For example: set interface ge-0/0/0 3. Repeat Steps 1 and 2 for each transit interface on the routers in the MPLS network.
Set the keepalive interval to 10 seconds. The keepalive interval specifies the number of seconds between the transmission of keepalive messages along the LDP link.	1. In the Keepalive interval box, type 10. 2. Click **OK**. 3. Repeat Steps 1 and 2 for each router in the MPLS network.	On each router in the MPLS network, enter set keepalive-interval 10

Configuring RSVP-Signaled LSPs

Using RSVP as a signaling protocol, you create LSPs between routers in an IP network. A sample network is shown in Figure 78 on page 431.

Figure 78: Typical RSVP-Signaled LSP



To establish an LSP between routers R1 and R7, you must configure RSVP on all MPLS transit interfaces in the network. This configuration ensures that Hosts C1 and C2 use the RSVP-signaled LSP corresponding to the network IGP's shortest path. Additionally, this configuration reserves 10 Mbps of bandwidth.

To configure RSVP to establish the LSP shown in Figure 78 on page 431, perform these steps:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 131 on page 431.
3. If you are finished configuring the router, commit the configuration.
4. Go on to "Verifying an RSVP-Signaled LSP" on page 435.

Table 131: Configuring an RSVP-Signaled LSP

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Interfaces** level of the configuration hierarchy	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces

Table 131: Configuring an RSVP-Signaled LSP *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Enable the MPLS family on all transit interfaces on each router in the MPLS network.	1. Click the transit interface on which you want to configure MPLS. 2. In the Unit table, click the unit number for which you want to enable MPLS. 3. In the Family area, select the **Mpls** check box. 4. Click **OK**. 5. Repeat Steps 1 through 4 for each transit interface on the routers in the MPLS network.	1. Add the MPLS family to all transit interfaces. For example: set ge-0/0/0 unit 0 family mpls 2. Repeat Step 1 for each transit interface on the routers in the MPLS network.
Enable the MPLS process on all MPLS interfaces for each router in the MPLS network.	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Mpls, click **Configure** or **Edit**. 3. Next to Interface, click **Add new entry**. 4. In the Interface name box, type all. 5. Click **OK**. 6. Repeat Steps 1 through 5 for each transit interface on the routers in the MPLS network.	1. From the [edit] hierarchy level, enter edit protocols mpls 2. Enter set interface all 3. Repeat Steps 1 and 2 for each transit interface on the routers in the MPLS network.
Create the RSVP instance on each router in the MPLS network. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Rsvp, click **Configure** or **Edit**. 3. Next to Interface, click **Add new entry**. 4. In the Interface name box, type the name of a transit interface—for example, ge-0/0/0. 5. Click **OK**. 6. Repeat Steps 1 through 5 for each transit interface on the routers in the MPLS network.	1. From the [edit] hierarchy level, enter edit protocols rsvp 2. Enable RSVP on a transit interface. For example: set interface ge-0/0/0 3. Repeat Steps 1 and 2 for each transit interface on the routers in the MPLS network.
On the entry (ingress) router, R1, define the LSP r1–r7, using Router R7's loopback address (10.0.9.7).	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Mpls, click **Configure** or **Edit**. 3. Next to Label switched path, click **Add new entry**. 4. In the Path name box, type r1–r7. 5. In the To box, type 10.0.9.7.	1. From the [edit] hierarchy level, enter edit protocols mpls 2. Enter set label-switched-path r1–r7 to 10.0.9.7

Table 131: Configuring an RSVP-Signaled LSP *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Reserve 10 Mbps of bandwidth on the LSP.	1. In the Bandwidth box, click **Configure**. 2. In the Ct0 box, type 10m. 3. Click **OK**.	Enter set label-switched-path r1–r7 bandwidth 10m
Disable the use of the Constrained Shortest Path First (CSPF) algorithm. By disabling the CSPF algorithm, you specify that traffic through the LSP is to be routed along the network IGP's shortest path.	1. Select the **No cspf** check box. 2. Click **OK**.	Enter set label-switched-path r1–r7 no-cspf

Verifying an MPLS Configuration

The tasks required to verify your MPLS configuration depend on the signaling protocol used. To validate the configuration, perform the appropriate set of tasks:

- Verifying an LDP-Signaled LSP on page 433
- Verifying an RSVP-Signaled LSP on page 435

Verifying an LDP-Signaled LSP

Suppose that LDP is configured to establish an LSP as shown in Figure 77 on page 429.

To verify the LDP configuration, perform these verification tasks:

- Verifying LDP Neighbors on page 433
- Verifying LDP Sessions on page 434
- Verifying the Presence of LDP-Signaled LSPs on page 435
- Verifying Traffic Forwarding over the LDP-Signaled LSP on page 435

Verifying LDP Neighbors

Purpose Verify that each router shows the appropriate LDP neighbors—for example, that Router R5 has both Router R6 and Router R7 as LDP neighbors.

Action From the CLI, enter the show ldp neighbor command.

Sample Output

```
user@r5> show ldp neighbor
Address     Interface       Label space ID      Hold time
10.0.8.5    ge-0/0/0.0      10.0.9.6:0            14
10.0.8.10   ge-0/0/1.0      10.0.9.7:0            11
```

Meaning The output shows the IP addresses of the neighboring interfaces along with the interface through which the neighbor adjacency is established. Verify the following information:

- Each interface on which LDP is enabled is listed.
- Each neighboring LDP interface address is listed with the appropriate corresponding LDP interface.
- Under Label space ID, the appropriate loopback address for each neighbor appears.

Related Topics For a complete description of show ldp neighbor output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying LDP Sessions

Purpose Verify that a TCP-based LDP session has been established between all LDP neighbors. Also, verify that the modified keepalive value is active.

Action From the CLI, enter the show ldp session detail command.

Sample Output

```
user@r5> show ldp session detail
Address: 10.0.9.7, State: Operational, Connection: Open, Hold time: 28
  Session ID: 10.0.3.5:0--10.0.9.7:0
  Next keepalive in 3 seconds
  Passive, Maximum PDU: 4096, Hold time: 30, Neighbor count: 1
  Keepalive interval: 10, Connect retry interval: 1
  Local - Restart: disabled, Helper mode: enabled
  Remote - Restart: disabled, Helper mode: disabled
  Local maximum recovery time: 240000 msec
  Next-hop addresses received:
    10.0.8.10
    10.0.2.17
```

Meaning The output shows the detailed information, including session IDs, keepalive interval, and next-hop addresses, for each established LDP session. Verify the following information:

- Each LDP neighbor address has an entry, listed by loopback address.
- The state for each session is Operational, and the connection for each session is Open. A state of Nonexistent or a connection of Closed indicates a problem with one of the following:
  - LDP configuration
  - Passage of traffic between the two devices
  - Physical link between the two routers
- For Keepalive interval, the appropriate value, 10, appears.

Related Topics For a complete description of show ldp session detail output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying the Presence of LDP-Signaled LSPs

Purpose Verify that each Juniper Networks device's inet.3 routing table has an LSP for the loopback address on each of the other routers.

Action From the CLI, enter the show route table inet.3 command.

Sample Output

```
user@r5> show route table inet.3
inet.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.9.6/32         *[LDP/9/0] 00:05:29, metric 1
                    > to 10.0.8.5 via ge-0/0/0.0
10.0.9.7/32         *[LDP/9/0] 00:05:37, metric 1
                    > to 10.0.8.10 via ge-0/0/1.0
```

Meaning The output shows the LDP routes that exist in the inet.3 routing table. Verify that an LDP-signaled LSP is associated with the loopback addresses of the other routers in the MPLS network.

Related Topics For a complete description of show route table output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying Traffic Forwarding over the LDP-Signaled LSP

Purpose Verify that traffic between Hosts C1 and C2 is forwarded over the LDP-signaled LSP between Router R6 and Router R7. Because traffic uses any configured gateway address by default, you must explicitly specify that the gateway address is to be bypassed.

Action If Host C1 is a Juniper Networks router, from the CLI enter the traceroute 220.220.0.0 source 200.200.0.1 bypass-routing gateway 172.16.0.1 command.

Sample Output

```
user@c1> traceroute 220.220.0.0 source 200.200.0.1 bypass-routing gateway
172.16.0.1
traceroute to 220.220.0.1 (172.16.0.1) from 200.200.0.1, 30 hops max, 40 byte
packets
 1  172.16.0.1 (172.16.0.1)  0.661 ms  0.538 ms  0.449 ms
 2  10.0.8.9 (10.0.8.9)  0.511 ms  0.479 ms  0.468 ms
    MPLS Label=100004 CoS=0 TTL=1 S=1
 3  10.0.8.5 (10.0.8.5)  0.476 ms  0.512 ms  0.441 ms
 4  220.220.0.1 (220.220.0.1)  0.436 ms  0.420 ms  0.416 ms
```

Meaning The output shows the route that traffic travels between Hosts C1 and C2, without using the default gateway. Verify that traffic sent from C1 to C2 travels through Router R7. The 10.0.8.9 address is the interface address for Router R5.

Related Topics For information about the traceroute command and its output. see the *JUNOS System Basics and Services Command Reference*.

Verifying an RSVP-Signaled LSP

Suppose that RSVP is configured to establish an LSP as shown in Figure 78 on page 431.

To verify the RSVP configuration, perform these verification tasks:

Verifying RSVP Neighbors

Purpose Verify that each device shows the appropriate RSVP neighbors—for example, that Router R1 lists both Router R3 and Router R2 as RSVP neighbors.

Action From the CLI, enter the show rsvp neighbor command.

Sample Output

```
user@r1> show rsvp neighbor
RSVP neighbor: 2 learned
Address            Idle Up/Dn LastChange HelloInt HelloTx/Rx
10.0.6.2              0  3/2       13:01        3   366/349
10.0.3.3              0  1/0       22:49        3   448/448
```

Meaning The output shows the IP addresses of the neighboring routers. Verify that each neighboring RSVP router loopback address is listed.

Related Topics For a complete description of show rsvp neighbor output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying RSVP Sessions

Purpose Verify that an RSVP session has been established between all RSVP neighbors. Also, verify that the bandwidth reservation value is active.

Action From the CLI, enter the show rsvp session detail command.

Sample Output

```
user@r1> show rsvp session detail
Ingress RSVP: 1 sessions

10.0.9.7
  From: 10.0.6.1, LSPstate: Up, ActiveRoute: 0
  LSPname: r1-r7, LSPpath: Primary
  Bidirectional, Upstream label in: -, Upstream label out: -
  Suggested label received: -, Suggested label sent: -
  Recovery label received: -, Recovery label sent: 100000
  Resv style: 1 FF, Label in: -, Label out: 100000
  Time left:    -,  Since: Thu Jan 26 17:57:45 2002
  Tspec: rate 10Mbps size 10Mbps peak Infbps m 20 M 1500
  Port number: sender 3 receiver 17 protocol 0
  PATH rcvfrom: localclient
  PATH sentto: 10.0.4.13 (ge-0/0/1.0) 1467 pkts
  RESV rcvfrom: 10.0.4.13  (ge-0/0/1.0) 1467 pkts
  Record route: <self> 10.0.4.13  10.0.2.1  10.0.8.10
```

Meaning The output shows the detailed information, including session IDs, bandwidth reservation, and next-hop addresses, for each established RSVP session. Verify the following information:

- Each RSVP neighbor address has an entry for each neighbor, listed by loopback address.

- The state for each LSP session is Up.
- Under Tspec, the appropriate bandwidth value, 10Mbps, appears.

Related Topics For a complete description of show rsvp session detail output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying the Presence of RSVP-Signaled LSPs

Purpose Verify that the inet.3 routing table of the entry (ingress) router, R1, has a configured LSP to the loopback address of Router R7.

Action From the CLI, enter the show route table inet.3 command.

Sample Output

```
user@r1> show route table inet.3
inet.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.9.7/32         *[RSVP/7] 00:05:29, metric 10
                    > to 10.0.4.17 via ge-0/0/0.0, label-switched-path r1-r7
```

Meaning The output shows the RSVP routes that exist in the inet.3 routing table. Verify that an RSVP-signaled LSP is associated with the loopback address of the exit (egress) router, R7, in the MPLS network.

Related Topics For a complete description of show route table output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Chapter 21

Configuring Virtual Private Networks

You can configure a Services Router to participate in several types of virtual private networks (VPNs). A VPN allows remote sites and users to use a public communication infrastructure to create secure access to an organization's network. VPNs are a cost-effective alternative to expensive dedicated lines.

There are many ways to set up a VPN and direct traffic through it. This chapter describes the most common tasks involved in setting up a basic Layer 2 VPN, Layer 2 circuit, or Layer 3 VPN configuration. For more information about VPNs, including other configurations and advanced or less common tasks, see the *JUNOS VPNs Configuration Guide*.

You can use either the J-Web configuration editor or the CLI configuration editor to configure VPNs.

This chapter contains the following topics:

VPN Configuration Overview

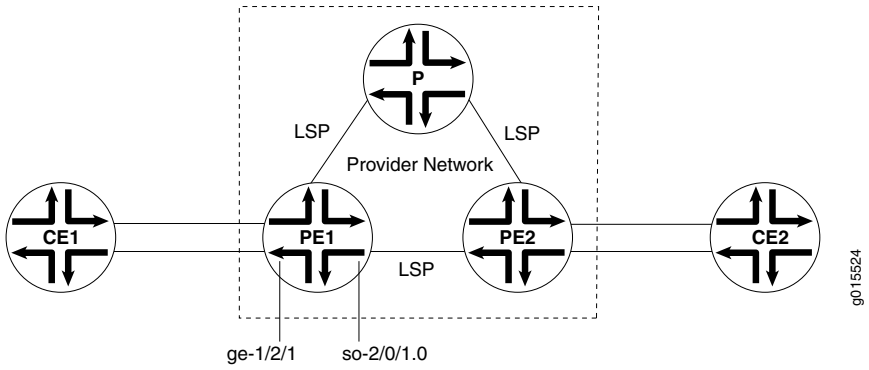
To configure VPN functionality on a Services Router, you must enable support on the provider edge (PE) Services Router as well as configure the Services Router to distribute routing information to other Services Routers in the VPN. The sample configurations in this chapter describe setting up a basic Multiprotocol Label Switching (MPLS) Layer 2 VPN, Layer 3 VPN, and Layer 2 circuit.

This section contains the following topics:

Sample VPN Topology

Figure 79 on page 440 shows the overview of a basic VPN topology for the sample configurations in this chapter.

Figure 79: Basic VPN Topology



Basic Layer 2 VPN Configuration

Implementing a Layer 2 VPN on the Services Router is similar to implementing a VPN using a Layer 2 technology such as Asynchronous Transfer Mode (ATM) or Frame Relay. However, for a Layer 2 VPN on the Services Router, traffic is forwarded to the router in a Layer 2 format. Traffic is then carried by Multiprotocol Label Switching (MPLS) over the service provider's network, and then converted back to Layer 2 format at the receiving end.

On a Layer 2 VPN, routing occurs on the customer's Services Routers, typically on the customer edge (CE) router. The CE Services Router connected to a service provider on a Layer 2 VPN must select the appropriate circuit on which to send traffic. The provider edge (PE) Services Router receiving the traffic sends it across the service provider's network to the PE Services Router connected to the receiving site. PE Services Routers are not required to learn the customer's routes or routing topology, but they must identify the tunnel through which to send the data.

In this sample Layer 2 VPN configuration, the PE routers use the same autonomous system (AS). Within the AS, routing information is communicated through an interior gateway protocol (IGP). Outside the AS, routing information is shared with other ASs through Border Gateway Protocol (BGP). Each AS has a single routing policy and uses a group of one or more IP prefixes. The PE routers must use the same signaling protocols to communicate.

Each routing instance that you configure on a PE router must have a unique route distinguisher associated with it. VPN routing instances need a route distinguisher to help BGP identify overlapping network layer reachability information (NLRIs) messages from different VPNs.

Basic Layer 2 Circuit Configuration

A Layer 2 circuit is a point-to-point Layer 2 connection that transports traffic by means of Multiprotocol Label Switching (MPLS) or another tunneling technology on the service provider network. The Layer 2 circuit creates a virtual connection to direct traffic between two CE Services Routers across a service provider network. The main difference between a Layer 2 VPN and a Layer 2 circuit is the method of setting up the virtual connection. As with a leased line, a Layer 2 circuit forwards all packets received from the local interface to the remote interface.

On the interface communicating with the other PE router, you must specify MPLS and IPv4, and include the IP address. For the loopback interface, you must specify inet, and include the IP address. For IPv4, you must designate the loopback interface as primary so it can receive control packets. Because it is always operational, the loopback interface is best able to perform the control function.

On the PE router interface facing the CE router, you must specify a circuit cross-connect (CCC) encapsulation type. The type of encapsulation depends on the interface type. For example, an Ethernet interface uses ethernet-ccc. The encapsulation type determines how the packet is constructed for that interface.

On the CE router interface that faces the PE router, you must specify inet (for IPv4), and include the IP address. You also specify a routing protocol such as Open Shortest Path First (OSPF) which specifies the area and IP address of the Services Router interface.

With this information, the Services Routers can send and receive packets across the circuit.

Basic Layer 3 VPN Configuration

A Layer 3 VPN operates at the Layer 3 level of the OSI model, the Network layer. In this configuration, the service provider network must learn the IP addresses of devices sending traffic across the VPN. The Layer 3 VPN requires more processing power on the PE Services Routers, because it has larger routing tables for managing network traffic on the customer sites.

A Layer 3 VPN is a set of sites that share common routing information, and connectivity of the sites is controlled by a collection of policies. The sites making up a Layer 3 VPN are connected over a service provider's existing public Internet backbone.

An interface on each CE Services Router communicates with an interface on a PE Services Router through the external Border Gateway Protocol (EBGP).

On the provider Services Router, you configure two interfaces: one to communicate with each PE Services Router. The interfaces communicate with the PE Services Routers by using IPv4 and MPLS. The provider router is in the same AS as the PE routers, which is typically the case for Layer 3 VPNs.

The provider router uses OSPF and Label Distribution Protocol (LDP) to communicate with the PE Services Routers. For OSPF, the provider Services Router interfaces that communicate with the PE routers are specified, as well as the loopback interface.

For the PE routers, the loopback interface is in passive mode, meaning it does not send OSPF packets to perform the control function. In this example, the provider router and PE routers are in the same backbone area. For the LDP configuration, the provider router interfaces that communicate with the PE routers are specified.

Before You Begin

Before you begin configuring VPNs, perform the following tasks:

- Determine which Services Routers are participating in the VPN configuration. This chapter describes configuring an interface for basic VPN connectivity. To configure an interface, see "Configuring Ethernet, DS1, DS3, and Serial Interfaces" on page 73.
- Determine the protocols to use in the VPN configuration. These protocols include
  - MPLS—See "Multiprotocol Label Switching Overview" on page 405 and the *JUNOS Routing Protocols Configuration Guide*.
  - BGP, EBGP, and internal BGP (IBGP)—See "Configuring BGP Sessions" on page 387 and the *JUNOS Routing Protocols Configuration Guide*.
  - LDP and Resource Reservation Protocol (RSVP)—See "Configuring Signaling Protocols for Traffic Engineering" on page 427 and the *JUNOS MPLS Applications Configuration Guide*.
  - OSPF—See "Configuring an OSPF Network" on page 359 and the *JUNOS Routing Protocols Configuration Guide*.

Configuring VPNs with a Configuration Editor

To configure a basic Layer 3 VPN, Layer 2 VPN, or Layer 2 circuit, perform the following tasks. Use Table 132 on page 442 to help you select the tasks for your VPN type. For information about using the J-Web and CLI configuration editors, see the *J-Web Interface User Guide* and the *JUNOS CLI User Guide*.

- Configuring Interfaces Participating in a VPN on page 443
- Configuring Protocols Used by a VPN on page 445
- Configuring a VPN Routing Instance on page 453
- Configuring a VPN Routing Policy on page 455

Table 132: VPN Configuration Task Summary

Section	Layer 3 VPN	Layer 2 VPN	Layer 2 Circuit
"Configuring Interfaces Participating in a VPN" on page 443	All Services Routers	All Services Routers	All Services Routers
"Configuring Protocols Used by a VPN" on page 445	All Services Routers	All Services Routers	All Services Routers

Table 132: VPN Configuration Task Summary *(continued)*

Section	Layer 3 VPN	Layer 2 VPN	Layer 2 Circuit
"Configuring a VPN Routing Instance" on page 453	PE Services Routers	PE Services Routers	N/A
"Configuring a VPN Routing Policy" on page 455	CE Services Routers (PE Services Routers if you are not using a route target)	PE Services Routers if you are not using a route target	N/A

Configuring Interfaces Participating in a VPN

Configuring the Services Router interfaces that participate in the VPN is similar to configuring them for other uses, with a few requirements for VPN.

Before following the procedures in this section, make sure you have initially configured the interface as described in "Configuring Ethernet, DS1, DS3, and Serial Interfaces" on page 73.

To configure an interface for a VPN:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 133 on page 444 for each interface involved in the VPN, except Layer 3 loopback interfaces, which do not require other configuration.
3. Go on to "Configuring Protocols Used by a VPN" on page 445.

Table 133: Configuring an Interface for a VPN

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure IPv4. (interfaces on all Services Routers) (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**. 3. In the Interface name column, select the interface. 4. For Layer 2 VPNs on the interface facing a CE router, select an encapsulation type, such as ethernet-ccc from the Encapsulation list. For Fast Ethernet interfaces, you also must select **Vlan tagging** from the Vlan tag mode list. 5. In the Interface unit number column, select the logical interface. 6. In the Family group, select **Inet** and click **Edit**. 7. Next to Address, click **Add new entry** 8. In the Source box, type the IPv4 address—for example, 10.49.102.1/30. For a loopback address on a Layer 2 configuration, select **Primary**. 9. Click **OK** to return to the Unit page.	■ For all interfaces except loopback, and a Layer 2 VPN interface facing a CE router: From the [edit] hierarchy level, enter edit interfaces *interface-name* unit *logical_interface* family inet address *ipv4_address* ■ For a loopback address on a Layer 2 configuration: From the [edit] hierarchy level, enter edit interfaces lo0 unit *logical_interface*family inet address *ipv4_address* primary ■ For a Layer 2 VPN interface facing a CE router: From the [edit] hierarchy level, enter set interfaces*interface-name* vlan-tagging encapsulation vlan-ccc unit *logical_interface* encapsulation vlan-ccc vlan-id *id-number*
Configure the MPLS address family. (for interfaces on a PE or provider Services Router that communicate with a PE or provider Services Router only, and not for loopback addresses)	On the Unit page, select **Mpls** in the Family group.	At the [edit interfaces *interface*] level, enter set unit *logical_interface*family mpls
For Layer 2 VPNs and circuits, configure encapsulation. If multiple logical units are configured, the encapsulation type is needed at the interface level only. It is always required at the unit level. (for interfaces on a PE Services Router that communicate with a CE Services Router)	1. On the Unit page, select an encapsulation type from the Encapsulation list. 2. Click **OK**. 3. On the Interface page, select an encapsulation type from the Encapsulation list. 4. Click **OK** until you see the Configuration Interfaces page displaying all interfaces on the router.	1. At the [edit interfaces *interface*] level, enter set encapsulation *encapsulation_type* 2. Enter set unit *logical_interface*encapsulation *encapsulation_type*

Configuring Protocols Used by a VPN

The Services Routers in a VPN use a variety of protocols to communicate between PE and provider Services Routers. Use Table 134 on page 445 to help you select the tasks for your VPN type. For more information about configuring routing protocols, see the *JUNOS Routing Protocols Configuration Guide* and the *JUNOS MPLS Applications Configuration Guide*.

This section contains the following topics:

Table 134: VPN Protocol Configuration Task Summary

Section	Layer 3 VPN	Layer 2 VPN	Layer 2 Circuit
"Configuring MPLS for VPNs" on page 445	N/A unless you are using RSVP	PE and provider Services Routers	PE Services Routers
"Configuring a BGP Session" on page 447	PE Services Routers	PE Services Routers	PE Services Routers
"Configuring Routing Options for VPNs" on page 448	All Services Routers	All Services Routers	All Services Routers
"Configuring an IGP and a Signaling Protocol" on page 449—*one* of the following tasks: ■ Configuring LDP for Signaling on page 449 ■ Configuring RSVP for Signaling on page 451	PE and provider Services Routers	PE Services Routers	PE Services Routers
"Configuring a Layer 2 Circuit" on page 452	N/A	N/A	PE Services Routers

Configuring MPLS for VPNs

For Layer 2 VPN and Layer 2 circuit interfaces that communicate with other PE Services Routers and provider Services Routers, you must advertise the interface using MPLS. Unless you are using RSVP, this section does not apply to Layer 3 VPNs because MPLS is configured on the interface.

For more information about configuring MPLS, see "Multiprotocol Label Switching Overview" on page 405*JUNOS MPLS Applications Configuration Guide*.

To configure MPLS for VPNs:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 135 on page 446 on each PE Services Router and provider Services Router interface that communicates with another PE Services Router.
3. If you are finished configuring the router, commit the configuration.
4. To verify the configuration, see "Verifying a VPN Configuration" on page 460
5. Go on to "Configuring a BGP Session" on page 447.

Table 135: Configuring MPLS for VPNs

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the top of the configuration hierarchy and specify the interfaces used for communication between PE routers and between PE routers and provider routers. (PE and provider Services Routers) (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Mpls, click **Configure** or **Edit**. 3. Next to Interface, click **Configure** or **Edit**. 4. In the Interface name box, type *interface-name*. 5. Click **OK**.	From the [edit] hierarchy level, enter the following command for each interface you want to enable: edit protocols mpls interface *interface-name*
For RSVP only, configure an MPLS label-switched path (LSP) to the destination point on the PE router for LSP. During configuration, you specify the IP address of the LSP destination point, which is an address on the remote PE router. The path name is defined on the source Services Router only and is unique between two routers. (PE Services Router interface communicating with another PE Services Router)	1. In the MPLS page, click **Add New Entry** in the Label switched path group. 2. Type a path name in the Path name box and an IP address in the To box. 3. Click **OK**. 4. Next to Interface, click **Add New Entry**. 5. Type *interface-name* in the Interface name box. 6. Click **OK**. 7. Repeat Steps 4 through 6 for each interface.	1. From the [edit] hierarchy level, enter edit protocols mpls label-switched-path *path-name* 2. Enter set to *ip-address* 3. Enter up. 4. Enter interface *interface-name*

Configuring a BGP Session

You must configure an internal BGP (IBGP) session between PE Services Routers so the Services Routers can exchange information about routes originating and terminating in the VPN. The PE routers use this information to determine which labels to use for traffic destined for remote sites. The IBGP session for the VPN runs through the loopback address. This section is valid for Layer 2 VPNs and Layer 3 VPNs, but not Layer 2 circuits.

For the Layer 3 example, you also configure an EBGP session.

For more information about configuring IBGP sessions, see "Configuring BGP Within a Network (Required)" on page 393 and the *JUNOS Routing Protocols Configuration Guide*.

To configure an IBGP session:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 136 on page 448 on each PE router.
3. If you are finished configuring the router, commit the configuration.
4. To verify the configuration, "Verifying a VPN Configuration" on page 460.
5. Go on to "Configuring Routing Options for VPNs" on page 448.

Table 136: Configuring an IBGP Session

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the top of the configuration hierarchy and configure the IGBP session. (PE Services Router)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Bgp, click **Configure** or **Edit**. 3. Next to Group, click **Add New Entry**. 4. Type a name in the Group name box. 5. From the Type list, select **Internal**. 6. In the Local address box, type the local loopback IP address. 7. In the Family group, select **L2vpn** for a Layer 2 VPN or **Inet vpn** for a Layer 3 VPN. 8. Select **Unicast**. 9. Click **OK**. 10. In the Neighbor group, click **Add new entry**. 11. In the Address box, type the loopback IP address of the neighboring PE router. 12. Click **OK** until you return to the BGP page.	1. From the [edit] hierarchy level, enter edit protocols bgp group *group-name* 2. Enter set type internal 3. Enter set local-address *loopback-interface-ip-address* 4. Enter set family *family-type* unicast Replace *family-type* with l2vpn for a Layer 2 VPN or inet–vpn for a Layer 3 VPN. 5. Enter up. 6. Enter the loopback address of the neighboring PE router: set neighbor *ip-address*

Configuring Routing Options for VPNs

The only required routing option for VPNs is the autonomous system (AS) number. You must specify it on each router involved in the VPN.

To configure routing options for a VPN:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration task described in Table 137 on page 449.
3. If you are finished configuring the router, commit the configuration.
4. To verify the configuration, see “Verifying a VPN Configuration” on page 460
5. Go on to “Configuring an IGP and a Signaling Protocol” on page 449.

Table 137: Configuring Routing Options for a VPN

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure the AS number.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing options, click **Configure** or **Edit**. 3. In the AS number box, type the AS number. 4. Click **OK**.	From the [edit] hierarchy level, enter set routing-options autonomous-system *as-number*

Configuring an IGP and a Signaling Protocol

The PE Services Routers and provider Services Routers must be able to exchange routing information. To enable this exchange, you must configure either an IGP such as OSPF or static routes on these routers. You must configure the IGP at the [edit protocols] level, not within the routing instance at the [edit routing-instances] level.

You can use LDP or RSVP between PE routers and between PE routers and provider routers, but not for interfaces between PE routers and CE routers. LDP routes traffic using IGP metrics. RSVP has traffic engineering that lets you override IGP metrics as needed. For more information about these protocols, see “Signaling Protocols Overview” on page 414.

Each PE Services Router's loopback address must appear as a separate route. Do not configure any summarization of the PE Services Router's loopback addresses at the area boundary.

For more information about configuring IGPs and static routes, see “Configuring a RIP Network” on page 345, “Configuring an OSPF Network” on page 359 , “Configuring the IS-IS Protocol” on page 379, “Configuring Static Routes” on page 333, and the *JUNOS Routing Protocols Configuration Guide*.

Configure the appropriate signaling protocol for your VPN:

- Configuring LDP for Signaling on page 449
- Configuring RSVP for Signaling on page 451

Configuring LDP for Signaling

You must configure LDP and OSPF on PE and provider routers. For more information about configuring OSPF see “Configuring an OSPF Network” on page 359.

To configure LDP and OSPF:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 138 on page 450 on PE and provider router interfaces that communicate with a PE router or provider router.

   For the protocols to work properly, you also must configure the MPLS address family for each interface that uses LDP or RSVP, as described previously in "Configuring Interfaces Participating in a VPN" on page 443.
3. If you are finished configuring the router, commit the configuration.
4. To verify the configuration, see "Verifying a VPN Configuration" on page 460.
5. Go on to "Configuring a VPN Routing Instance" on page 453.

Table 138: Configuring LDP and OSPF for Signaling

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the top of the configuration hierarchy and specify the LDP protocol. Enable local interfaces that communicate with a PE router or provider router, and the loopback interface of the PE router. (PE and provider Services Routers) (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Ldp, click **Configure** or **Edit**. 3. Next to Interface, click **Configure** or **Edit**. 4. In the Interface name column, type *interface-name*. 5. Click **OK**. 6. Repeat Steps 4 and 5 for each interface you want to enable.	From the [edit] hierarchy level, enter the following command for each interface you want to enable: edit protocols ldp interface *interface-name*

Table 138: Configuring LDP and OSPF for Signaling *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure OSPF for each interface that uses LDP. For OSPF, you must configure at least one area on at least one of the router's interfaces. An AS can be divided into multiple areas. This example uses the backbone area 0.0.0.0. (PE and provider Services Routers)	For OSPF: 1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Ospf, click **Configure** or **Edit**. 3. For Layer 2 VPN or circuit, select **Traffic engineering**. 4. Next to Area group, click **Add new entry** and add the area. 5. Next to Area group, select the area (**0.0.0.0**). 6. Next to Interface group, select **Add new entry**. 7. In the Interface name box, type ***interface-name***. 8. Click **OK**. 9. Repeat Steps 5 through 7 to enable additional interfaces. 10. Click **OK** twice to return to the Protocols page.	For OSPF: 1. From the [edit] hierarchy level, enter the following command for each interface you want to enable: edit protocols ospf area 0.0.0.0 interface *interface-name* 2. For Layer 2 VPN or circuit, move up to the [edit protocols ospf] level and enter set traffic-engineering

Configuring RSVP for Signaling

You must enable RSVP for all connections that participate in the label-switched path (LSP) on PE and provider Services Routers. In addition, you must configure OSPF on various interfaces.

For more information about configuring OSPF see “Configuring an OSPF Network” on page 359.

To configure RSVP and OSPF:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 139 on page 452 on each PE router and provider router, as specified.
3. If you are finished configuring the router, commit the configuration.
4. To verify the configuration, see “Verifying a VPN Configuration” on page 460.
5. Go on to “Configuring a VPN Routing Instance” on page 453.

Table 139: Configuring RSVP and OSPF for Signaling

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the top of the configuration hierarchy and configure OSPF with traffic engineering support. (PE Services Router)	For OSPF, follow these steps: 1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Configure** or **Edit**. 3. Next to Ospf, click **Configure** or **Edit**. 4. Select **Traffic engineering**, and then click **Configure**. 5. Select **Shortcuts**. 6. Click **OK** until you return to the Protocols page.	From the [edit] hierarchy level, enter the following command for each interface you want to enable: edit protocols ospf traffic-engineering shortcuts
Enable RSVP on interfaces that participate in the LSP. (PE Services Router) Enable interfaces on the source and destination points. (provider Services Router) Enable interfaces that connect the LSP between the PE Services Routers. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Rsvp, click **Configure** or **Edit**. 3. In the Interface group, click **Add New Entry**. 4. Type an interface name. 5. Click **OK**. 6. Repeat Steps 2 through 4 for each interface you want to enable. 7. Click **OK**.	From the [edit] hierarchy level, enter the following command for each interface you want to enable: edit protocols rsvp interface *interface-name*

Configuring a Layer 2 Circuit

Each Layer 2 circuit is represented by the logical interface connecting the local PE Services Router to the local CE Services Router. All Layer 2 circuits using a particular remote PE Services Router neighbor is identified by its IP address and is usually the endpoint destination for the LSP tunnel transporting the Layer 2 circuit.

You configure a virtual circuit ID on each interface. Each virtual circuit ID uniquely identifies the Layer 2 circuit among all the Layer 2 circuits to a specific neighbor. The key to identifying a particular Layer 2 circuit on a PE router is the neighbor address and the virtual circuit ID. Based on the virtual circuit ID and the neighbor relationship, an LDP label is bound to an LDP circuit. LDP uses the binding for sending traffic on that Layer 2 circuit to the remote CE router.

To configure a Layer 2 circuit:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 140 on page 453 on each PE router and provider router, as specified.
3. If you are finished configuring the router, commit the configuration.
4. To verify the configuration, see "Verifying a VPN Configuration" on page 460.

Table 140: Configuring a Layer 2 Circuit

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the top of the configuration hierarchy and enable a Layer 2 circuit on the appropriate interface. (PE Services Router) (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Configure** or **Edit**. 3. Next to L2circuit, click **Configure** or **Edit**. 4. Next to Neighbor, click **Add new entry**. 5. In the Neighbor box, enter the loopback address of the local router. 6. Next to Interface, click **Add new entry**. 7. In the **Interface** box, type the interface name of the remote PE router. 8. In the Virtual circuit id box, type an ID number. 9. Click **OK** until you return to the Protocols page.	1. From the [edit] hierarchy level, enter edit protocols l2circuit neighbor *interface-name* interface *interface-name* For neighbor, specify the local loopback address, and for interface, specify the interface name of the remote PE router. 2. Enter set virtual-circuit-id *id-number*

Configuring a VPN Routing Instance

You must configure a routing instance for each VPN on each PE Services Router participating in the VPN. The routing instance has the same name on each PE router. VPN routing instances need a route distinguisher to help BGP distinguish between potentially identical network layer reachability information (NLRI) messages received from different VPNs. This section does not apply to Layer 2 circuit configurations.

Each routing instance that you configure on a PE router must have a unique route distinguisher. There are two possible formats:

- *as-number*:*number*, where *as-number* is an autonomous system (AS) number (a 2–byte value) in the range 1 through 65,535, and *number* is any 4–byte value. We recommend that you use an Internet Assigned Numbers Authority (IANA)-assigned, nonprivate AS number, preferably the ISP or the customer AS number.
- *ip-address*:*number*, where *ip-address* is an IP address (a 4–byte value) and *number* is any 2–byte value. The IP address can be any globally unique unicast address.

We recommend that you use the address that you configure in the router-id statement, which is a public IP address in your assigned prefix range.

The route target defines which route is part of a VPN. A unique route target helps distinguish between different VPN services on the same router. Each VPN also has a policy that defines how routes are imported into the VPN routing and forwarding (VRF) table on the router. A Layer 2 VPN is configured with import and export policies. A Layer 3 VPN uses a unique route target to distinguish between VPN routes.

To configure a VPN routing instance:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 141 on page 454 on each PE router.
3. If you are finished configuring the router, commit the configuration.
4. To verify the configuration, see "Verifying a VPN Configuration" on page 460.
5. Go on to "Configuring a VPN Routing Policy" on page 455.

Table 141: Configuring a VPN Routing Instance

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the top of the configuration hierarchy and create the routing instance. (PE Services Router)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing instances, click **Configure** or **Edit**. 3. Next to Mpls, click **Configure** or **Edit**. 4. In the Instance group, click **Add New Entry**. 5. Type a name in the Instance name box.	From the [edit] hierarchy level, enter edit routing-instances *routing-instance-name*
Specify a text description for the routing instance. This text appears in the output of the show route instance detail command. (PE Services Router)	In the Description box, type a description.	Enter set description "*text*"
Specify the instance type, either l2vpn for Layer 2 VPNs or vrf for Layer 3 VPNs. (PE Services Router)	From the Instance type list, select an instance type.	Enter set instance-type*instance-type*

Table 141: Configuring a VPN Routing Instance *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Specify the interface of the remote PE Services Router. (PE Services Router) (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. Next to Interface group, click **Add New Entry**. 2. In the Interface name box, enter *interface-name*. 3. Click **OK**.	Enter set interface *interface-name*
Specify the route distinguisher. (PE Services Router)	In the Rd type box, enter a route distinguisher in the format *as-number*:*number*or *ip-address*:*number*.	Enter one of the following commands: ■ set route-distinguisher*as-number*:*number* ■ set route-distinguisher *ip-address*:*number*
Specify the policy for the Layer 2 VRF table. For the Layer 2 VPN example, the routing policies are defined in "Configuring a Routing Policy for Layer 2 VPNs" on page 456. (PE Services Router)	For the sample Layer 2 VPN configuration, which uses import and export policies: 1. Next to Vrf export group, select **Add new entry**. 2. In the Value box, type the export routing policy name. 3. Click **OK**. 4. Next to Vrf import group, click **Add new entry**. 5. In the Value box, type the import routing policy name. 6. Click **OK**.	For the sample Layer 2 VPN configuration, which uses import and export policies, enter set vrf-import *import-policy-name* vrf-export *export-policy-name*
Specify the policy for the Layer 3 VRF table. For the Layer 3 VPN example, the routing policy is defined in "Configuring a Routing Policy for Layer 3 VPNs" on page 459. (PE Services Router)	For the sample Layer 3 VPN configuration, which uses a route target: 1. In the Vrf target box, click **Configure**. 2. In the Community box, type the community (target:*community-id*, where *community-id* is *as-number*:*number* or *ip-address*:*number*). 3. Click **OK**.	For the sample Layer 3 VPN configuration, which uses a route target, enter set vrf-target target:*community-id* Replace *community-id* with either of the following: ■ *as-number*:*number* ■ *ip-address*:*number*

Configuring a VPN Routing Policy

Layer 2 and Layer 3 VPNs require a routing policy that describes which packets are sent and received across the VPN. Layer 2 circuits do not use a policy, and therefore, Layer 2 circuits send and receive all packets. For Layer 2 VPNs, the routing policy resides on the PE Services Routers. For the Layer 3 VPN example, the routing policy resides on the CE Services Routers.

This section contains the following topics. For more information about configuring routing policies, see "Configuring Routing Policies" on page 501 and the *JUNOS Routing Protocols Configuration Guide*.

Configuring a Routing Policy for Layer 2 VPNs

If the routing instance uses a policy for accepting and rejecting packets instead of a route target, you must specify the import and export routing policies and the community on each PE Services Router.

To configure a Layer 2 VPN routing policy on a PE Services Router:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 142 on page 456 and Table 143 on page 458 on each PE router.
3. If you are finished configuring the router, commit the configuration.
4. To verify the configuration, see "Verifying a VPN Configuration" on page 460.

Table 142: Configuring an Import Routing Policy for Layer 2 VPNs

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the top of the configuration hierarchy and configure the import routing policy. (PE Services Router)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**. 3. Next to Policy statement, click **Add new entry**. 4. In the Policy name box, type the policy name—for example, import_vpn.	From the [edit] hierarchy level, enter edit policy-options policy-statement *import-policy-name*

Table 142: Configuring an Import Routing Policy for Layer 2 VPNs *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define the term for accepting packets. (PE Services Router)	1. Next to Term group, click **Add new entry**. 2. In the Term name box, type a term name—for example, 10. 3. Next to From, click **Configure**. 4. Click **Add new entry**. 5. Click **Protocol** and select **bgp** from the Value menu. 6. Click **OK**. 7. Next to Community, click **Add new entry**. 8. Type the *community-name* value in the Community Name box. 9. Click **OK**. 10. Next to Then, click **Configure**. 11. From the Accept reject list, select **accept**. 12. Click **OK** until you are at the Policy statement page.	1. Enter set term*term-name-accept* from protocol bgp community *community-name* 2. Enter set term*term-name-accept* then accept
Define the term for rejecting packets. (PE Services Router)	1. Next to the Term group, click **Add new entry**. 2. In the Term name box, type a term name—for example, 20. 3. Next to Then, click **Configure**. 4. From the Accept list, select **reject**. 5. Click **OK** until you return to the Policy options page.	Enter set term *term-name-reject* then reject

After configuring an import routing policy for a Layer 2 VPN, configure an export routing policy for the Layer 2 VPN. The export routing policy defines how routes are exported from the PE Services Router routing table. An export policy is applied to routes sent to other PE Services Routers in the VPN. The export policy must also evaluate all routes received over the routing protocol session with the CE Services Router. The export policy must also contain a second term for rejecting all other routes.

Table 143: Configuring an Export Routing Policy for Layer 2 VPNs

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure the export routing policy. (PE Services Router)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**. 3. Next to Policy statement, click **Add new entry**. 4. In the Policy name box, type the policy name—for example, export_vpn.	From the [edit] hierarchy level, enter edit policy-options policy-statement *export-policy-name*
Define the term for accepting packets. (PE Services Router)	1. Next to the Term group, click **Add new entry**. 2. In the Term name box, type a term name—for example, 10. 3. Next to From, click **Configure**. 4. Next to Community, click **Add new entry**. 5. Type the *community-name* value in the Community Name box. 6. Click **OK**. 7. Next to Then, click **Configure**. 8. From the Accept reject list, select **accept**. 9. Click **OK** twice until you are at the Policy statement page.	1. Enter set term*term-name-accept* from community add *community-name* 2. Enter set term*term-name-accept* then accept
Define the term for rejecting packets. (PE Services Router)	1. Next to the Term group, click **Add new entry**. 2. In the Term name box, type a term name—for example, 20. 3. Next to Then, click **Configure**. 4. From the Accept reject list, select **reject**. 5. Click **OK** until you return to the Policy options page.	1. Enter set term*term-name-reject* from community add *community-name* 2. Enter set term*term-name-reject* then reject

Table 143: Configuring an Export Routing Policy for Layer 2 VPNs *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define the community. (PE Services Router)	1. In the Community group, click **Add new entry**. 2. In the Community name box, type a community name—for example, VPN. 3. In the Members group, click **Add new entry**. 4. In the Value box, type target:*community-id*, where *community-id* is *as-number*:*number* or *ip-address*:*number*. 5. Click **OK** until you return to the Policy options page.	Type the following commands: community*community-name*target:*as-number* or *ip-address*:*number*

Configuring a Routing Policy for Layer 3 VPNs

To configure a Layer 3 VPN routing policy on a CE Services Router:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 144 on page 459 on each CE Services Router.
3. If you are finished configuring the router, commit the configuration.
4. To verify the configuration, see "Verifying a VPN Configuration" on page 460.

Table 144: Configuring a Routing Policy for Layer 3 VPNs

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the top of the configuration hierarchy and configure the routing policy for the loopback interface. (CE Services Router)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**. 3. Next to Policy statement, click **Configure** or **Edit**. 4. In the Policy name box, type the policy name—for example, loopback.	From the [edit] hierarchy level, enter edit policy-options policy-statement *policy-name*

Table 144: Configuring a Routing Policy for Layer 3 VPNs *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define the term for accepting packets. (CE Services Router)	1. In the Term group, click **Add new entry**. 2. In the Term name box, type a term name—for example, 1. 3. Next to From, click **Configure**. 4. Click **protocol**, then **Add new entry**. 5. Select **direct** from the Value menu, and click **OK**. 7. Next to Route Filter, click **Add new entry**. 8. Type *local-loopback-address/netmask* in the Address box. 9. Select **exact** from the Modifier list. 10. Click **OK** twice. 11. Next to Then, click **Configure**. 12. From the Accept reject list, select **accept**. 13. Click **OK** until you are at the Policy statement page.	1. Enter set term*term-name-accept* from protocol direct route-filter *local-loopback-address/netmask* exact 2. Enter set term*term-name-accept* then accept
Define the term for rejecting packets. (CE Services Router)	1. Next to the Term group, click **Add new entry**. 2. In the Term name box, type a term name—for example, 2. 3. Next to Then, click **Configure**. 4. From the Accept reject list, select **reject**. 5. Click **OK** until you return to the Policy options page.	Enter set term*term-name-reject* then reject

Verifying a VPN Configuration

To verify the connectivity of Layer 2 VPNs, Layer 3 VPNs, and Layer 2 circuits, use the ping mpls command. This command helps to verify that a VPN or circuit has been enabled. This command tests the integrity of the VPN or Layer 2 circuit connection between the PE Services Routers. It does not test the connection between a PE and a CE Services Router.

This section contains the following topics:

Pinging a Layer 2 VPN

To ping a Layer 2 VPN, use one of the following commands:

- ping mpls l2vpn interface*interface-name*

  Ping an interface configured for the Layer 2 VPN on the PE router.

- ping mpls l2vpn instance *l2vpn-instance-name* local-site-id*local-site-id-number* remote-site-id*remote-site-id-number*

  Ping a combination of the Layer 2 VPN routing instance name, the local site identifier, and the remote site identifier to test the integrity of the Layer 2 VPN connection (specified by identifiers) between the two PE Services Routers.

Pinging a Layer 3 VPN

To ping a Layer 3 VPN, use the following command:

ping mpls l3vpn *l3vpn-name*prefix*prefix* <count *count*>

Ping a combination of a IPv4 destination prefix and a Layer 3 VPN name on the destination PE Services Router to test the integrity of the VPN connection between the source and destination Services Routers. The destination prefix corresponds to a prefix in the Layer 3 VPN. However, ping tests only whether the prefix is present in a PE VRF table.

Pinging a Layer 2 Circuit

To ping a Layer 2 circuit, use one of the following commands:

- ping mpls l2circuit interface*interface-name*

  Ping an interface configured for the Layer 2 circuit on the PE Services Router.

- ping mpls l2circuit virtual-circuit<*prefix*> <*virtual-circuit-id*>

  Ping a combination of the IPv4 prefix and the virtual circuit ID on the destination PE Services Router to test the integrity of the Layer 2 circuit between the source and destination Services Routers.

Chapter 22

Configuring CLNS VPNs

Connectionless Network Service (CLNS) is a Layer 3 protocol similar to IPv4 for linking hosts (end systems) with routers (intermediate systems) in an Open Systems Interconnection (OSI) network. CLNS and its related OSI protocols, Intermediate System-to-Intermediate System (IS-IS) and End System-to-Intermediate System (ES-IS), are International Organization for Standardization (ISO) standards.

You can configure J-series Services Routers as provider edge (PE) routers within a CLNS network. CLNS networks can be connected over an IP MPLS network core using BGP and MPLS Layer 3 virtual private networks (VPNs). For more information, see RFC 2547, *BGP/MPLS VPNs*.

You can use either the J-Web configuration editor or CLI configuration editor to configure CLNS.

This chapter contains the following topics. For more information about CLNS, IS-IS, and ES-IS, see the *JUNOS Routing Protocols Configuration Guide*.

CLNS Terms

Before configuring CLNS, become familiar with the terms defined in Table 145 on page 463.

Table 145: CLNS Terms

Term	Definition
CLNS island	Typically one IS-IS level 1 area that is part of a single IGP routing domain. An island can contain more than one area. CLNS islands can be connected by virtual private networks (VPNs).
Connectionless Network Service (CLNS)	Layer 3 protocol similar to IPv4 for linking hosts (end systems) with routers (intermediate systems) in an Open Systems Interconnection (OSI) network, by using network service access points (NSAPs) instead of prefix addresses to specify hosts and routers.

Table 145: CLNS Terms *(continued)*

Term	Definition
customer edge (CE) router	Router or switch in the customer's network that is connected to a service provider's provider edge (PE) router and participates in a Layer 3 VPN.
end system	A host in an Open Systems Interconnection (OSI) network.
End System-to-Intermediate System (ES-IS)	Protocol that enables end systems (hosts) and intermediate systems (routers) to discover each other, by a method similar to Address Resolution Protocol (ARP) discovery in an IPv4 network.
intermediate system	A router in an Open Systems Interconnection (OSI) network.
International Organization for Standardization (ISO)	Worldwide federation of standards bodies that promotes international standardization and published international agreements as International Standards.
network layer reachability information (NLRI)	Information about routes exchanged in update messages by Border Gateway Protocol (BGP) systems, to enable routers to determine the relationships among all known BGP autonomous systems.
network services access point (NSAP)	International Standards Organization (ISO) addressing method for identifying hosts (end systems) and routers (intermediate systems) at the data-link layer (Layer 3) in an Open Systems Interconnection (OSI) network. An NSAP is from 8 to 20 bytes long and consists of an area address, a system ID, and an NSAP selector (NSEL) byte.
Open Systems Interconnection (OSI)	Standard reference model for representing the way messages are transmitted between two points on a network.
provider edge (PE) router	Services Router in the service provider network that is connected to a customer edge (CE) router and participates in a virtual private network (VPN).
virtual private network (VPN)	Private data network that uses a public TCP/IP network, typically the Internet, while maintaining privacy with a tunneling protocol, encryption, and security procedures.

CLNS Overview

CLNS uses network service access points (NSAPs), similar to IP addresses found in IPv4, to identify end systems (hosts) and intermediate systems (routers). ES-IS enables the hosts and routers to discover each other. IS-IS is the interior gateway protocol (IGP) that carries ISO CLNS routes through a network.

Depending on your network topology, one or more of the following components are needed to route within a CLNS environment:

- ES-IS—Provides the basic interaction between CLNS hosts (end systems) and routers (intermediate systems). Using ES-IS, hosts advertise their ISO NSAP addresses and subnetwork point-of-attachment (SNPA) addresses to other routers and hosts attached to the subnetwork. The resolution of Layer 3 ISO NSAPs to Layer 2 SNPAs by ES-IS is equivalent to ARP within an IPv4 network.

  If a CLNS island does not contain any end systems, you do not need to configure ES-IS on a device.

- IS-IS extensions—Provide the basic IGP support for collecting intradomain routing information for CLNS destinations within a CLNS network. Routers learning host addresses through ES-IS can advertise them to other routers (intermediate systems) using IS-IS.

- Static routes—You can configure static routes to exchange CLNS routes within a CLNS island. You can use static routing with or without IS-IS.

- Border Gateway Protocol (BGP) extensions—BGP extensions allow BGP to carry CLNS VPN network layer reachability information (NLRI) between PE routers. Each CLNS route is encapsulated into a CLNS VPN NLRI and propagated between remote sites in a VPN.

For more information about CLNS, see the ISO 8473 standards. For more information about IS-IS, see the ISO 10589 standard. For more information about ES-IS, see the ISO 9542 standard.

Before You Begin

Before you begin configuring CLNS, complete the following tasks:

- Configure IS-IS. See the *JUNOS Routing Protocols Configuration Guide*.
- Configure the network interfaces. See “Configuring Ethernet, DS1, DS3, and Serial Interfaces” on page 73.
- If applicable, configure BGP and VPNs. See “Configuring BGP Sessions” on page 387 and “Configuring Virtual Private Networks” on page 439.

Configuring CLNS with a Configuration Editor

To configure CLNS, you must perform the first task and then one or more of the following tasks (depending on your network):

- Configuring a VPN Routing Instance (Required) on page 466
- Configuring ES-IS on page 467
- Configuring IS-IS for CLNS on page 468
- Configuring CLNS Static Routes on page 470
- Configuring BGP for CLNS on page 471

NOTE: Many of the configuration statements used in this section can be included at different hierarchy levels in the configuration. For more information, see the *JUNOS Routing Protocols Configuration Guide*.

Configuring a VPN Routing Instance (Required)

You typically configure ES-IS, IS-IS, and CLNS static routes using a VPN routing instance. For more information about routing instances, see "Configuring a VPN Routing Instance" on page 453.

To configure a VPN routing instance:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 146 on page 466.
3. Go on to one of the following tasks:
   - Configuring IS-IS for CLNS on page 468
   - Configuring CLNS Static Routes on page 470
   - Configuring BGP for CLNS on page 471
   - Verifying CLNS VPN Configuration on page 471

Table 146: Configuring a VPN Routing Instance for CLNS

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the top of the configuration hierarchy and create the routing instance aaaa.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing instances, click **Configure** or **Edit**. 3. Next to Instance, click **Add new entry**. 4. In the Instance name box, type aaaa. 5. Click **OK**.	From the [edit] hierarchy level, enter edit routing-instances aaaa
Specify the instance type vrf for Layer 3 VPNs.	In the Instance type list, select **vrf**.	Enter set instance-type vrf

Table 146: Configuring a VPN Routing Instance for CLNS *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Specify the interfaces that belong to the routing instance aaaa—for example, lo0.1, e1–2/0/0.0, and t1–3/0/0.0. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. Next to Interface, click **Add New Entry**. 2. In the Interface name box, type lo0.1. 3. Click **OK**. 4. Next to Interface, click **Add New Entry**. 5. In the Interface name box, type e1–2/0/0.0. 6. Click **OK**. 7. Next to Interface, click **Add New Entry**. 8. In the Interface name box, type t1–3/0/0.0. 9. Click **OK**.	Enter 1. set interface lo0.1 2. set interface e1–2/0/0.0 3. set interface t1–3/0/0.0
Specify the route distinguisher—for example, 10.255.245.1:1.	In the Rd type box, type 10.255.245.1:1.	Enter set route-distinguisher 10.255.245.1:1
Specify the policy for the Layer 3 VRF table—for example, target:11111:1.	1. Next to Vrf target, click **Configure**. 2. In the Community box, type target:11111:1. 3. Click **OK**.	Enter set vrf-target target:11111:1

Configuring ES-IS

If a device is a PE router within a CLNS island that contains any end systems, you must configure ES-IS on the device.

To configure ES-IS for the J-series Services Router:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 147 on page 468.
3. If you are finished configuring the router, commit the configuration.
4. If applicable, go on to one of the following tasks:
   - Configuring IS-IS for CLNS on page 468
   - Configuring CLNS Static Routes on page 470
   - Configuring BGP for CLNS on page 471
   - Verifying CLNS VPN Configuration on page 471

Table 147: Configuring ES-IS

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Routing instances** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing instances, click **Configure** or **Edit**. 3. Under Instance name, click **aaaa**.	From the [edit] hierarchy level, enter edit routing-instances aaaa
Enable ES-IS on all interfaces.	1. Next to Protocols, click **Configure**. 2. Next to Esis, click **Configure**. 3. Next to Interface, click **Add new entry**. 4. In the Interface name box, type **all**. 5. Click **OK** until you return to the Protocols statement page.	Enter set protocols esis interface all

Configuring IS-IS for CLNS

You can configure IS-IS to exchange CLNS routes within a CLNS island. To export BGP routes into IS-IS, you must configure and apply an export policy. For more information about policies, see “Configuring Routing Policies” on page 501.

If you have a pure CLNS island—an island that does not contain any IP devices—you must disable IPv4 and IPv6 routing.

To configure IS-IS for CLNS:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 148 on page 468.
3. If you are finished configuring the router, commit the configuration.
4. If applicable, go on to one of the following tasks:
   - Configuring CLNS Static Routes on page 470
   - Configuring BGP for CLNS on page 471
   - Verifying CLNS VPN Configuration on page 471

Table 148: Configuring IS-IS to Exchange CLNS Routes

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Routing instances** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing instances, click **Configure** or **Edit**. 3. Under Instance name, click **aaaa**.	From the [edit] hierarchy level, enter edit routing-instances aaaa

Table 148: Configuring IS-IS to Exchange CLNS Routes *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Enable CLNS routing.	1. Next to Protocols, click **Configure**. 2. Next to Isis, click **Configure**. 3. Next to CLNS routing, select the **Yes** box.	Enter set protocols isis clns-routing
Enable IS-IS on all interfaces. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. Next to Interface, click **Add new entry**. 2. In the Interface name box, type all. 3. Click **OK**.	Enter set protocols isis interface all
(Optional) To configure a pure CLNS network, disable IPv4 and IPv6 routing.	1. Next to No ipv4 routing, select the **Yes** box. 2. Next to No ipv6 routing, select the **Yes** box. 3. Click **OK**.	Enter set protocols isis no-ipv4-routing no-ipv6-routing
Define the BGP export policy name—for example, dist-bgp—and the family and protocol.	1. On the main Configuration page next to Policy options, click **Configure** or **Edit**. 2. Next to Policy statement, click **Add new entry**. 3. In the Policy name box, type dist-bgp. 4. Next to From, click **Configure**. 5. In the Family list, select **iso**. 6. Next to Protocol, click **Add new entry**. 7. In the Value list, select **bgp**. 8. Click **OK** until you return to the Policy statement page.	From the [edit] hierarchy level, enter set policy-options policy-statement dist-bgp from family iso protocol bgp
Define the action for the export policy.	1. Next to Then, click **Configure**. 2. In the Accept reject list, select **accept**. 3. Click **OK** until you return to the main Configuration page.	From the [edit] hierarchy level, enter set policy-options policy-statement dist-bgp then accept
Apply the export policy to IS-IS.	1. On the main Configuration page next to Routing instances, click **Configure** or **Edit**. 2. Next to aaaa, click **Protocols**. 3. Next to Isis, click **Edit**. 4. Next to Export, click **Add new entry**. 5. In the Value box, type dist-bgp. 6. Click **OK** until you return to the Instance page.	From the [edit] hierarchy level, enter set routing-instances aaaa protocols isis export dist-bgp

Configuring CLNS Static Routes

If some devices in your network do not support IS-IS, you must configure CLNS static routes. You might also consider using static routes if your network is simple.

This procedure, as well as the configuration provided in "Verifying CLNS VPN Configuration" on page 471, uses the following ISO NET address and NSAP prefix:

- 47.0005.80ff.f800.0000.aaaa.1000.1921.6800.4196.00
- 47.0005.80ff.f800.0000.bbbb.1022/104

To configure CLNS static routes:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 149 on page 470.
3. If you are finished configuring the router, commit the configuration.
4. If applicable, go on to one of the following tasks:
   - Configuring BGP for CLNS on page 471
   - Verifying CLNS VPN Configuration on page 471

Table 149: Configuring Static CLNS Routes

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Routing instances** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Routing instances, click **Configure** or **Edit**. 3. Under Instance name, click **aaaa**.	From the [edit] hierarchy level, enter edit routing-instances aaaa
Configure the next-hop ISO NET address for an NSAP prefix.	1. Next to Routing options, click **Configure**. 2. Next to Rib, click **Add new entry**. 3. In the Rib name box, type aaaa.iso.0. 4. Next to Static, click **Configure**. 5. Next to Iso route, click **Add new entry**. 6. In the Destination box, type 47.0005.80ff.f800.0000.bbbb.1022/104. 7. From the Next hop list, select **Next hop**. 8. Next to Next hop, click **Add new entry**. 9. In the Value box, type 47.0005.80ff.f800.0000.aaaa.1000.1921.6800.4196.00. 10. Click **OK**.	Enter set routing-options iso-route 47.0005.80ff.f800.0000.bbbb.1022/104 next-hop 47.0005.80ff.f800.0000.aaaa.1000.1921.6800.4196.00

Configuring BGP for CLNS

To configure BGP to carry CLNS VPN NLRI:

1. Navigate to the top of the configuration hierarchy in either the J-Web or the CLI configuration editor.
2. Perform the configuration tasks described in Table 150 on page 471.
3. If you are finished configuring the router, commit the configuration.
4. To verify the configuration, see "Verifying CLNS VPN Configuration" on page 471.

Table 150: Configuring BGP to Carry CLNS VPN NLRI Messages

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Bgp** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Protocols, click **Configure** or **Edit**. 3. Next to Bgp, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter set protocols bgp group pedge-pedge neighbor 10.255.245.215 family iso-vpn unicast
Define a BGP group name—for example, pedge-pedge.	1. Next to Group, click **Add new entry**. 2. In the Group name box, type pedge-pedge.	
Define a BGP peer neighbor address for the group—for example, 10.255.245.215.	1. Next to Neighbor, click **Add new entry**. 2. In the Address box, type 10.255.245.215.	
Define the family.	1. Under Family, next to Iso vpn, click **Configure**. 2. Next to Unicast, select the **Yes** box. 3. Click **OK**.	

Verifying CLNS VPN Configuration

Verify that the device is configured correctly for CLNS VPNs.

Displaying CLNS VPN Configuration

Purpose Verify the configuration of CLNS VPNs.

Action From the J-Web interface, select **Configuration > View and Edit > View Configuration Text**. Alternatively, from configuration mode in the CLI, enter the show command.

```
[edit]
user@host# show
interfaces {
   e1–2/0/0.0 {
      unit 0 {
```

```
            family inet {
               address 192.168.37.51/31;
            }
            family iso;
            family mpls;
         }
      }
      t1–3/0/0.0 {
         unit 0 {
            family inet {
               address 192.168.37.24/32;
            }
            family iso;
            family mpls;
         }
      }
      lo0 {
         unit 0 {
            family inet {
               address 127.0.0.1/32;
               address 10.255.245.215/32;
            }
            family iso {
               address 47.0005.80ff.f800.0000.0108.0001.1921.6800.4215.00;
            }
         }
         unit 1 {
            family iso {
               address 47.0005.80ff.f800.0000.0108.aaa2.1921.6800.4215.00;
            }
         }
      }
   }
   routing-options {
      autonomous-system 230;
   }
   protocols {
      bgp {
         group pedge-pedge {
            type internal;
            local-address 10.255.245.215;
            neighbor 10.255.245.212 {
               family iso-vpn {
                  unicast;
               }
            }
         }
      }
   }
   policy-options {
      policy-statement dist-bgp {
         from {
            protocol bgp;
            family iso;
         }
         then accept;
```

```
    }
  }
  routing-instances {
    aaaa {
      instance-type vrf;
      interface lo0.1;
      interface e1–2/0/0.0;
      interface t1–3/0/0.0;
      route-distinguisher 10.255.245.1:1;
      vrf-target target:11111:1;
      routing-options {
        rib aaaa.iso.0 {
          static {
            iso-route 47.0005.80ff.f800.0000.bbbb.1022/104
              next-hop 47.0005.80ff.f800.0000.aaaa.1000.1921.6800.4196.00;
          }
        }
      }
      protocols {
        esis {
          interface all;
        }
        isis {
          export dist-bgp;
          no-ipv4–routing;
          no-ip64–routing;
          clns–routing;
          interface all;
        }
      }
    }
  }
```

Meaning Verify that the output shows the intended configuration of CLNS VPNs.

Related Topics For more information about the format of a configuration file, see the *JUNOS CLI User Guide*.

Chapter 23

Configuring Virtual Private LAN Service

Virtual private LAN service (VPLS) is an Ethernet-based point-to-multipoint Layer 2 virtual private network (VPN). It allows you to connect geographically dispersed Ethernet LAN sites to each other across an MPLS backbone. For customers who implement VPLS, all sites appear to be in the same Ethernet LAN even though traffic travels across the service provider's network.

This chapter includes:

VPLS Overview

VPLS is an Ethernet-based point-to-multipoint Layer 2 VPN. It allows you to connect geographically dispersed Ethernet LAN sites to each other across an MPLS backbone. For customers who implement VPLS, all sites appear to be in the same Ethernet LAN even though traffic travels across the service provider's network.

VPLS, in its implementation and configuration, has much in common with a Layer 2 VPN. In a VPLS topology, a packet originating within a customer’s network is sent first to a customer edge (CE) device (for example, a router or Ethernet switch). It is then sent to a provider edge (PE) router within the service provider’s network. The

packet traverses the service provider's network over an MPLS label-switched path (LSP). It arrives at the egress PE router, which then forwards the traffic to the CE device at the destination customer site.

The difference is that for VPLS, packets can traverse the service provider's network in point-to-multipoint fashion, meaning that a packet originating from a CE device can be broadcast to all the PE routers participating in a VPLS routing instance. In contrast, a Layer 2 VPN forwards packets in point-to-point fashion only. The paths carrying VPLS traffic between each PE router participating in a routing instance are signaled using BGP.

This topic covers:

Supported Devices and Interfaces

VPLS allows a J-series Services Router to act as a PE router. Besides configuring a VPLS routing instance on a Services Router, you must also configure the interfaces that will carry VPLS traffic between the PE router and CE devices. VPLS traffic to CE devices are supported on the following J-series Services Router interfaces and PIMs:

- Built-in Ethernet ports on front panel
- Gigabit Ethernet uPIMs
- Gigabit Ethernet ePIMs
- Fast Ethernet PIMs
- Fast Ethernet ePIMs

NOTE: Ports on uPIMs and ePIMs must be in routing mode before you can configure the corresponding interfaces for VPLS.

VPLS Terms

Before configuring VPLS, become familiar with the terms defined in Table 151 on page 476.

Table 151: VPLS Terms

Term	Definition
Customer edge (CE) devices	Routers or switches located at the customer site that connect to the provider's network. CE devices are typically IP routers, but could also be an Asynchronous Transfer Mode (ATM), Frame Relay, or Ethernet switch.
Class of service (CoS)	Method of classifying traffic on a packet-by-packet basis using information in the type-of-service (ToS) byte to provide different service levels to different traffic.

Table 151: VPLS Terms *(continued)*

Term	Definition
Label switched path (LSP)	Sequence of routers that cooperatively perform MPLS operations for a packet stream. The first router in an LSP is called the *ingress router* and the last router in the path is called the *egress router*. An LSP is a point-to-point, half-duplex connection from the ingress router to the egress router. (The ingress and egress routers cannot be the same router.)
Media access control	In the OSI seven-layer networking model defined by the IEEE, MAC is the lower sublayer of the data link layer. The MAC sublayer governs protocol access to the physical network medium. By using the MAC addresses that are assigned to all ports on a router, multiple devices on the same physical link can uniquely identify one another at the data link layer.
Multiprotocol Label Switching (MPLS)	Mechanism for engineering network traffic patterns that functions by assigning to network packets short labels that describe how to forward them through the network. Also called *label switching*.
Point-to-multipoint LSP	RSVP-signaled LSP with a single source and multiple destinations.
Provider edge (PE) router	A router in the service provider's network that is connected to a customer edge (CE) device and participates in a virtual private network (VPN or VPLS).
Quality of service (QoS)	Performance, such as transmission rates and error rates, of a communications channel or system.
Virtual private LAN service (VPLS)	An Ethernet-based multipoint-to-multipoint Layer 2 VPN service used for interconnecting multiple Ethernet LANs across an MPLS backbone. VPLS is specified in IETF RFC 4761, *Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling*.

Related Topics

- Understanding VPLS on page 477
- VPLS on a PE Router Configuration Overview on page 484

Understanding VPLS

This topic describes VPLS functions on provider edge (PE) routers.

Before You Begin
For background information, read "VPLS Overview" on page 475.

Because a VPLS carries Ethernet traffic across a service provider network, it must mimic an Ethernet network in some ways. When a PE router configured with a VPLS routing instance receives a packet from a customer edge (CE) device, it first determines whether it has the destination of the VPLS packet in the appropriate routing table. If it does, it forwards the packet to the appropriate PE router or CE device. If it does not, it broadcasts the packet to all other PE routers and CE devices

that are members of that VPLS routing instance. In both cases, the CE device receiving the packet must be different from the one sending the packet.

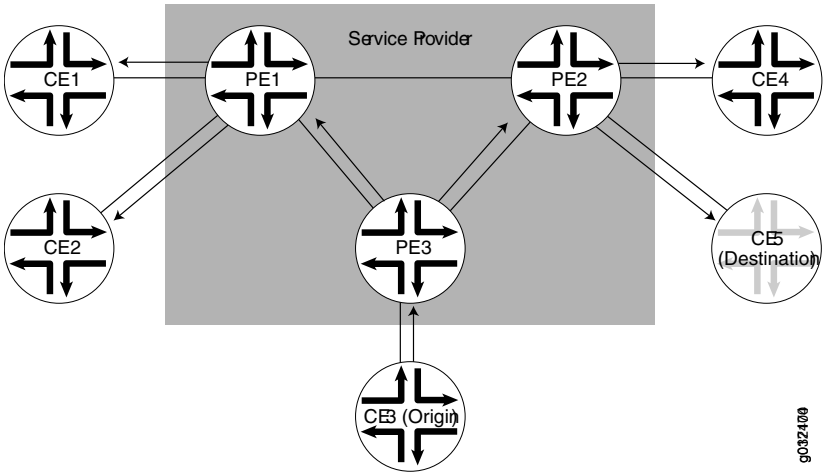
When a PE router receives a packet from another PE router, it first determines whether it has the destination of the VPLS packet in the appropriate routing table. If it does, the PE router either forwards the packet or drops it depending on whether the destination is a local or remote CE device:

- If the destination is a local CE device, the PE router forwards the packet to it.
- If the destination is a remote CE device (connected to another PE router), the PE router discards the packet.

If the PE router cannot determine the destination of the VPLS packet, it floods the packet to all attached CE devices.

Figure 80 on page 478 illustrates this process.

Figure 80: Flooding a Packet with an Unknown Destination



A VPLS can be directly connected to an Ethernet switch. Layer 2 information gathered by an Ethernet switch, for example, MAC addresses and interface ports, is included in the VPLS routing instance table.

An MPLS label-switched interface (LSI) label is used as the inner label for VPLS. This label maps to a VPLS routing instance. On the PE router, the LSI label is stripped and then mapped to a logical LSI interface. The Layer 2 Ethernet frame is then forwarded using the LSI interface to the correct VPLS routing instance.

One restriction on flooding behavior in VPLS is that traffic received from remote PE routers is never forwarded to other PE routers. This restriction helps prevent loops in the core network. However, if a CE Ethernet switch has two or more connections to the same PE router, you must enable the Spanning Tree Protocol (STP) on the CE switch to prevent loops.

The JUNOS software allows standard Bridge Protocol Data Unit (BPDU) frames to pass through emulated Layer 2 connections, such as those configured with Layer 2

VPNs, Layer 2 circuits, and VPLS routing instances. However, CE Ethernet switches that generate proprietary BPDU frames might not be able to run STP across Juniper Networks routing platforms configured for these emulated Layer 2 connections.

NOTE: Under certain circumstances, VPLS PE routers might duplicate an Internet Control Message Protocol (ICMP) reply from a CE device when a PE router has to flood an ICMP request because the destination MAC address has not yet been learned. The duplicate ICMP reply can be triggered when a CE device with promiscuous mode enabled is connected to a PE router. The PE router automatically floods the promiscuous mode enabled CE device, which then returns the ICMP request to the VPLS PE routers. The VPLS PE routers consider the ICMP request to be new and flood the request again, creating a duplicate ping reply.

Related Topics

- Understanding VPLS Routing Instances on page 479
- Understanding VPLS Interfaces on page 482
- VPLS Exceptions on J-Series Services Routers on page 484

Understanding VPLS Routing Instances

To configure VPLS functionality, you must enable VPLS support on the PE router. You must also configure PE routers to distribute routing information to the other PE routers in the VPLS and configure the circuits between the PE routers and the CE devices.

You create a VPLS routing instance on each PE router that is participating in the VPLS. The routing instance has the same name on each PE router. To configure the VPLS routing instance, you specify the following:

- Route distinguisher—Helps BGP distinguish between potentially identical network layer reachability information (NLRI) messages received from different VPLS instances. Each routing instance that you configure on a PE router must have a unique route distinguisher.
- Route target—Defines which route is part of a VPLS. A unique route target helps distinguish between different VPLS services on the same router.
- Site name—Provides unique name for the VPLS site.
- Site identifier—Provides unique numerical identifier for the VPLS site.
- Site range—Specifies total number of sites in the VPLS. The site range must be greater than the site identifier.
- Interface to the CE router—Specifies the physical interface to the CE router that carries VPLS traffic. The interface must be configured for a VPLS encapsulation type.

NOTE: In addition to the VPLS routing instance, you must configure MPLS label-switched paths (LSPs) between the PE routers, internal BGP (IBGP) sessions between the PE routers, and an interior gateway protocol (IGP) on the PE routers.

CAUTION: MPLS is disabled by default on J-series Services Routers and you must explicitly configure your router to allow MPLS traffic. However, when MPLS is enabled, all flow-based security features are deactivated and the router performs packet-based processing. Flow-based services such as security policies, zones, NAT, ALGs, chassis clustering, screens, firewall authentication, and IPSec VPNs are unavailable on the router. For more information on flow-based and packet-based processing, see the *JUNOS Software Security Configuration Guide*.

BGP Signaling

BGP is used to signal the paths between each of the PE routers participating in the VPLS routing instance. These paths carry VPLS traffic across the service provider's network between the VPLS sites.

NOTE: LDP signaling is not supported for the VPLS routing instance on J-series Services Routers.

VPLS Site Name and Site Identifier

When you configure BGP signaling for the VPLS routing instance, you must specify each VPLS site that has a connection to the router. For each VPLS site, you must configure a site name and site identifier (a numerical identifier between 1–65,534 that uniquely identifies the VPLS site).

Site Range

When you enable BGP signaling for the VPLS routing instance, you need to configure a site range. The site range specifies the total number of sites in the VPLS.

NOTE: The site range value must be greater than the largest site identifier.

Site Preference

You can specify the preference value advertised for a particular VPLS site. The site preference value is encoded in the BGP local preference attribute. When a PE router receives multiple advertisements with the same VPLS edge (VE) device identifier, the advertisement with the highest local preference value is preferred.

VPLS Routing Table

The VPLS routing table contains MAC addresses and interface information for both physical and virtual ports. You can configure the following characteristics for the table:

- Table size—You can modify the size of the VPLS MAC address table. The default table size is 512 MAC addresses; the minimum is 16 addresses, and the maximum is 65,536 addresses.

  If the MAC table limit is reached, new MAC addresses can no longer be added to the table. Eventually the oldest MAC addresses are removed from the MAC address table automatically. This frees space in the table, allowing new entries to be added. However, as long as the table is full, new MAC addresses are dropped.

  The interfaces affected include all of the interfaces within the VPLS routing instance, including the local interfaces and the LSI interfaces.

- Timeout interval—You can modify the timeout interval for the VPLS table. The default timeout interval is 300 seconds; the minimum is 10 seconds, and the maximum is 1,000,000 seconds. We recommend you configure longer values for small, stable VPLS networks and shorter values for large, dynamic VPLS networks. If the VPLS table does not receive any updates during the timeout interval, the router waits one additional interval before automatically clearing the MAC address entries from the VPLS table.

- Number of addresses learned from an interface—You can configure a limit on the number of MAC addresses learned by a VPLS routing instance by setting the MAC table size. The default is 512 addresses; the minimum is 16, and the maximum is 65,536 addresses. If the MAC table limit is reached, new MAC addresses can no longer be added to the table. Eventually the oldest MAC addresses are removed from the MAC address table automatically. This frees space in the table, allowing new entries to be added. However, as long as the table is full, new MAC addresses are dropped.

  Because this limit applies to each VPLS routing instance, the MAC addresses of a single interface can consume all the available space in the table, preventing the routing instance from acquiring addresses from other interfaces. You can limit the number of MAC addresses learned from all interfaces configured for a VPLS routing instance, as well as limit the number of MAC addresses learned from a specific interface.

  The MAC limit configured for an individual interface overrides the limit configured for all interfaces for the VPLS routing instance. Also, the table limit can override the limits configured for the interfaces.

  The MAC address limit applies only to interfaces to CE devices.

Trace Options

The following trace flags display operations associated with VPLS:

- all—All VPLS tracing options
- connections—VPLS connections (events and state changes)
- error—Error conditions
- nlri—VPLS advertisements received or sent using BGP
- route—Trace-routing information
- topology—VPLS topology changes caused by reconsideration or advertisements received from other PE routers using BGP

Related Topics

- Understanding VPLS Interfaces on page 482
- VPLS Exceptions on J-Series Services Routers on page 484

Understanding VPLS Interfaces

For each VPLS routing instance on a PE router, you specify which interfaces are to be used to carry VPLS traffic between the PE and CE devices.

Interface Name

Specify both the physical and logical portions of the interface name, in the following format:

physical.logical

For example, in ge-1/2/1.2, ge-1/2/1 is the physical portion of the interface name and 2 is the logical portion. If you do not specify the logical portion of the interface name, 0 is set by default. A logical interface can be associated with only one routing instance.

Encapsulation Type

The physical link-layer encapsulation type for VPLS interface can be one of the following:

- ethernet-vpls—Use Ethernet VPLS encapsulation on Ethernet interfaces that have VPLS enabled and that must accept packets carrying standard Tag Protocol Identifier (TPID) values.
- extended-vlan-vpls—Use extended virtual LAN (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. All VLAN IDs from 1 through 1023 are valid for VPLS VLANs on Fast Ethernet interfaces, and all VLAN IDs from 1 through 4094 are valid for VPLS VLANs on Gigabit Ethernet interfaces.
- 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. You must configure this encapsulation type on both the physical interface and the logical interface. VLAN IDs 1 through 511 are reserved for normal Ethernet VLANs, IDs 512 through 1023 are reserved for VPLS VLANs on Fast Ethernet interfaces, and IDs 512 through 4094 are reserved for VPLS VLANs on Gigabit Ethernet interfaces.

Flexible VLAN Tagging

For untagged packets to be accepted on an 802.1Q VLAN-tagged port, specify the native VLAN ID with the flexible VLAN tagging option. (No other flexible VLAN tagging features are supported.)

VLAN Rewrite

You can rewrite VLAN tags on VPLS interfaces. Rewriting VLAN tags allows you to use an additional (outer) VLAN tag to differentiate between CE devices that share a VLAN ID.

You can configure rewrite operations to stack (push), remove (pop), or rewrite (swap) tags on single-tagged frames. If a port is not configured for VLAN tagging, rewrite operations are not supported on any logical interface on that port.

You can configure the following VLAN rewrite operations:

- pop—Remove a VLAN tag from the top of the VLAN tag stack. The outer VLAN tag of the frame is removed.
- push—Add a new VLAN tag to the top of the VLAN stack. An outer VLAN tag is pushed in front of the existing VLAN tag.
- swap—Replace the VLAN tag at the top of the VLAN tag stack with a user-specified VLAN tag value.

You perform VLAN rewrite operations by applying input and output VLAN maps at the ingress and egress, respectively, of the interface. For incoming frames, use the input-vlan-map; for outgoing frames, use the output-vlan-map.

The VPLS implementation on Services Routers does not support dual-tagged frames. Therefore, VLAN rewrite operations are not supported on dual-tagged frames. VLAN rewrite operations such as pop-pop, pop-swap, push-push, swap-push, and swap-swap, which are supported on M-series and T-series routing platforms, are not supported on Services Routers.

Related Topics

- Understanding VPLS Routing Instances on page 479
- VPLS Exceptions on J-Series Services Routers on page 484

VPLS Exceptions on J-Series Services Routers

The VPLS implementation on a J-series Services Router is similar to VPLS implementations on M-series, T-series, and MX-series routers, with the following exceptions:

- Services Routers do not support aggregated Ethernet interfaces. Therefore, aggregated Ethernet interfaces between CE devices and PE routers are not supported for VPLS routing instances on Services Routers.
- VPLS routing instances on Services Routers use BGP to send signals to other PE routers. LDP signaling is not supported.
- VPLS multihoming, which allows connecting a CE device to multiple PE routers to provide redundant connectivity, is not supported on Services Routers.
- Services Routers do not support BGP mesh groups.
- Services Routers only support the following encapsulation types on VPLS interfaces that face CE devices: extended VLAN VPLS, Ethernet VPLS, and VLAN VPLS. Ethernet VPLS over ATM LLC encapsulation is not supported.
- Virtual ports are generated dynamically on a Tunnel Services PIC on some Juniper Networks routing platforms. Services Routers do not support Tunnel Services modules or virtual ports.
- The VPLS implementation on Services Routers does not support dual-tagged frames. Therefore, VLAN rewrite operations are not supported on dual-tagged frames. VLAN rewrite operations such as pop-pop, pop-swap, push-push, swap-push, and swap-swap, which are supported on M-series and T-series routing platforms, are not supported on Services Routers.
- Firewall filters for VPLS are not supported.

Related Topics

- Understanding VPLS Routing Instances on page 479
- Understanding VPLS Interfaces on page 482

VPLS on a PE Router Configuration Overview

Many configuration procedures for VPLS are identical to the procedures for Layer 2 and Layer 3 virtual private networks (VPNs), as described in “Configuring Virtual Private Networks” on page 439.

Before You Begin
For background information, read “Understanding VPLS” on page 477.

To prepare a provider edge (PE) router for VPLS, you must first configure the router to distribute routing information to other PE routers in the VPLS and configure the

circuits between the PE routers. The interior BGP (IBGP), MPLS, OSPF, and RSVP protocols are the basis for most Layer 2 VPN-related applications including VPLS.

On the PE router interface facing the customer edge (CE) device, you must specify a VPLS encapsulation type. The type of encapsulation depends on the interface type. Create the VPLS routing instance and add the interface. Specify the site range, ID number, and name for the VPLS routing instance.

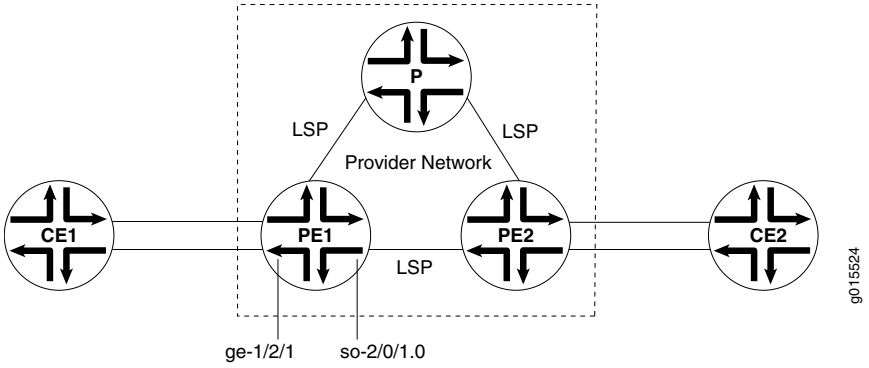
Each routing instance that you configure on a PE router must have a unique route distinguisher associated with it. VPN routing instances need a route distinguisher to help BGP identify overlapping network layer reachability information (NLRI) messages from different VPNs.

You can use either J-Web or the CLI configuration editor to configure VPLS on a PE router.

Sample VPLS Topology

Figure 81 on page 485 shows the overview of a basic VPLS topology for the sample configurations in this chapter.

Figure 81: Basic VPLS Topology



In this sample, the PE routers use the same autonomous system (AS). Within the AS, routing information is communicated through an interior gateway protocol (IGP). Outside the AS, routing information is shared with other ASs through BGP. The PE routers must use the same signaling protocols to communicate.

On the CE device interface that faces the PE router, you must specify inet (for IPv4), and include the IP address.

Related Topics

- Configuring Routing Options on the VPLS PE Router on page 486
- Configuring Routing Interfaces on the VPLS PE Router on page 487
- Configuring MPLS on the VPLS PE Router on page 489
- Configuring RSVP on the VPLS PE Router on page 490

Configuring Routing Options on the VPLS PE Router

For each router involved in the VPLS, specify the router ID and autonomous system (AS) number. In this sample, PE1 and PE2 use the same AS number (100).

Before You Begin

For background information, read "VPLS on a PE Router Configuration Overview" on page 484.

You can use either J-Web or the CLI configuration editor to configure VPLS on a PE router.

This topic covers:

J-Web Configuration

To configure the router ID on the VPLS PE router:

1. Select **Configuration > View and Edit > Edit Configuration**.

   The Configuration page appears.
2. Select **Routing**.
3. In the Router ID box, type **10.255.7.168**.

To configure the AS number on the VPLS PE router:

1. In the As number box, type **100**.
2. Click **OK**.

CLI Configuration

To configure the router ID on the VPLS PE router:

```
user@host# set routing-options router-id 10.255.7.168
```

To configure the AS number on the VPLS PE router:

```
user@host# set routing-options autonomous-system 100
```

Related Topics

- Configuring Routing Interfaces on the VPLS PE Router on page 487

Configuring Routing Interfaces on the VPLS PE Router

On the PE1 router, configure the loopback and the interface to the PE2 router (so-2/0/1 in this example).

Before You Begin
For background information, read "VPLS on a PE Router Configuration Overview" on page 484.

You can use either J-Web or the CLI configuration editor to configure VPLS on a PE router.

This topic covers:

- J-Web Configuration on page 487
- CLI Configuration on page 488
- Related Topics on page 489

J-Web Configuration

To configure the loopback interface on the VPLS PE router:

1. Select **Configuration > View and Edit > Edit Configuration**.

   The Configuration page appears.
2. Select **Interfaces**.
3. In the Interface name column, select **lo0**.
4. Under Unit, in the Interface unit number column, click **0**.
5. In the Family group, select **Inet** and click **Edit**.
6. Next to Address, click **Add new entry**.
7. In the Source box, type the address **127.0.0.1/32**.
8. Click **OK** to return to the Inet page.
9. Next to Address, click **Add new entry**.
10. In the Source box, type the address **10.255.7.168/32**.
11. Select **Primary**.

12. Click **OK** to return to the Family page.
13. In the Family group, select **Iso** and click **Edit**.
14. Next to Address, click **Add new entry**.
15. In the Source box, type the address **47.0005.80ff.f800.0000.0108.001.0102.5500.7168.00**.
16. Click **OK** to return to the Family page.
17. In the Family group, select **inet6** and click **Edit**.
18. Next to Address, click **Add new entry**.
19. In the Source box, type the address address **abcd::10:255:7:168/128**.
20. Select **Primary**.
21. Click **OK** to return to the Family page.
22. Click **OK** to return to the Unit page.
23. Click **OK** to return to the Interface page.
24. Click **OK** to return to the Interfaces page.

To configure the interface to the PE2 router on the VPLS PE router:

1. On the Interfaces page, select the interface to the PE2 router (**so-2/0/1** in this example) from the Interface name column.
2. Under Unit, in the Interface unit number column, click **0**.
3. In the Family group, select **Inet** and click **Edit**.
4. Next to Address, click **Add new entry**.
5. In the Source box, type the address **10.1.1.1/30**.
6. Click **OK** to return to the Unit page.
7. On the Unit page, select **Mpls** in the Family group.
8. Click **OK** to return to the Interfaces page.

CLI Configuration

To configure the loopback interface on the VPLS PE router:

```
user@host# set interfaces lo0 unit 0 family inet address 127.0.0.1/32
user@host# set interfaces lo0 unit 0 family inet address 10.255.7.168/32 primary
user@host# set interfaces lo0 unit 0 family iso address
    47.0005.80ff.f800.0000.0108.001.0102.5500.7168.00
user@host# set interfaces lo0 unit 0 family inet6 address abcd::10:255:7:168/128
    primary
```

To configure the interface to the PE2 router on the VPLS PE router:

```
user@host# set interfaces so-2/0/1 unit 0 family inet address 10.1.1.1/30
user@host# set interfaces so-2/0/1 unit 0 family mpls
```

Related Topics

- Configuring MPLS on the VPLS PE Router on page 489
- Configuring RSVP on the VPLS PE Router on page 490
- Configuring BGP on the VPLS PE Router on page 492
- Configuring OSPF on the VPLS PE Router on page 493

Configuring MPLS on the VPLS PE Router

Configure MPLS on the PE1 router to advertise the Layer 2 VPN interface that communicates with the PE2 router.

Before You Begin
For background information, read "VPLS on a PE Router Configuration Overview" on page 484.

CAUTION: MPLS is disabled by default on J-series Services Routers and you must explicitly configure your router to allow MPLS traffic. However, when MPLS is enabled, all flow-based security features are deactivated and the router performs packet-based processing. Flow-based services such as security policies, zones, NAT, ALGs, chassis clustering, screens, firewall authentication, and IPSec VPNs are unavailable on the router. For more information on flow-based and packet-based processing, see the *JUNOS Software Security Configuration Guide*.

You can use either J-Web or the CLI configuration editor to configure MPLS on the VPLS PE router.

This topic covers:

- J-Web Configuration on page 489
- CLI Configuration on page 490
- Related Topics on page 490

J-Web Configuration

To configure the interface to the PE2 router for MPLS:

1. Select **Configuration > View and Edit > Edit Configuration**.

   The Configuration page appears.
2. Select **Protocols**.
3. Next to Mpls, click **Configure** or **Edit**.
4. Next to Interface, click **Add new entry**.

5. In the Interface name box, type **so-2/0/1.0**.
6. Click **OK**.

To configure the loopback for MPLS:

1. In the Mpls page, click **Add new entry** next to Interface.
2. In the Interface name box, type **lo0.0**.
3. Click **OK**.

To configure the path to destination 10.255.7.164:

1. In the Mpls page, click **Add new entry** next to Label switched path.
2. In the Path name box, type **chelsea-sagar**.
3. In the To box, type **10.255.7.164**.
4. Click **OK**.

CLI Configuration

To configure the interface to the PE2 router for MPLS:

```
user@host# set protocols mpls interface so-2/0/1.0
```

To configure the loopback for MPLS:

```
user@host# set protocols mpls interface lo0.0
```

To configure the path to destination 10.255.7.164:

```
user@host# set protocols mpls label-switched-path chelsea-sagar to 10.255.7.164
```

Related Topics

- Configuring RSVP on the VPLS PE Router on page 490
- Configuring BGP on the VPLS PE Router on page 492
- Configuring OSPF on the VPLS PE Router on page 493

Configuring RSVP on the VPLS PE Router

Enable RSVP for all connections that participate in the label-switched path (LSP) on the PE1 router.

Before You Begin
For background information, read “VPLS on a PE Router Configuration Overview” on page 484.

You can use either J-Web or the CLI configuration editor to configure RSVP on the VPLS PE router.

This topic covers:

J-Web Configuration

To configure the interface to the PE2 router for RSVP:

1. Select **Configuration > View and Edit > Edit Configuration**. The Configuration page appears.
2. Select **Protocols**.
3. Next to Rsvp, click **Configure** or **Edit**.
4. Next to Interface, click **Add new entry**.
5. In the Interface name box, type **so-2/0/1.0**.
6. Click **OK**.

To configure the loopback interface for RSVP:

1. In the Rsvp page, click **Add new entry** next to Interface.
2. In the Interface name box, type **lo0.0**.
3. Click **OK**.
4. In the Rsvp page, click **OK**.

CLI Configuration

To configure the interface to the PE2 router for RSVP:

```
user@host# set protocols rsvp interface so-2/0/1.0
```

To configure the loopback interface for RSVP:

```
user@host#set protocols rsvp interface lo0.0
```

Related Topics

Configuring BGP on the VPLS PE Router

You configure an internal BGP (IBGP) session between PE routers so that the routers can exchange information about routes originating and terminating in the VPLS. The PE routers use this information to determine which labels to use for traffic destined for remote sites.

Before You Begin

For background information, read "VPLS on a PE Router Configuration Overview" on page 484.

You can use either J-Web or the CLI configuration editor to configure BGP on the VPLS PE1 router.

This topic covers:

J-Web Configuration

To configure the BGP internal group on the VPLS PE router:

1. Select **Configuration > View and Edit > Edit Configuration**. The Configuration page appears.
2. Select **Protocols**.
3. Next to Bgp, click **Configure** or **Edit**.
4. Next to Group, click **Add new entry**.
5. In the Group name box, type **ibgp**.
6. In the Local address box, type **10.255.7.168**.
7. From the Type list, select **internal**.
8. Next to Neighbor, click **Add new entry**.
9. In the Address box, type **10.255.7.164**.
10. Click **OK**.

To configure the BGP family L2vpn and specify NLRI signaling:

1. In the Group page, under Family, select **Configure** next to L2vpn.
2. Select **Signaling**.
3. Click **OK** to return to the Family page.
4. Click **OK** to return to the Group page.

CLI Configuration

To configure the BGP internal group on the VPLS PE router:

```
user@host# set protocols bgp group ibgp type internal local-address 10.255.7.168
  neighbor 10.255.7.164
```

To configure the BGP family L2vpn and specify NLRI signaling:

```
user@host# set protocols bgp family l2vpn signaling
```

Related Topics

- Configuring MPLS on the VPLS PE Router on page 489
- Configuring RSVP on the VPLS PE Router on page 490
- Configuring OSPF on the VPLS PE Router on page 493

Configuring OSPF on the VPLS PE Router

The PE routers exchange routing information using an IGP such as OSPF.

Before You Begin
For background information, read "VPLS on a PE Router Configuration Overview" on page 484.

You can use either J-Web or the CLI configuration editor to configure OSPF on the VPLS PE router.

This topic covers:

- J-Web Configuration on page 493
- CLI Configuration on page 494
- Related Topics on page 494

J-Web Configuration

To configure OSPF area 0.0.0.0 on the VPLS PE router:

1. Select **Configuration > View and Edit > Edit Configuration**.

   The Configuration page appears.
2. Expand **Protocols**.
3. Select **ospf**.
4. Next to Area, click **Add new entry**.
5. In the Area ID, type **0.0.0.0**.
6. Click **OK**.

7. Next to Area select the area **0.0.0.0**.
8. Next to Interface, click **Add new entry**.
9. In the Interface name box, type **so-2/0/1.0**.
10. Click **OK** to return to the Area page.
11. Next to Interfaces, click **Add new entry**.
12. In the Interface name box, type **lo0.0**.
13. Click **OK** to return to the Area page.

To configure traffic engineering for OSPF:

1. Click **OK** to return to the Ospf page.
2. Select **Traffic engineering**.
3. Click **OK**.

CLI Configuration

To configure OSPF area 0.0.0.0 on the VPLS PE router:

```
user@host# set protocols ospf area 0.0.0.0 interface so-2/0/1.0
user@host# set protocols ospf area 0.0.0.0 interface lo0.0
```

To configure traffic engineering for OSPF:

```
user@host# set protocols ospf traffic-engineering
```

Related Topics

- Configuring MPLS on the VPLS PE Router on page 489
- Configuring RSVP on the VPLS PE Router on page 490
- Configuring BGP on the VPLS PE Router on page 492

Configuring the Interface to the CE Device

On the PE1 router, configure the interface connected to the CE device to include VPLS encapsulation.

VPLS traffic to CE devices are supported on the following Services Router PIMs:

- Gigabit Ethernet uPIMs
- Gigabit Ethernet ePIMs
- Fast Ethernet PIMs
- Fast Ethernet ePIMs

NOTE: Ports on uPIMs and ePIMs must be in routing mode before you can configure the corresponding interfaces for VPLS.

Before You Begin

For background information, read "VPLS on a PE Router Configuration Overview" on page 484.

You can use either J-Web or the CLI configuration editor to configure an interface to the CE device for VPLS.

This topic covers:

J-Web Configuration

To configure VPLS encapsulation for the interface facing the CE router:

1. Select **Configuration > View and Edit > Edit Configuration**.

   The Configuration page appears.
2. Select **Interfaces**.
3. In the Interface name column, select the interface facing the CE1 router (**ge-1/2/1** in this example).
4. Select the encapsulation type **ethernet-vpls** from the encapsulation list.

To configure the interface for the vpls family group:

1. Under Unit, in the Interface unit number column, select **0**.
2. In the Family group, select **vpls**.
3. Click **OK**.

CLI Configuration

To configure VPLS encapsulation for the interface facing the CE router:

```
user@host# set interfaces ge-1/2/1 encapsulation ethernet-vpls
```

To configure the interface for the vpls family group:

```
user@host# set interfaces ge-1/2/1 unit 0 family vpls
```

Related Topics

- Configuring the VPLS Routing Instance on page 496

Configuring the VPLS Routing Instance

Create a VPLS routing instance on each PE router that is participating in the VPLS. The routing instance has the same name on each PE router.

NOTE: You must specify no-tunnel-services in the VPLS routing instance configuration, as Services Routers do not support tunnel PICs.

Before You Begin

For background information, read:

- VPLS on a PE Router Configuration Overview on page 484
- Configuring the Interface to the CE Device on page 494

You can use either J-Web or the CLI configuration editor to configure a VPLS routing instance.

This topic covers:

- J-Web Configuration on page 496
- CLI Configuration on page 497
- Related Topics on page 498

J-Web Configuration

To create a VPLS routing instance:

1. Select **Configuration > View and Edit > Edit Configuration**.

   The Configuration page appears.
2. Expand **Routing instances**.
3. Select **instance**.

   The Instance page appears.
4. Next to Instance, click **Add New Entry**.
5. In the Instance name box, type **green**.
6. For Instance type, select **vpls**.

To configure the VPLS identifer and range for the VPLS routing instance:

1. Next to Protocols, click **Configure**.
2. For L2vpn or vpls, select **vpls**.
3. Next to Vpls, click **Configure**.
4. In the Site range box, enter **10**.
5. For Tunnel services choice, select **No tunnel services**.
6. Next to Site, click **Add New Entry**. For Site Name, enter **R3**.
7. For Site identifier mode, select Site identifier. In the Site identifier box, enter **2**.
8. Click **OK**.
9. Click **OK** to return to the Protocols page.
10. Click **OK** to return to the Instance page.

To configure the route distinguisher and route target for the VPLS routing instance:

1. Next to Vrf target, click **Configure**.
2. In the Community box, enter **11111:1**. Click **OK**.

   The Routing Instance page reappears.
3. In the Route distinguisher box, enter **10.255.7.1:1**.

To specify the VPLS interface to the CE router:

1. Next to Interface, click **Add New Entry**.
2. In the Interface name box, enter **ge-1/2/1.0**.
3. Click **OK**.

CLI Configuration

To create a VPLS routing instance:

```
user@host# set routing-instances green instance-type vpls
```

To configure the VPLS site identifer and range for the VPLS routing instance:

```
user@host# set routing-instances green protocols vpls site-range 10 site R3
  site-identifier 2
```

To configure the no-tunnel-services option for the VPLS routing instance green:

```
user@host# set routing-instances green protocols vpls no-tunnel-services
```

To configure the route distinguisher and route target for the VPLS routing instance:

```
user@host# set routing-instances green route-distinguisher 10.255.7.1:1
user@host# set routing-instances green vrf-target target:11111:1
```

To specify the VPLS interface to the CE router:

```
user@host# set routing-instances green instance-type vpls interface ge-1/2/1.0
```

Related Topics

- Configuring an Ethernet Switch as the CE Device on page 498

Configuring an Ethernet Switch as the CE Device

For VPLS configurations, the CE device does not necessarily need to be a router. You can link the PE routers directly to Ethernet switches. However, be aware of the following configuration issues:

- When you configure VPLS routing instances and establish two or more connections between a CE Ethernet switch and a PE router, you must enable the Spanning Tree Protocol (STP) on the switch to prevent loops.
- The JUNOS software allows standard Bridge Protocol Data Unit (BPDU) frames to pass through emulated Layer 2 connections, such as those configured with Layer 2 VPNs, Layer 2 circuits, and VPLS instances. However, CE Ethernet switches that generate proprietary BPDU frames might not be able to run STP across Juniper Networks routing platforms configured for these emulated Layer 2 connections.

Part 5
Configuring Routing Policies and Stateless Firewall Filters

Chapter 24

Configuring Routing Policies

Use routing policies as filters to control the information from routing protocols that a Juniper Networks device imports into its routing table and the information that the router exports (advertises) to its neighbors. To create a routing policy, you configure criteria against which routes are compared, and the action that is performed if the criteria are met.

You use either the J-Web configuration editor or CLI configuration editor to configure a routing policy.

This chapter contains the following topics. For more information about routing policies, see the *JUNOS Policy Framework Configuration Guide*.

For information about security policies and stateful firewalls, see the *JUNOS Software Security Configuration Guide*.

Routing Policies

This section contains the following topics:

Routing Policy Overview

Routing protocols send information about routes to a router's neighbors. This information is processed and used to create routing tables, which are then distilled into forwarding tables. Routing policies control the flow of information between the routing protocols and the routing tables and between the routing tables and the forwarding tables. Using policies, you can determine which routes are advertised, specify which routes are imported into the routing table, and modify routes to control which routes are added to the forwarding table. For more information, see the *JUNOS Policy Framework Configuration Guide*.

Routing policies are made up of one or more terms, each of which contains a set of match conditions and a set of actions. Match conditions are criteria that a route must match before the actions can be applied. If a route matches all criteria, one or more actions are applied to the route. These actions specify whether to accept or reject the route, control how a series of policies are evaluated, and manipulate the characteristics associated with a route.

Routing Policy Terms

Generally, a router compares a route against the match conditions of each term in a routing policy, starting with the first and moving through the terms in the order in which they are defined, until a match is made and an explicitly configured or default action of accept or reject is taken. If none of the terms in the policy match the route, the router compares the route against the next policy, and so on, until either an action is taken or the default policy is evaluated.

Default and Final Actions

If none of the terms' match conditions evaluate to true, the final action is executed. The final action is defined in an unnamed term. Additionally, you can define a default action (either accept or reject) that overrides any action intrinsic to the protocol.

Applying Routing Policies

Once a policy is created, it must be applied before it is active. You apply routing policies using the import and export statements at the **Protocols >** ***protocol-name*** level in the configuration hierarchy.

In the import statement, you list the name of the routing policy to be evaluated when routes are imported into the routing table from the routing protocol.

In the export statement, you list the name of the routing policy to be evaluated when routes are being exported from the routing table into a dynamic routing protocol. Only active routes are exported from the routing table.

To specify more than one policy and create a policy chain, you list the policies using a space as a separator. If multiple policies are specified, the policies are evaluated in the order in which they are specified. As soon as an accept or reject action is executed, the policy chain evaluation ends.

Routing Policy Match Conditions

A match condition defines the criteria that a route must match for an action to take place. Each term can have one or more match conditions. If a route matches all the match conditions for a particular term, the actions defined for that term are processed.

Each term can consist of two statements, to and from, that define match conditions:

- In the from statement, you define the criteria that an *incoming* route must match. You can specify one or more match conditions. If you specify more than one, all conditions must match the route for a match to occur.

- In the to statement, you define the criteria that an *outgoing* route must match. You can specify one or more match conditions. If you specify more than one, all conditions must match the route for a match to occur.

The order of match conditions in a term is not important, because a route must match all match conditions in a term for an action to be taken.

Table 152 on page 503 summarizes key routing policy match conditions.

Table 152: Summary of Key Routing Policy Match Conditions

Match Condition	Description
aggregate-contributor	Matches routes that are contributing to a configured aggregate. This match condition can be used to suppress a contributor in an aggregate route.
area *area-id*	Matches a route learned from the specified OSPF area during the exporting of OSPF routes into other protocols.
as-path *name*	Matches the name of an autonomous systems (AS) path regular expression. BGP routes whose AS path matches the regular expression are processed.
color *preference*	Matches a color value. You can specify preference values that are finer-grained than those specified in the *preference* match conditions. The color value can be a number from 0 through 4,294,967,295 ($2^{32} - 1$). A lower number indicates a more preferred route.
community	Matches the name of one or more communities. If you list more than one name, only one name needs to match for a match to occur. (The matching is effectively a logical OR operation.)
external [type *metric-type*]	Matches external OSPF routes, including routes exported from one level to another. In this match condition, type is an optional keyword. The metric-type value can be either 1 or 2. When you do not specify type, this condition matches all external routes.
interface *interface-name*	Matches the name or IP address of one or more router interfaces. Use this condition with protocols that are interface-specific. For example, do not use this condition with internal BGP (IBGP). Depending on where the policy is applied, this match condition matches routes learned from or advertised through the specified interface.
internal	Matches a routing policy against the internal flag for simplified next-hop self policies.
level *level*	Matches the IS-IS level. Routes that are from the specified level or are being advertised to the specified level are processed.
local-preference *value*	Matches a BGP local preference attribute. The preference value can be from 0 through 4,294,967,295 ($2^{32} - 1$).
metric *metric* metric2 *metric*	Matches a metric value. The metric value corresponds to the multiple exit discriminator (MED), and metric2 corresponds to the interior gateway protocol (IGP) metric if the BGP next hop runs back through another route.

Table 152: Summary of Key Routing Policy Match Conditions *(continued)*

Match Condition	Description
neighbor *address*	Matches the address of one or more neighbors (peers). For BGP export policies, the address can be for a directly connected or indirectly connected peer. For all other protocols, the address is for the neighbor from which the advertisement is received.
next-hop *address*	Matches the next-hop address or addresses specified in the routing information for a particular route. For BGP routes, matches are performed against each protocol next hop.
origin *value*	Matches the BGP origin attribute, which is the origin of the AS path information. The value can be one of the following: ■ egp—Path information originated from another AS. ■ igp—Path information originated from within the local AS. ■ incomplete—Path information was learned by some other means.
preference *preference* preference2 *preference*	Matches the preference value. You can specify a primary preference value (preference) and a secondary preference value (preference2). The preference value can be a number from 0 through 4,294,967,295 ($2^{32} - 1$). A lower number indicates a more preferred route.
protocol *protocol*	Matches the name of the protocol from which the route was learned or to which the route is being advertised. It can be one of the following: aggregate, bgp, direct, dvmrp, isis, local, ospf, pim-dense, pim-sparse, rip, ripng, or static.
route-type *value*	Matches the type of route. The value can be either external or internal.

Routing Policy Actions

An action defines what the router does with the route when the route matches all the match conditions in the from and to statements for a particular term. If a term does not have from and to statements, all routes are considered to match and the actions apply to all routes.

Each term can have one or more of the following types of actions. The actions are configured under the then statement.

- Flow control actions, which affect whether to accept or reject the route and whether to evaluate the next term or routing policy
- Actions that manipulate route characteristics
- Trace action, which logs route matches

Table 153 on page 505 summarizes the routing policy actions.

If you do not specify an action, one of the following results occurs:

- The next term in the routing policy, if one exists, is evaluated.

- If the routing policy has no more terms, the next routing policy, if one exists, is evaluated.
- If there are no more terms or routing policies, the accept or reject action specified by the default policy is executed.

Table 153: Summary of Key Routing Policy Actions

Action	Description
Flow Control Actions	These actions control the flow of routing information into and out of the routing table.
accept	Accepts the route and propagates it. After a route is accepted, no other terms in the routing policy and no other routing policies are evaluated.
reject	Rejects the route and does not propagate it. After a route is rejected, no other terms in the routing policy and no other routing policies are evaluated.
next term	Skips to and evaluates the next term in the same routing policy. Any accept or reject action specified in the then statement is ignored. Any actions specified in the then statement that manipulate route characteristics are applied to the route.
next policy	Skips to and evaluates the next routing policy. Any accept or reject action specified in the then statement is ignored. Any actions specified in the then statement that manipulate route characteristics are applied to the route.
Route Manipulation Actions	These actions manipulate the route characteristics.
as-path-prepend *as-path*	Appends one or more autonomous system (AS) numbers at the beginning of the AS path. If you are specifying more than one AS number, include the numbers in quotation marks. The AS numbers are added after the local AS number has been added to the path. This action adds AS numbers to AS sequences only, not to AS sets. If the existing AS path begins with a confederation sequence or set, the appended AS numbers are placed within a confederation sequence. Otherwise, the appended AS numbers are placed with a nonconfederation sequence.
as-path-expand last-as count *n*	Extracts the last AS number in the existing AS path and appends that AS number to the beginning of the AS path *n* times. Replace *n* with a number from 1 through 32. The AS numbers are added after the local AS number has been added to the path. This action adds AS numbers to AS sequences only, not to AS sets. If the existing AS path begins with a confederation sequence or set, the appended AS numbers are placed within a confederation sequence. Otherwise, the appended AS numbers are placed with a nonconfederation sequence.
class *class-name*	Applies the specified class-of-service (CoS) parameters to routes installed into the routing table.
color *preference* color2 *preference*	Sets the preference value to the specified value. The color and color2 preference values can be a number from 0 through 4,294,967,295 ($2^{32} - 1$). A lower number indicates a more preferred route.
damping *name*	Applies the specified route-damping parameters to the route. These parameters override BGP's default damping parameters. This action is useful only in import policies.

Table 153: Summary of Key Routing Policy Actions *(continued)*

Action	Description
local-preference *value*	Sets the BGP local preference attribute. The preference can be a number from 0 through 4,294,967,295 ($2^{32} - 1$).
metric *metric* metric2 *metric* metric3 *metric* metric4 *metric*	Sets the metric. You can specify up to four metric values, starting with metric (for the first metric value) and continuing with metric2, metric3, and metric4. For BGP routes, metric corresponds to the MED, and metric2 corresponds to the IGP metric if the BGP next hop loops through another router.
next-hop *address*	Sets the next hop. If you specify *address* as self, the next-hop address is replaced by one of the local router's addresses. The advertising protocol determines which address to use.

Before You Begin

Before you begin configuring a routing policy, complete the following tasks:

- If you do not already have a basic understanding of routing policies, read "Routing Policies" on page 501.
- Determine what you want to accomplish with the policy, and thoroughly understand how to achieve your goal using the various match conditions and actions.
- Make certain that you understand the default policies and actions for the policy you are configuring.
- Configure an interface on the router. See "Configuring Ethernet, DS1, DS3, and Serial Interfaces" on page 73.
- Configure an Interior Gateway Protocol (IGP) and Border Gateway Protocol (BGP), if necessary. See "Configuring BGP Sessions" on page 387.
- Configure the router interface to reject or accept routes, if necessary. See "Configuring Stateless Firewall Filters (ACLs)" on page 521.
- Configure static routes, if necessary. See "Configuring Static Routes" on page 333.

Configuring a Routing Policy with a Configuration Editor

A routing policy has a major impact on the flow of routing information or packets within and through the device. The match conditions and actions allow you to configure a customized policy to fit your needs.

To configure a routing policy, you must perform the following tasks marked *(Required)*. Perform additional tasks as needed for your router. For information about using the

J-Web and CLI configuration editors, see the *J-Web Interface User Guide* and the *JUNOS CLI User Guide*.

Configuring the Policy Name (Required)

Each routing policy is identified by a policy name. The name can contain letters, numbers, and hyphens (-) and can be up to 255 characters long. To include spaces in the name, enclose the entire name in double quotation marks.

Each routing policy name must be unique within a configuration.

To configure the policy name:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 154 on page 507.
3. Go on to "Configuring a Policy Term (Required)" on page 507.

Table 154: Configuring the Policy Name

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Policy statement** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**. 3. Next to Policy statement, click **Add new entry**.	From the [edit] hierarchy level, enter edit policy-options
Enter the policy name—for example, policy1.	1. In the Policy name box, type policy1. 2. Click **OK**.	Type the policy-name value: set policy-statement policy1

Configuring a Policy Term (Required)

Each routing policy term is identified by a term name. The name can contain letters, numbers, and hyphens (-) and can be up to 255 characters long. To include spaces in the name, enclose the entire name in double quotation marks.

To configure a policy term:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 155 on page 508.
3. If you are finished configuring the router, commit the configuration.
4. To configure additional routing policy features, go on to one of the following procedures:
   - To remove useless routes, see "Rejecting Known Invalid Routes (Optional)" on page 508.
   - To advertise additional routes, see "Injecting OSPF Routes into the BGP Routing Table (Optional)" on page 510.
   - To create a forwarding class, see "Grouping Source and Destination Prefixes in a Forwarding Class (Optional)" on page 512.
   - To make a route less preferable to BGP, see "Configuring a Policy to Prepend the AS Path (Optional)" on page 513.
   - To suppress route information, see "Configuring Damping Parameters (Optional)" on page 516.

Table 155: Configuring a Policy Term

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Policy statement** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**. 3. Under Policy name, click **policy1**.	From the [edit] hierarchy level, enter edit policy-options policy-statement policy1
Create and name a policy term—for example, term1.	1. In the Term box, click **Add new entry**. 2. In the Term name box, type term1. 3. Click **OK**.	Create and name a policy term: set term term1

Rejecting Known Invalid Routes (Optional)

You can specify known invalid ("bad") routes to ignore by specifying matches on destination prefixes. When specifying a destination prefix, you can specify an exact match with a specific route, or a less precise match by using match types. You can configure either a common reject action that applies to the entire list, or an action associated with each prefix. Table 156 on page 509 lists route list match types.

Table 156: Route List Match Types

Match Type	Match Conditions
exact	The route shares the same most-significant bits (described by *prefix-length*), and *prefix-length* is equal to the route's prefix length.
longer	The route shares the same most-significant bits (described by *prefix-length*), and *prefix-length* is greater than the route's prefix length.
orlonger	The route shares the same most-significant bits (described by *prefix-length*), and *prefix-length* is equal to or greater than the route's prefix length.
prefix-length-range *prefix-length2-prefix-length3*	The route shares the same most-significant bits (described by *prefix-length*), and the route's prefix length falls between *prefix-length2* and *prefix-length3*, inclusive.
through *destination-prefix*	All the following are true: ■ The route shares the same most-significant bits (described by *prefix-length*) of the first destination prefix. ■ The route shares the same most-significant bits (described by *prefix-length*) of the second destination prefix for the number of bits in the prefix length. ■ The number of bits in the route's prefix length is less than or equal to the number of bits in the second prefix. You do not use the through match type in most routing policy configurations. For more information, see the *JUNOS Policy Framework Configuration Guide*.
upto *prefix-length2*	The route shares the same most-significant bits (described by *prefix-length*) and the route's prefix length falls between *prefix-length* and *prefix-length2*.

For example, you can create a policy named rejectpolicy1 to reject routes with a mask of /8 and greater (/8, /9, /10, and so on) that have the first 8 bits set to 0, and to accept routes less than 8 bits in length.

To create rejectpolicy1:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 157 on page 510.
3. If you are finished configuring the router, commit the configuration.
4. To configure additional routing policy features, go on to one of the following procedures:
   - To advertise additional routes, see "Injecting OSPF Routes into the BGP Routing Table (Optional)" on page 510.
   - To create a forwarding class, see "Grouping Source and Destination Prefixes in a Forwarding Class (Optional)" on page 512.

- To make a route less preferable to BGP, see "Configuring a Policy to Prepend the AS Path (Optional)" on page 513.
- To suppress route information, see "Configuring Damping Parameters (Optional)" on page 516.

Table 157: Creating a Policy to Reject Known Invalid Routes

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Policy statement** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**. 3. Next to Policy statement, click **Add new entry**.	From the [edit] hierarchy level, enter edit policy-options policy-statement
Create a rejection policy and term—for example, rejectpolicy1 and rejectterm1.	1. In the Policy name box, type rejectpolicy1. 2. Next to Term, click **Add new entry**. 3. In the Term name box, type rejectterm1.	Enter set rejectpolicy1 term rejectterm1
Specify the routes to accept—for example, routes with a mask of 0/0 up to /7.	1. Next to From, click **Configure**. 2. Next to Route filter, click **Add new entry**. 3. In the Address box, type 0/0. 4. From the Modifier list, select **Upto**. 5. In the Upto box, type /7. 6. From the Accept reject list, select **accept**. 7. Click **OK**.	Accept routes less than 8 bits in length: set from route-filter 0/0 up to /7 accept
Specify the routes to reject—for example, routes with a mask of /8 or greater.	1. Next to Route filter, click **Add new entry**. 2. In the Address box, type /8. 3. From the Modifier list, select **Orlonger**. 4. From the Accept reject list, select **reject**. 5. Click **OK**.	1. Specify routes less than 8 bits in length: set from route-filter /8 orlonger 2. Reject these routes: set then reject

Injecting OSPF Routes into the BGP Routing Table (Optional)

You can specify a match condition for policies based on protocols by naming a protocol from which the route is learned or to which the route is being advertised. You can specify one of the following protocols: aggregate, BGP, direct, DVMRP, IS-IS, local, OSPF, PIM-dense, PIM-sparse, RIP, or static

For example, you can inject or redistribute OSPF routes into the BGP routing table by creating a routing policy.

To create a routing policy named injectpolicy1 that redistributes OSPF routes from Area 1 only into BGP and does not advertise routes learned by BGP:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 158 on page 511.
3. If you are finished configuring the router, commit the configuration.
4. To configure additional routing policy features, go on to one of the following procedures:
   - To create a forwarding class, see "Grouping Source and Destination Prefixes in a Forwarding Class (Optional)" on page 512.
   - To make a route less preferable to BGP, see "Configuring a Policy to Prepend the AS Path (Optional)" on page 513.
   - To suppress route information, see "Configuring Damping Parameters (Optional)" on page 516.

Table 158: Creating a Policy to Inject OSPF Routes into BGP

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Policy statement** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**. 3. Next to Policy statement, click **Add new entry**.	From the [edit] hierarchy level, enter edit policy-options policy-statement
Create an injection policy and term—for example, injectpolicy1 and injectterm1.	1. In the Policy name box, type injectpolicy1. 2. Next to Term, click **Add new entry**. 3. In the Term name box, type injectterm1.	Enter set injectpolicy1 term injectterm1
Specify the OSPF routes.	1. In the From option, click **Configure**. 2. In the Protocol box, click **Add new entry**. 3. In the Value drop box, select **ospf**. 4. Click **OK**.	Specify the OSPF match condition: set from ospf
Specify the routes from a particular OSPF area—for example, Area 1.	1. In the Area box, type 1. 2. Click **OK**.	Specify Area 1 as a match condition: set from area 1

Table 158: Creating a Policy to Inject OSPF Routes into BGP *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Specify that the route is to be accepted if the previous conditions are matched. Set the default option to reject other OSPF routes.	1. Next to Then, click **Configure**. 2. From the Accept reject list, Select **accept**. 3. From the Default action list, Select **reject**. 4. Click **OK** until you return to the main Configuration page.	Specify the action to accept: set then accept
Navigate to the **Bgp** level in the configuration hierarchy.	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Bgp, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit protocols bgp
Apply the routing policy injectpolicy1 to BGP.	1. Next to Export, click **Add new entry**. 2. In the Value option, type injectpolicy1. 3. Click **OK**.	Specify the OSPF match condition: set export injectpolicy1

Grouping Source and Destination Prefixes in a Forwarding Class (Optional)

Create a forwarding class called forwarding-class1 that includes packets based on both the destination address and the source address in the packet.

To configure and apply the routing policy policy1, which you configured in Table 154 on page 507 and Table 155 on page 508, to group source and destination prefixes in a forwarding class:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 159 on page 513.
3. If you are finished configuring the router, commit the configuration.
4. To configure additional routing policy features, go on to one of the following procedures:
   - To make a route less preferable to BGP, see "Configuring a Policy to Prepend the AS Path (Optional)" on page 513.
   - To suppress route information, see "Configuring Damping Parameters (Optional)" on page 516.

Table 159: Creating a Policy to Group Source and Destination Prefixes in a Forwarding Class

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **term1** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**. 3. Under Policy name, click **policy1**. 4. Under Term name, click **term1**.	From the [edit] hierarchy level, enter edit policy-options policy-statement policy1 term term1
Specify the routes to include in the route filter. For example: ■ Source routes greater than or equal to 10.210.0.0/16 ■ Destination routes greater than or equal to 10.215.0.0/16	1. Next to From, click **Configure**. 2. Next to Route filter, click **Add new entry**. 3. In the Address box, type 10.210.0.0/16. 4. From the Modifier list, select Orlonger. 5. Click **OK** to return to the From page.	Specify the source routes for the route filter: set from route-filter 10.210.0.0/16 orlonger
	1. Next to Route filter, click **Add new entry**. 2. In the Address box, type 10.215.0.0/16. 3. From the Modifier list, select Orlonger. 4. Click **OK** until you return to the Term page.	Specify the destination routes for the route filter: set from route-filter 10.215.0.0/16 orlonger
Group the source and destination prefixes into a forwarding class—for example, forwarding-class1.	1. Next to Then, click **Configure**. 2. In the Forwarding class box, type forwarding-class1. 3. Click **OK**.	Specify the forwarding class name: set then forwarding class forwarding-class1
Navigate to the **Forwarding table** level in the configuration hierarchy.	1. On the main Configuration page next to Routing options, click **Configure** or **Edit**. 2. Next to Forwarding table, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit routing-options forwarding-table
Apply the policy1 policy to the forwarding table. The routing policy is evaluated when routes are being exported from the routing table into the forwarding table. Only active routes are exported from the routing table.	1. Next to Export, click **Add new entry**. 2. In the Value box, type policy1. 3. Click **OK**.	Specify the routing policy to apply: set export policy1 You can refer to the same routing policy one or more times in the same or a different export statement.

Configuring a Policy to Prepend the AS Path (Optional)

You can *prepend* or add one or more autonomous system (AS) numbers at the beginning of an AS path. The AS numbers are added after the local AS number has

been added to the path. Prepending an AS path makes a shorter AS path look longer and therefore less preferable to the Border Gateway Protocol (BGP).

For example, from AS 1, there are two equal paths (through AS 2 and AS 3) to reach AS 4. You might want packets from certain sources to use the path through AS 2. Therefore, you must make the path through AS 3 look less preferable so that BGP chooses the path through AS 2. In AS 1, you can prepend multiple AS numbers.

To create a routing policy prependpolicy1 that prepends multiple AS numbers:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 160 on page 514.
3. If you are finished configuring the router, commit the configuration.
4. To suppress route information, see "Configuring Damping Parameters (Optional)" on page 516.

Table 160: Creating a Policy to Prepend AS Numbers

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Policy statement** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**. 3. Next to Policy statement, click **Add new entry**.	From the [edit] hierarchy level, enter edit policy-options policy-statement
Create a prepend policy and term—for example, prependpolicy1 and prependterm1.	1. In the Policy name box, type prependpolicy1. 2. Next to Term, click **Add new entry**. 3. In the Term name box, type prependterm1.	Enter set prependpolicy1 term prependterm1

Table 160: Creating a Policy to Prepend AS Numbers *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Specify the routes to prepend AS numbers to. For example: ■ Routes greater than or equal to 172.16.0.0/12 ■ Routes greater than or equal to 192.168.0.0/16 ■ Routes greater than or equal to 10.0.0.0/8	1. Next to From, click **Configure**. 2. Next to Route filter, click **Add new entry**. 3. In the Value box, type 172.16.0.0/12. 4. From the Modifier list, select Orlonger. 5. Click **OK**.	Specify the first routes to prepend: set from route-filter 172.16.0.0/12 orlonger
	1. Next to From, click **Configure**. 2. Next to Route filter, click **Add new entry**. 3. In the Value box, type 192.168.0.0/16. 4. From the Modifier list, select Orlonger. 5. Click **OK**.	Specify the next routes to prepend: set from route-filter 192.168.0.0/16 orlonger
	1. Next to From, click **Configure**. 2. Next to Route filter, click **Add new entry**. 3. In the Value box, type 10.0.0.0/8. 4. From the Modifier list, select Orlonger. 5. Click **OK** until you return to the Term page.	Specify the last routes to prepend: set from route-filter 10.0.0.0/8 orlonger
Specify the AS numbers to prepend. Separate each AS number with a space—for example, 1 1 1 1.	1. Next to Then, click **Configure**. 2. In the AS path prepend box, type 1 1 1 1. 3. Click **OK**.	Specify the AS numbers to prepend, and enclose them inside double quotation marks: set then as-path-prepend “1 1 1 1”
Navigate to the **Bgp** level in the configuration hierarchy.	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Bgp, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit protocols bgp

Table 160: Creating a Policy to Prepend AS Numbers *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Apply the prependpolicy1 policy as an import policy for all BGP routes. The routing policy is evaluated when routes are being imported to the routing table.	1. Next to Import, click **Add new entry**. 2. In the Value box, type prependpolicy1. 3. Click **OK**.	Apply the policy: set import prependpolicy1 You can refer to the same routing policy one or more times in the same or a different import statement.

Configuring Damping Parameters (Optional)

Flap damping reduces the number of update messages by marking routes as ineligible for selection as the active or preferable route. Marking routes in this way leads to some delay, or *suppression*, in the propagation of route information, but the result is increased network stability. You typically apply flap damping to external BGP (EBGP) routes (routes in different ASs). You can also apply flap damping within a confederation, between confederation member ASs. Because routing consistency within an AS is important, do not apply flap damping to internal BGP (IBGP) routes. (If you do, it is ignored.)

You can specify one or more of the damping parameters described in Table 161 on page 516. If you do not specify a damping parameter, the default value of the parameter is used.

Table 161: Damping Parameters

Damping Parameter	Description	Default Value	Possible Values
half-life *minutes*	Decay half-life—Number of minutes after which an arbitrary value is halved if a route stays stable.	15 (minutes)	1 through 4
max-suppress *minutes*	Maximum hold-down time for a route, in minutes.	60 (minutes)	1 through 720
reuse	Reuse threshold—Arbitrary value below which a suppressed route can be used again.	750	1 through 20000
suppress	Cutoff (suppression) threshold—Arbitrary value above which a route can no longer be used or included in advertisements.	3000	1 through 20000

To change the default BGP flap damping values, you define actions by creating a named set of damping parameters and including it in a routing policy with the damping action. For the damping routing policy to work, you also must enable BGP route flap damping.

To configure damping with a policy named dampenpolicy1, perform these steps:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.

2. Perform the configuration tasks described in Table 162 on page 517.
3. If you are finished configuring the router, commit the configuration.

Table 162: Creating a Policy to Accept and Apply Damping on Routes

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Policy statement** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**. 3. Next to Policy statement, click **Add new entry**.	From the [edit] hierarchy level, enter edit policy-options policy-statement
Create a damping policy and term—for example, dampenpolicy1 and dampenterm1.	1. In the Policy name box, type dampenpolicy1. 2. Next to Term, click **Add new entry**. 3. In the Term name box, type dampenterm1.	Enter set dampenpolicy1 term dampenterm1

Table 162: Creating a Policy to Accept and Apply Damping on Routes *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Specify the routes to dampen and associate each group of routes with a group name. For example: ■ group1—Routes greater than or equal to 172.16.0.0/12 ■ group2—Routes greater than or equal to 192.168.0.0/16 ■ group3—Routes greater than or equal to 10.0.0.0/8	1. Next to From, click **Configure**. 2. Next to Route filter, click **Add new entry**. 3. In the Address box, type 172.16.0.0/12. 4. In the Damping box, type group1. 5. From the Modifier list, select Orlonger. 6. Click **OK**.	Specify the first routes to dampen: set from route-filter 172.16.0.0/12 orlonger damping group 1
	1. Next to Route filter, click **Add new entry**. 2. In the Address box, type 192.168.0.0/16. 3. In the Damping box, type group2. 4. From the Modifier list, select Orlonger. 5. Click **OK**.	Specify the next routes to dampen: set from route-filter 192.168.0.0/16 orlonger
	1. Next to Route filter, click **Add new entry**. 2. In the Address box, type 10.0.0.0/8. 3. In the Damping box, type group3. 4. From the Modifier list, select Orlonger. 5. Click **OK** until you return to the Policy options page.	Specify the last routes to dampen: set from route-filter 10.0.0.0/8 orlonger

Table 162: Creating a Policy to Accept and Apply Damping on Routes *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Create three damping parameter groups with different damping actions. For example: ■ group1—Increases the half-life to 30 minutes. All other parameters are left at their default values. ■ group2—Increases the half-life to 40 minutes, decreases the maximum hold-down time for a route to 45 minutes, increases the reuse value to 1000, and reduces the cutoff (suppression) threshold to 400. ■ group3—Disables route damping.	For *each* damping group: 1. Next to Damping, click **Add new entry**. 2. In the Damping object name box, type the name of a damping group—for example, group1. 3. In the Half life box, type the half-life duration, in minutes: ■ For group1—30 ■ For group2—40 4. In the Max suppress box, type the maximum hold-down time, in minutes: ■ For group1—60 (the default) ■ For group2—45 5. In the Reuse box, type the reuse threshold, for this damping group: ■ For group1—750 (the default) ■ For group2—1000 6. In the Suppress box, type the cutoff threshold, for this damping group: ■ For group1—3000 (the default) ■ For group2—400 7. To disable damping for the group3 damping group, select the **Disable** check box. 8. Click **OK** when you finish configuring each group.	Create and configure the damping parameter groups: edit damping group1 half-life 30 max-suppress 60 reuse 750 suppress 3000 edit damping group2 half-life 40 max-suppress 45 reuse 1000 suppress 400 edit damping group3 disable
Navigate to the **Bgp** level in the configuration hierarchy.	1. On the main Configuration page next to Protocols, click **Configure** or **Edit**. 2. Next to Bgp, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit protocols bgp
Enable damping.	1. Select the **Damping** check box. 2. Click **OK**.	Enable damping: set damping
Navigate to the **Neighbor** level in the configuration hierarchy, for the BGP neighbor to which you want to apply the damping policy—for example, the neighbor at IP address 172.16.15.14.	1. On the main Configuration page next to Protocols, click **Edit**. 2. Next to Bgp, click **Edit**. 3. Under Group name, click **groupA**. 4. Under Neighbor Address, click **172.16.15.14**.	From the [edit] hierarchy level, enter edit protocols bgp group groupA neighbor 172.16.15.14

Table 162: Creating a Policy to Accept and Apply Damping on Routes *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Apply the policy as an import policy for the BGP neighbor. The routing policy is evaluated when routes are imported to the routing table.	1. Next to Import, click **Add new entry**. 2. In the Value box, type the name of the policy. 3. Click **OK**.	Apply the policy: set import dampenpolicy1 You can refer to the same routing policy one or more times in the same or a different import statement.

Chapter 25

Configuring Stateless Firewall Filters (ACLs)

A *stateless* firewall filter evaluates the contents of packets transiting the device from a source to a destination, or packets originating from, or destined for, the Routing Engine. Stateless firewall filters applied to the Routing Engine interface protect the processes and resources owned by the Routing Engine. A stateless firewall filter evaluates every packet, including fragmented packets.

A stateless firewall filter, often called a firewall filter or access control list (ACL), statically evaluates packet contents. In contrast, a *stateful* firewall filter uses connection state information derived from past communications and other applications to make dynamic control decisions.

For information about security policies and *stateful* firewalls, see the *JUNOS Software Security Configuration Guide*.

You can use either the J-Web configuration editor or the CLI to configure stateless firewall filters.

This chapter contains the following topics. For more information about stateless firewall filters, see the *JUNOS Policy Framework Configuration Guide*.

Stateless Firewall Filters

This section contains the following topics:

Stateless Firewall Filter Overview

A *stateless* firewall filter can filter packets transiting the device from a source to a destination, or packets originating from, or destined for, the Routing Engine. Stateless firewall filters applied to the Routing Engine interface protect the processes and resources owned by the Routing Engine.

You can apply a stateless firewall filter to an input or output interface, or to both. Every packet, including fragmented packets, is evaluated against stateless firewall filters.

Stateless Firewall Filter Terms

All stateless firewall filters contain one or more terms, and each term consists of two components—match conditions and actions. The match conditions define the values or fields that the packet must contain to be considered a match. If a packet is a match, the corresponding action is taken. By default, a packet that does not match a firewall filter is discarded.

NOTE: A firewall filter with a large number of terms can adversely affect both the configuration commit time and the performance of the Routing Engine.

Chained Stateless Firewall Filters

You can configure a stateless firewall filter within the term of another filter. This method enables you to add common terms to multiple filters without having to modify all filter definitions. You can configure one filter with the desired common terms, and configure this filter as a term in other filters. Consequently, to make a change in these common terms, you need to modify only one filter that contains the common terms, instead of multiple filters. For more information about how to configure a filter within a filter, see the *JUNOS Policy Framework Configuration Guide*.

Planning a Stateless Firewall Filter

Before creating a stateless firewall filter and applying it to an interface, determine what you want the firewall filter to accomplish and how to use its match conditions and actions to achieve your goal. Also, make sure you understand how packets are matched and the default action of the resulting firewall filter.

CAUTION: If a packet does not match any terms in a stateless firewall filter rule, the packet is discarded. Take care that you do not configure a firewall filter that prevents you from accessing the device after you commit the configuration. For example, if you configure a firewall filter that does not match HTTP or HTTPS packets, you cannot access the device with the J-Web interface.

To configure a stateless firewall filter, determine the following:

- Purpose of the firewall filter—for example, to limit traffic to certain protocols, IP source or destination addresses, or data rates, or to prevent denial-of-service (DoS) attacks.
- Appropriate match conditions. The packet header fields to match—for example, IP header fields (such as source and destination IP addresses, protocols, and IP options), TCP header fields (such as source and destination ports and flags), and ICMP header fields (such as ICMP packet type and code).
- Action to take if a match occurs—for example, accept, discard, or evaluate the next term.
- (Optional) Action modifiers. Additional actions to take if a packet matches—for example, count, log, rate limit, or police a packet.
- Interface on which the firewall filter is applied. The input or output side, or both sides, of the Routing Engine interface or a non-Routing Engine interface.

For more information about what a stateless firewall filter can include, see "Stateless Firewall Filter Match Conditions" on page 523. For more information about stateless firewall filters, see the *JUNOS Policy Framework Configuration Guide*.

Stateless Firewall Filter Match Conditions

Table 163 on page 524 lists the match conditions you can specify in stateless firewall filter terms. Some of the numeric range and bit-field match conditions allow you to specify a text synonym. For a complete list of the synonyms, do any of the following:

- If you are using the J-Web interface, select the synonym from the appropriate list.
- If you are using the CLI, type a question mark (?) after the from statement.
- See the *JUNOS Policy Framework Configuration Guide*.

To specify a bit-field match condition with values, such as tcp-flags, you must enclose the values in quotation marks (" "). You can use bit-field logical operators to create expressions that are evaluated for matches. For example, if the following expression is used in a filter term, a match occurs if the packet is the initial packet of a TCP session:

```
tcp-flags "syn & !ack"
```

Table 164 on page 526 lists the bit-field logical operators in order of highest to lowest precedence.

You can use text synonyms to specify some common bit-field matches. In the previous example, you can specify tcp-initial to specify the same match condition.

NOTE: When the device compares the stateless firewall filter match conditions to a packet, it compares only the header fields specified in the match condition. There is no implied protocol match. For example, if you specify a match of destination-port ssh, the device checks for a value of 0x22 in the 2-byte field that is two bytes after the IP packet header. The protocol field of the packet is not checked.

Table 163: Stateless Firewall Filter Match Conditions

Match Condition	Description
Numeric Range Match Conditions	
keyword-except	Negates a match—for example, destination-port-except *number*. The following keywords accept the -except extension: destination-port, dscp, esp-spi, forwarding-class, fragment-offset, icmp-code, icmp-type, interface-group, ip-options, packet-length, port, precedence, protocol and source-port.
destination-port *number*	Matches a TCP or User Datagram Protocol (UDP) destination port field. You cannot specify both the port and destination-port match conditions in the same term. Normally, you specify this match in conjunction with the protocol tcp or protocol udp match statement to determine which protocol is being used on the port. In place of the numeric value, you can specify a text synonym. For example, you can specify telnet or 23.
esp-spi *spi-value*	Matches an IPSec encapsulating security payload (ESP) security parameter index (SPI) value. Match on this specific SPI value. You can specify the ESP SPI value in either hexadecimal, binary, or decimal form.
forwarding-class *class*	Matches a forwarding class. Specify assured-forwarding, best-effort, expedited-forwarding, or network-control.
fragment-offset *number*	Matches the fragment offset field.
icmp-code *number*	Matches the ICMP code field. Normally, you specify this match condition in conjunction with the protocol icmp match statement to determine which protocol is being used on the port. This value or keyword provides more specific information than icmp-type. Because the value's meaning depends on the associated icmp-type, you must specify icmp-type along with icmp-code. In place of the numeric value, you can specify a text synonym. For example, you can specify ip-header-bad or 0.
icmp-type *number*	Matches the ICMP packet type field. Normally, you specify this match condition in conjunction with the protocol icmp match statement to determine which protocol is being used on the port. In place of the numeric value, you can specify a text synonym. For example, you can specify time-exceeded or 11.
interface-group *group-number*	Matches the interface group on which the packet was received. An interface group is a set of one or more logical interfaces. For information about configuration interface groups, see the *JUNOS Policy Framework Configuration Guide*.

Table 163: Stateless Firewall Filter Match Conditions *(continued)*

Match Condition	Description
packet-length *bytes*	Matches the length of the received packet, in bytes. The length refers only to the IP packet, including the packet header, and does not include any Layer 2 encapsulation overhead.
port *number*	Matches a TCP or UDP source or destination port field. You cannot specify both the port match and either the destination-port or source-port match conditions in the same term. Normally, you specify this match condition in conjunction with the protocol tcp or protocol udp match statement to determine which protocol is being used on the port. In place of the numeric value, you can specify a text synonym. For example, you can specify bgp or 179.
precedence *ip-precedence-field*	Matches the IP precedence field. You can specify precedence in either hexadecimal, binary, or decimal form. In place of the numeric value, you can specify a text synonym. For example, you can specify immediate or 0x40.
protocol *number*	Matches the IP protocol field. In place of the numeric value, you can specify a text synonym. For example, you can specify ospf or 89.
source-port *number*	Matches the TCP or UDP source port field. You cannot specify the port and source-port match conditions in the same term. Normally, you specify this match condition in conjunction with the protocol tcp or protocol udp match statement to determine which protocol is being used on the port. In place of the numeric value, you can specify a text synonym. For example, you can specify http or 80.
Address Match Conditions	
address *prefix*	Matches the IP source or destination address field. You cannot specify both the address and the destination-address or source-address match conditions in the same term.
destination-address *prefix*	Matches the IP destination address field. You cannot specify the destination-address and address match conditions in the same term.
destination-prefix-list *prefix-list*	Matches the IP destination prefix list field. You cannot specify the destination-prefix-list and prefix-list match conditions in the same term.
prefix-list *prefix-list*	Matches the IP source or destination prefix list field. You cannot specify both the prefix-list and the destination-prefix-list or source-prefix-list match conditions in the same term.
source-address *prefix*	Matches the IP source address field. You cannot specify the source-address and address match conditions in the same rule.
source-prefix-list *prefix-list*	Matches the IP source prefix list field. You cannot specify the source-prefix-list and prefix-list match conditions in the same term.
Bit-Field Match Conditions with Values	
fragment-flags *number*	Matches an IP fragmentation flag. In place of the numeric value, you can specify a text synonym. For example, you can specify more-fragments or 0x2000.

Table 163: Stateless Firewall Filter Match Conditions *(continued)*

Match Condition	Description
ip-options *number*	Matches an IP option. In place of the numeric value, you can specify a text synonym. For example, you can specify record-route or 7.
tcp-flags *number*	Matches a TCP flag. Normally, you specify this match condition in conjunction with the protocol tcp match statement to determine which protocol is being used on the port. In place of the numeric value, you can specify a text synonym. For example, you can specify syn or 0x02.
Bit-Field Text Synonym Match Conditions	
first-fragment	Matches the first fragment of a fragmented packet. This condition does not match unfragmented packets.
is-fragment	Matches the trailing fragment of a fragmented packet. It does not match the first fragment of a fragmented packet. To match both first and trailing fragments, you can use two terms, or you can use fragment-offset 0-8191.
tcp-established	Matches a TCP packet other than the first packet of a connection. This match condition is a synonym for "(ack | rst)". This condition does not implicitly check that the protocol is TCP. To do so, specify the protocol tcp match condition.
tcp-initial	Matches the first TCP packet of a connection. This match condition is a synonym for "(syn & !ack)". This condition does not implicitly check that the protocol is TCP. To do so, specify the protocol tcp match condition.

Table 164: Stateless Firewall Filter Bit-Field Logical Operators

Logical Operator	Description
(...)	Grouping
!	Negation
& or +	Logical AND
| or ,	Logical OR

Stateless Firewall Filter Actions and Action Modifiers

Table 165 on page 527 lists the actions and action modifiers you can specify in stateless firewall filter terms.

Table 165: Stateless Firewall Filter Actions and Action Modifiers

Action or Action Modifier	Description
accept	Accepts a packet. This is the default if the packet matches. However, we strongly recommend that you always explicitly configure an action in the **then** statement.
discard	Discards a packet silently, without sending an Internet Control Message Protocol (ICMP) message. Packets are available for logging and sampling before being discarded.
next term	Continues to the next term for evaluation.
reject *<message-type>*	Discards a packet, sending an ICMP destination unreachable message. Rejected packets are available for logging and sampling. You can specify one of the following message types: **administratively-prohibited** (default), **bad-host-tos**, **bad-network-tos**, **host-prohibited**, **host-unknown**, **host-unreachable**, **network-prohibited**, **network-unknown**, **network-unreachable**, **port-unreachable**, **precedence-cutoff**, **precedence-violation**, **protocol-unreachable**, **source-host-isolated**, **source-route-failed**, or **tcp-reset**. If you specify **tcp-reset**, a TCP reset is returned (indicating the end of a TCP flow), if the packet is a TCP packet. Otherwise, nothing is returned.
routing-instance *routing-instance*	Routes the packet using the specified routing instance.
Action Modifiers	
count *counter-name*	Counts the number of packets passing this term. The name can contain letters, numbers, and hyphens (-), and can be up to 24 characters long. A counter name is specific to the filter that uses it, so all interfaces that use the same filter increment the same counter.
forwarding-class *class-name*	Classifies the packet to the specified forwarding class.
log	Logs the packet's header information in the Routing Engine. You can access this information by entering the **show firewall log** command at the CLI.
loss-priority *priority*	Sets the scheduling priority of the packet. The priority can be **low** or **high**.
policer *policer-name*	Applies rate limits to the traffic using the named policer.
sample	Samples the traffic on the interface. Use this modifier only when traffic sampling is enabled. For more information, see the *JUNOS Policy Framework Configuration Guide*.
syslog	Records information in the system logging facility. This action can be used in conjunction with all options except **discard**.

Before You Begin

If you do not already have an understanding of firewall filters, read "Stateless Firewall Filters" on page 521.

Unlike a stateful firewall filter, you can configure a stateless firewall filter before configuring the interfaces on which they are applied.

CAUTION: If a packet does not match any terms in a firewall filter rule, the packet is discarded. Take care you do not configure a stateless firewall filter that prevents

you from accessing the device after you commit the configuration. For example, if you configure a firewall filter that does not match HTTP or HTTPS packets, you cannot access the device with the J-Web interface.

Configuring a Stateless Firewall Filter with a Configuration Editor

The section contains the following topics. For stateless firewall match conditions, actions, and modifiers, see "Stateless Firewall Filter Match Conditions" on page 523 and "Stateless Firewall Filter Actions and Action Modifiers" on page 526.

Stateless Firewall Filter Strategies

For best results, use the following sections to plan the purpose and contents of a stateless firewall filter before starting configuration.

Strategy for a Typical Stateless Firewall Filter

A primary goal of a typical stateless firewall filter is to protect the Routing Engine processes and resources from malicious or untrusted packets. You can configure a firewall filter like the sample filter protect-RE to restrict traffic destined for the Routing Engine based on its source, protocol, and application. In addition, you can limit the traffic rate of packets destined for the Routing Engine to protect against flood, or *denial-of-service* (DoS), attacks.

For details, see "Configuring a Routing Engine Firewall Filter for Services and Protocols from Trusted Sources" on page 529 and "Configuring a Routing Engine Firewall Filter to Protect Against TCP and ICMP Floods" on page 531.

Strategy for Handling Packet Fragments

You can configure a stateless firewall filter like the sample filter fragment-filter to address special circumstances associated with fragmented packets destined for the Routing Engine. Because the device evaluates every packet against a firewall filter (including fragments), you must configure the filter to accommodate fragments that do not contain packet header information. Otherwise, the filter discards all but the first fragment of a fragmented packet.

For details, see "Configuring a Routing Engine Firewall Filter to Handle Fragments" on page 536.

Configuring a Routing Engine Firewall Filter for Services and Protocols from Trusted Sources

The following example shows how to create a stateless firewall filter, protect-RE, that discards all traffic destined for the Routing Engine, except SSH and BGP protocol packets from specified trusted sources. Table 166 on page 529 lists the terms that are configured in this sample filter.

Table 166: Sample Stateless Firewall Filter protect-RE Terms to Allow Packets from Trusted Sources

Term	Purpose
ssh-term	Accepts TCP packets with a source address of 192.168.122.0/24 and a destination port that specifies SSH.
bgp-term	Accepts TCP packets with a source address of 10.2.1.0/24 and a destination port that specifies the BGP protocol.
discard-rest-term	For all packets that are not accepted by ssh-term or bgp-term, creates a firewall filter log and system logging records, then discards all packets. To view the log, enter the show firewall log operational mode command. (For more information, see "Displaying Stateless Firewall Filter Logs" on page 545.)

By applying firewall filter protect-RE to the Routing Engine, you specify which protocols and services, or applications, are allowed to reach the Routing Engine, and you ensure the packets are from a trusted source. This protects processes running on the Routing Engine from an external attack.

To use the configuration editor to configure the stateless firewall filter:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 167 on page 529.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following procedures:
   - To display the configuration, see "Displaying Stateless Firewall Filter Configurations" on page 542.
   - To apply the firewall filter to the Routing Engine, see "Applying a Stateless Firewall Filter to an Interface" on page 541.
   - To verify the firewall filter, see "Verifying a Services, Protocols, and Trusted Sources Firewall Filter" on page 547.

Table 167: Configuring a Protocols and Services Firewall Filter for the Routing Engine

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Firewall** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Firewall, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit firewall

Table 167: Configuring a Protocols and Services Firewall Filter for the Routing Engine *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define protect-RE and ssh-term, and define the protocol, destination port, and source address match conditions.	1. Next to Filter, click **Add new entry**. 2. In the Filter name box, type protect-RE. 3. Next to Term, click **Add New Entry**. 4. In the Rule name box, type ssh-term. 5. Next to From, click **Configure**. 6. In the Protocol choice list, select **Protocol**. 7. Next to Protocol, click **Add new entry**. 8. In the Value keyword list, select **tcp**. 9. Click **OK**. 10. In the Destination port choice list, select **Destination port**. 11. Next to Destination port, click **Add new entry**. 12. In the Value keyword list, select **ssh**. 13. Click **OK**. 14. Next to Source address, click **Add new entry**. 15. In the Address box, type 192.168.122.0/24. 16. Click **OK** twice.	Set the term name and define the match conditions: set family inet filter protect-RE term ssh-term from protocol tcp destination-port ssh source-address 192.168.122.0/24
Define the actions for ssh-term.	1. On the Term ssh-term page, next to Then, click **Configure**. 2. In the Designation list, select **Accept**. 3. Click **OK** twice.	Set the actions: set family inet filter protect-RE term ssh-term then accept

Table 167: Configuring a Protocols and Services Firewall Filter for the Routing Engine *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define bgp-term, and define the protocol, destination port, and source address match conditions.	1. On the Filter protect-RE page, next to Term, click **Add New Entry**. 2. In the Rule name box, type bgp-term. 3. Next to From, click **Configure**. 4. In the Protocol choice list, select **Protocol**. 5. Next to Protocol, click **Add new entry**. 6. In the Value keyword list, select **tcp**. 7. Click **OK**. 8. In the Destination port choice list, select **Destination port**. 9. Next to Destination port, click **Add new entry**. 10. In the Value keyword list, select **bgp**. 11. Click **OK**. 12. Next to Source address, click **Add new entry**. 13. In the Address box, type 10.2.1.0/24. 14. Click **OK** twice.	Set the term name and define the match conditions: set family inet filter protect-RE term bgp-term from protocol tcp destination-port bgp source-address 10.2.1.0/24
Define the action for bgp-term.	1. On the Term bgp-term page, next to Then, click **Configure**. 2. In the Designation list, select **Accept**. 3. Click **OK** twice.	Set the action: set family inet filter protect-RE term bgp-term then accept
Define discard-rest-term and its action.	1. On the Filter protect-RE page, next to Term, click **Add New Entry**. 2. In the Rule name box, type discard-rest-term. 3. Next to Then, click **Configure**. 4. Next to Log, select the check box. 5. Next to Syslog, select the check box. 6. In the Designation list, select **Discard**. 7. Click **OK** four times.	Set the term name and define its actions: set family inet filter protect-RE term discard-rest-term then log syslog discard

Configuring a Routing Engine Firewall Filter to Protect Against TCP and ICMP Floods

The procedure in this section creates a sample stateless firewall filter, protect-RE, that limits certain TCP and ICMP traffic destined for the Routing Engine. A router without

this kind of protection is vulnerable to TCP and ICMP flood attacks—also called denial-of-service (DoS) attacks. For example:

- A TCP flood attack of SYN packets initiating connection requests can so overwhelm the device that it can no longer process legitimate connection requests, resulting in denial of service.
- An ICMP flood can overload the device with so many echo requests (ping requests) that it expends all its resources responding and can no longer process valid network traffic, also resulting in denial of service.

Applying a firewall filter like protect-RE to the Routing Engine protects against these types of attacks.

For each term in the sample filter, you first create a policer and then incorporate it into the action of the term. For more information about firewall filter policers, see the *JUNOS Policy Framework Configuration Guide*.

If you want to include the terms created in this procedure in the protect-RE firewall filter configured in the previous section (see "Configuring a Routing Engine Firewall Filter for Services and Protocols from Trusted Sources" on page 529), perform the configuration tasks in this section first, then configure the terms as described in the previous section. This approach ensures that the rate-limiting terms are included as the first two terms in the firewall filter.

NOTE: You can move terms within a firewall filter by using the insert CLI command. For more information, see the *JUNOS CLI User Guide*.

Table 168 on page 532 lists the terms that are configured in this sample filter.

Table 168: Sample Stateless Firewall Filter protect-RE Terms to Protect Against Floods

Term	Purpose	Policer
tcp-connection-term	Polices the following types of TCP packets with a source address of 192.168.122.0/24 or 10.2.1.0/24: ■ Connection request packets (SYN and ACK flag bits equal 1 and 0) ■ Connection release packets (FIN flag bit equals 1) ■ Connection reset packets (RST flag bit equals 1)	tcp-connection-policer—Limits the traffic rate and burst size of these TCP packets to 500,000 bps and 15,000 bytes. Packets that exceed the traffic rate are discarded.
icmp-term	Polices the following types of ICMP packets. All are counted in counter icmp-counter. ■ Echo request packets ■ Echo response packets ■ Unreachable packets ■ Time-exceeded packets	icmp-policer—Limits the traffic rate and burst size of these ICMP packets to 1,000,000 bps and 15,000 bytes. Packets that exceed the traffic rate are discarded.

To use the configuration editor to configure the policers and the stateless firewall filter:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. To configure the firewall filter policers, perform the configuration tasks described in Table 169 on page 533.
3. To configure the prefix lists and the firewall filter, perform the configuration tasks described in Table 170 on page 534.
4. If you are finished configuring the router, commit the configuration.
5. Go on to one of the following procedures:
   - To display the configuration, see "Displaying Stateless Firewall Filter Configurations" on page 542.
   - To apply the firewall filter to the Routing Engine, see "Applying a Stateless Firewall Filter to an Interface" on page 541.
   - To verify the firewall filter, see "Verifying a TCP and ICMP Flood Firewall Filter" on page 547.

Table 169: Configuring Policers for TCP and ICMP

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Firewall** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Firewall, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit firewall
Define tcp-connection-policer and set its rate limits. The burst size limit can be from 1,500 bytes through 100,000,000 bytes. The bandwidth limit can be from 32,000 bps through 32,000,000,000 bps. Use the following abbreviations when specifying these limits: ■ k (1000) ■ m (1,000,000) ■ g (1,000,000,000)	1. Next to Policer, click **Add new entry**. 2. In the Policer name box, type tcp-connection-policer. 3. Next to Filter specific, select the check box. 4. Next to If Exceeding, select the check box and click **Configure**. 5. In the Burst size limit box, type 15k. 6. In the Bandwidth list, select **Bandwidth limit**. 7. In the Bandwidth limit box, type 500k. 8. Click **OK**.	Set the policer name and its rate limits: set policer tcp-connection-policer filter-specific if-exceeding burst-size-limit 15k bandwidth-limit 500k

Table 169: Configuring Policers for TCP and ICMP *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define the policer action for tcp-connection-policer.	1. On the Policer tcp-connection-policer page, next to Then, click **Configure**. 2. Next to Discard, select the check box. 3. Click **OK** twice.	Set the policer action: set policer tcp-connection-policer then discard
Define icmp-policer and set its rate limits. The burst size limit can be from 1,500 bytes through 100,000,000 bytes. The bandwidth limit can be from 32,000 bps through 32,000,000,000 bps. Use the following abbreviations when specifying these limits: ■ k (1000) ■ m (1,000,000) ■ g (1,000,000,000)	1. On the Firewall page, next to Policer, click **Add new entry**. 2. In the Policer name box, type icmp-policer. 3. Next to Filter specific, select the check box. 4. Next to If Exceeding, select the check box and click **Configure**. 5. In the Burst size limit box, type 15k. 6. In the Bandwidth list, select **Bandwidth limit**. 7. In the Bandwidth limit box, type 1m. 8. Click **OK**.	Set the policer name and its rate limits: set policer icmp-policer filter-specific if-exceeding burst-size-limit 15k bandwidth-limit 1m
Define the policer action for icmp-policer.	1. On the Policer icmp-policer page, next to Then, click **Configure**. 2. Next to Discard, select the check box. 3. Click **OK** three times.	Set the policer action: set policer icmp-policer then discard

Table 170: Configuring a TCP and ICMP Flood Firewall Filter for the Routing Engine

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Policy options** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Policy options, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit policy-options

Table 170: Configuring a TCP and ICMP Flood Firewall Filter for the Routing Engine *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define the prefix list trusted-addresses.	1. Next to Prefix list, click **Add new entry**. 2. In the Name box, type trusted-addresses. 3. Next to Prefix list item, click **Add new entry**. 4. In the Prefix box, type 192.168.122.0/24. 5. Click **OK**. 6. Next to Prefix list item, click **Add new entry**. 7. In the Prefix box, type 10.2.1.0/24. 8. Click **OK** three times.	Set the prefix list: set prefix-list trusted-addresses 192.168.122.0/24 set prefix-list trusted-addresses 10.2.1.0/24
Navigate to the **Firewall** level in the configuration hierarchy.	On the main Configuration page next to Firewall, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit firewall
Define protect-RE and tcp-connection-term, and define the source prefix list match condition.	1. Next to Filter, click **Add new entry**. 2. In the Filter name box, type protect-RE. 3. Next to Term, click **Add New Entry**. 4. In the Rule name box, type tcp-connection-term. 5. Next to From, click **Configure**. 6. Next to Source prefix list, click **Add new entry**. 7. In the Name box, type trusted-addresses. 8. Click **OK**.	Set the term name and define the source address match condition: set family inet filter protect-RE term tcp-connection-term from source-prefix-list trusted-addresses
Define the TCP flags and protocol match conditions for tcp-connection-term.	1. In the TCP flags box, type (syn & !ack) | fin | rst. 2. In the Protocol choice list, select **Protocol**. 3. Next to Protocol, click **Add new entry**. 4. In the Value keyword list, select **tcp**. 5. Click **OK**.	Set the TCP flags and protocol and protocol match conditions for the term: set family inet filter protect-RE term tcp-connection-term from protocol tcp tcp-flags "(syn & !ack) | fin | rst"
Define the actions for tcp-connection-term.	1. On the Term tcp-connection-term page, next to Then, click **Configure**. 2. In the Policer box, type tcp-connection-policer. 3. In the Designation list, select **Accept**. 4. Click **OK** twice.	Set the actions: set family inet filter protect-RE term tcp-connection-term then policer tcp-connection-policer accept

Table 170: Configuring a TCP and ICMP Flood Firewall Filter for the Routing Engine *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define icmp-term, and define the protocol.	1. On the Filter protect-RE page, next to Term, click **Add New Entry**. 2. In the Rule name box, type icmp-term. 3. Next to From, click **Configure**. 4. In the Protocol choice list, select **Protocol**. 5. Next to Protocol, click **Add new entry**. 6. In the Value keyword list, select **icmp**. 7. Click **OK**.	Set the term name and define the protocol: set family inet filter protect-RE term icmp-term from protocol icmp
Define the ICMP type match conditions.	1. In the Icmp type choice list, select **Icmp type**. 2. Next to Icmp type, click **Add new entry**. 3. In the Value keyword list, select **echo-request**. 4. Click **OK**. 5. Next to Icmp type, click **Add new entry**. 6. In the Value keyword list, select **echo-reply**. 7. Click **OK**. 8. Next to Icmp type, click **Add new entry**. 9. In the Value keyword list, select **unreachable**. 10. Click **OK**. 11. Next to Icmp type, click **Add new entry**. 12. In the Value keyword list, select **time-exceeded**. 13. Click **OK**.	Set the ICMP type match conditions: set family inet filter protect-RE term icmp-term from icmp-type [echo-request echo-reply unreachable time-exceeded]
Define the actions for icmp-term.	1. On the icmp-term page, next to Then, click **Configure**. 2. In the Count box, type icmp-counter. 3. In the Policer box, type icmp-policer. 4. In the Designation list, select **Accept**. 5. Click **OK** four times.	Set the actions: set family inet filter protect-RE term icmp-term then policer icmp-policer count icmp-counter accept

Configuring a Routing Engine Firewall Filter to Handle Fragments

The procedure in this section creates a sample stateless firewall filter, fragment-RE, that handles fragmented packets destined for the Routing Engine. By applying

fragment-RE to the Routing Engine, you protect against the use of IP fragmentation as a means to disguise TCP packets from a firewall filter.

Table 171 on page 537 lists the terms that are configured in this sample filter.

Table 171: Sample Stateless Firewall Filter fragment-RE Terms

Term	Purpose
small-offset-term	Discards IP packets with a fragment offset of 1 through 5, and adds a record to the system logging facility.
not-fragmented-term	Accepts unfragmented TCP packets with a source address of 10.2.1.0/24 and a destination port that specifies the BGP protocol. A packet is considered unfragmented if its MF flag and its fragment offset in the TCP header equal 0.
first-fragment-term	Accepts the first fragment of a fragmented TCP packet with a source address of 10.2.1.0/24 and a destination port that specifies the BGP protocol.
fragment-term	Accepts all packet fragments with an offset of 6 through 8191.

For example, consider an IP packet that is fragmented into the smallest allowable fragment size of 8 bytes (a 20-byte IP header plus an 8-byte payload). If this IP packet carries a TCP packet, the first fragment (fragment offset of 0) that arrives at the device contains only the TCP source and destination ports (first 4 bytes), and the sequence number (next 4 bytes). The TCP flags, which are contained in the next 8 bytes of the TCP header, arrive in the second fragment (fragment offset of 1). The fragment-RE filter works as follows:

- Term small-offset-term discards small offset packets to ensure that subsequent terms in the firewall filter can be matched against all the headers in the packet.
- Term fragment-term accepts all fragments that were not discarded by small-offset-term. However, only those fragments that are part of a packet containing a first fragment accepted by first-fragment-term are reassembled by the device.

For more information about IP fragment filtering, see RFC 1858, *Security Considerations for IP Fragment Filtering*.

To use the configuration editor to configure the stateless firewall filter:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. To configure the firewall filter, perform the configuration tasks described in Table 172 on page 538.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following procedures:
   - To display the configuration, see “Displaying Stateless Firewall Filter Configurations” on page 542.

- To apply the firewall filter to the Routing Engine, see "Applying a Stateless Firewall Filter to an Interface" on page 541.
- To verify the firewall filter, see "Verifying a Firewall Filter That Handles Fragments" on page 548.

Table 172: Configuring a Fragments Firewall Filter for the Routing Engine

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Firewall** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Firewall, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit firewall
Define fragment-RE and small-offset-term, and define the fragment offset match condition. The fragment offset can be from 1 through 8191.	1. Next to Filter, click **Add new entry**. 2. In the Filter name box, type fragment-RE. 3. Next to Term, click **Add New Entry**. 4. In the Rule name box, type small-offset-term. 5. Next to From, click **Configure**. 6. In the Fragment offset choice list, select **Fragment offset**. 7. Next to Fragment offset, select **Add New Entry**. 8. In the Range box, type 1-5. 9. Click **OK** twice.	Set the term name and define the fragment offset match condition: set family inet filter fragment-RE term small-offset-term from fragment-offset 1-5
Define the action for small-offset-term.	1. On the Term small-offset-term page, next to Then, click **Configure**. 2. Next to Syslog, select the check box. 3. In the Designation list, select **Discard**. 4. Click **OK** twice.	Set the action: set family inet filter fragment-RE term small-offset-term then syslog discard

Table 172: Configuring a Fragments Firewall Filter for the Routing Engine *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define not-fragmented-term, and define the fragment, protocol, destination port, and source address match conditions.	1. On the Filter fragment-RE page, next to Term, click **Add New Entry**. 2. In the Term name box, type not-fragmented-term. 3. Next to From, click **Configure**. 4. In the Fragment flags box, type 0x0. 5. In the Fragment offset choice list, select **Fragment offset**. 6. Next to Fragment offset, select **Add New Entry**. 7. In the Range box, type 0. 8. Click **OK**. 9. In the Protocol choice list, select **Protocol**. 10. Next to Protocol, click **Add new entry**. 11. In the Value keyword list, select **tcp**. 12. Click **OK**. 13. In the Destination port choice list, select **Destination port**. 14. Next to Destination port, click **Add new entry**. 15. In the Value keyword list, select **bgp**. 16. Click **OK**. 17. Next to Source address, click **Add new entry**. 18. In the Address box, type 10.2.1.0/24. 19. Click **OK** twice.	Set the term name and define match conditions: set family inet filter fragment-RE term not-fragmented-term from fragment-flags 0x0 fragment-offset 0 protocol tcp destination-port bgp source-address 10.2.1.0/24
Define the action for not-fragmented-term.	1. On the Term not-fragmented-term page, next to Then, click **Configure**. 2. In the Designation list, select **Accept**. 3. Click **OK** twice.	Set the action: set family inet filter fragment-RE term not-fragmented-term then accept

Table 172: Configuring a Fragments Firewall Filter for the Routing Engine *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define first-fragment-term, and define the fragment, protocol, destination port, and source address match conditions.	1. On the Filter fragment-RE page, next to Term, click **Add New Entry**. 2. In the Rule name box, type first-fragment-term. 3. Next to From, click **Configure**. 4. Next to First fragment, select the check box. 5. In the Protocol choice list, select **Protocol**. 6. Next to Protocol, click **Add new entry**. 7. In the Value keyword list, select **tcp**. 8. Click **OK**. 9. In the Destination port choice list, select **Destination port**. 10. Next to Destination port, click **Add new entry**. 11. In the Value keyword list, select **bgp**. 12. Click **OK**. 13. Next to Source address, click **Add new entry**. 14. In the Address box, type 10.2.1.0/24. 15. Click **OK** twice.	Set the term name and define match conditions: set family inet filter fragment-RE term first-fragment-term from first-fragment protocol tcp destination-port bgp source-address 10.2.1.0/24
Define the action for first-fragment-term.	1. On the Term first-fragment-term page, next to Then, click **Configure**. 2. In the Designation list, select **Accept**. 3. Click **OK** twice.	Set the action: set family inet filter fragment-RE term first-fragment-term then accept
Define fragment-term and define the fragment match condition.	1. On the Filter fragment-RE page, next to Term, click **Add New Entry**. 2. In the Rule name box, type fragment-term. 3. Next to From, click **Configure**. 4. In the Fragment offset choice list, select **Fragment offset**. 5. Next to Fragment offset, select **Add New Entry**. 6. In the Range box, type 6-8191. 7. Click **OK** twice.	Set the term name and define match conditions: set family inet filter fragment-RE term fragment-term from fragment-offset 6–8191

Table 172: Configuring a Fragments Firewall Filter for the Routing Engine *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define the action for fragment-term.	1. On the Term fragment-term page, next to Then, click **Configure**. 2. In the Designation list, select **Accept**. 3. Click **OK** four times.	Set the action: set family inet filter fragment-RE term fragment-term then accept

Applying a Stateless Firewall Filter to an Interface

You can apply a stateless firewall to the input or output sides, or both, of an interface. To filter packets transiting the device, apply the firewall filter to any non-Routing Engine interface. To filter packets originating from, or destined for, the Routing Engine, apply the firewall filter to the loopback (lo0) interface.

For example, to apply the firewall filter protect-RE to the input side of the Routing Engine interface, follow this procedure:

1. Perform the configuration tasks described in Table 173 on page 541.
2. If you are finished configuring the router, commit the configuration.

Table 173: Applying a Firewall Filter to the Routing Engine Interface

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Inet** level in the configuration hierarchy. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**. 3. Under Interface name, click **lo0**. 4. Under Interface unit number, click **0**. 5. Under Family, make sure the Inet check box is selected, and click **Configure** or **Edit**.	From the [edit] hierarchy level, apply the filter to the interface: set interfaces lo0 unit 0 family inet filter input protect-RE
Apply protect-RE as an input filter to the lo0 interface.	1. Next to Filter, click **Configure**. 2. In the Input box, type protect-RE. 3. Click **OK** five times.	

To view the configuration of the Routing Engine interface, enter the show interfaces lo0 command. For example:

```
user@host# show interfaces lo0
unit 0 {
   family inet {
```

```
        filter {
            input protect-RE;
        }
        address 127.0.0.1/32;
    }
}
```

Verifying Stateless Firewall Filter Configuration

To verify a stateless firewall filter configuration, perform these tasks:

Displaying Stateless Firewall Filter Configurations

Purpose Verify the configuration of the firewall filter. You can analyze the flow of the filter terms by displaying the entire configuration.

Action From the J-Web interface, select **Configuration > View and Edit > View Configuration Text**. Alternatively, from configuration mode in the CLI, enter the show firewall command.

The sample output in this section displays the following firewall filters (in order):

- Stateless protect-RE filter configured in "Configuring a Routing Engine Firewall Filter for Services and Protocols from Trusted Sources" on page 529
- Stateless protect-RE filter configured in "Configuring a Routing Engine Firewall Filter to Protect Against TCP and ICMP Floods" on page 531
- Stateless fragment-RE filter configured in "Configuring a Routing Engine Firewall Filter to Handle Fragments" on page 536

```
[edit]
user@host# show firewall
firewall {
    family inet {
        filter protect-RE {
            term ssh-term {
                from {
                    source-address {
                        192.168.122.0/24;
                    }
                    protocol tcp;
                    destination-port ssh;
                }
                then accept;
            }
```

```
        term bgp-term {
            from {
                source-address {
                    10.2.1.0/24;
                }
                protocol tcp;
                destination-port bgp;
            }
            then accept;
        }
        term discard-rest-term {
            then {
                log;
                syslog;
                discard;
            }
        }
    }
  }
}

[edit]
user@host# show firewall
firewall {
    policer tcp-connection-policer {
        filter-specific;
        if-exceeding {
            bandwidth-limit 500k;
            burst-size-limit 15k;
        }
        then discard;
    }
    policer icmp-policer {
        filter-specific;
        if-exceeding {
            bandwidth-limit 1m;
            burst-size-limit 15k;
        }
        then discard;
    }
    family inet {
        filter protect-RE {
            term tcp-connection-term {
                from {
                    source-prefix-list {
                        trusted-addresses;
                    }
                    protocol tcp;
                    tcp-flags “(syn & !ack) | fin | rst”;
                }
                then {
                    policer tcp-connection-policer;
                    accept;
                }
            }
            term icmp-term {
```

```
            from {
                protocol icmp;
                icmp-type [ echo-request echo-reply unreachable time-exceeded ];
            }
            then {
                policer icmp-policer;
                count icmp-counter;
                accept;
            }
        }
        additional terms...
    }
  }
}

[edit]
user@host# show firewall
firewall {
  family inet {
    filter fragment-RE {
      term small-offset-term {
        from {
          fragment-offset 1-5;
        }
        then {
          syslog;
          discard;
        }
      }
      term not-fragmented-term {
        from {
          source-address {
            10.2.1.0/24;
          }
          fragment-offset 0;
          fragment-flags 0x0;
          protocol tcp;
          destination-port bgp;
        }
        then accept;
      }
      term first-fragment-term {
        from {
          source-address {
            10.2.1.0/24;
          }
          first-fragment;
          protocol tcp;
          destination-port bgp;
        }
        then accept;
      }
      term fragment-term {
        from {
          fragment-offset 6-8191;
        }
```

```
            then accept;
          }
          additional terms ...
        }
      }
    }
```

Meaning Verify that the output shows the intended configuration of the firewall filter.

Verify that the terms are listed in the order in which you want the packets to be tested. You can move terms within a firewall filter by using the insert CLI command.

Related Topics For more information about the format of a configuration file, see the *J-Web Interface User Guide* or the *JUNOS CLI User Guide*.

For information about the insert command, see the *JUNOS CLI User Guide*.

Displaying Stateless Firewall Filter Logs

Purpose Verify that packets are being logged. If you included the log or syslog action in a term, verify that packets matching the term are recorded in the firewall log or your system logging facility.

Action From operational mode in the CLI, enter the show firewall log command.

The log of discarded packets generated from the stateless firewall filter configured in "Configuring a Routing Engine Firewall Filter for Services and Protocols from Trusted Sources" on page 529 is displayed in the following sample output.

Sample Output

```
user@host> show firewall log
Log :
Time      Filter    Action Interface     Protocol Src Addr      Dest Addr
15:11:02  pfe       D      ge-0/0/0.0    TCP      172.17.28.19  192.168.70.71
15:11:01  pfe       D      ge-0/0/0.0    TCP      172.17.28.19  192.168.70.71
15:11:01  pfe       D      ge-0/0/0.0    TCP      172.17.28.19  192.168.70.71
15:11:01  pfe       D      ge-0/0/0.0    TCP      172.17.28.19  192.168.70.71
...
```

Meaning Each record of the output contains information about the logged packet. Verify the following information:

- Under Time, the time of day the packet was filtered is shown.
- The Filter output is always pfe.
- Under Action, the configured action of the term matches the action taken on the packet—A (accept), D (discard), R (reject).
- Under Interface, the inbound (ingress) interface on which the packet arrived is appropriate for the filter.
- Under Protocol, the protocol in the IP header of the packet is appropriate for the filter.

- Under Src Addr, the source address in the IP header of the packet is appropriate for the filter.
- Under Dest Addr, the destination address in the IP header of the packet is appropriate for the filter.

Related Topics For a complete description of show firewall log output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Displaying Firewall Filter Statistics

Purpose Verify that packets are being policed and counted.

Action From operational mode in the CLI, enter the show firewall filter *filter-name* command.

The value of the counter, icmp-counter, and the number of packets discarded by the policers in the stateless firewall filter configured in "Configuring a Routing Engine Firewall Filter to Protect Against TCP and ICMP Floods" on page 531 are displayed in the following sample output.

Sample Output

```
user@host> show firewall filter protect-RE
Filter: protect-RE
Counters:
Name                                                Bytes              Packets
icmp-counter                                      1040000                 5600
Policers:
Name                                                Packets
tcp-connection-policer                            643254873
icmp-policer                                         7391
```

Meaning Verify the following information:

- Next to Filter, the name of the firewall filter is correct.
- Under Counters:
  - Under Name, the names of any counters configured in the firewall filter are correct.
  - Under Bytes, the number of bytes that match the filter term containing the count *counter-name* action are shown.
  - Under Packets, the number of packets that match the filter term containing the count *counter-name* action are shown.
- Under Policers:
  - Under Name, the names of any policers configured in the firewall filter are correct.
  - Under Packets, the number of packets that match the conditions specified for the policer are shown.

Related Topics For a complete description of the show firewall filter command and output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying a Services, Protocols, and Trusted Sources Firewall Filter

Purpose Verify the stateless firewall filter configured in "Configuring a Routing Engine Firewall Filter for Services and Protocols from Trusted Sources" on page 529.

Action To verify that the actions of the firewall filter terms are taken, send packets to the Juniper Networks device that match the terms. In addition, verify that the filter actions are *not* taken for packets that do not match.

- Use the ssh *host-name* command from a host at an IP address that matches 192.168.122.0/24 to verify that you can log in to the device using only SSH from a host with this address prefix.
- Use the show route summary command to verify that the routing table on the device does not contain any entries with a protocol other than Direct, Local, BGP, or Static.

Sample Output

```
% ssh 192.168.249.71
%ssh host
user@host's password:
--- JUNOS 6.4-20040518.0 (JSERIES) #0: 2004-05-18 09:27:50 UTC

user@host>

user@host> show route summary
Router ID: 192.168.249.71

inet.0: 34 destinations, 34 routes (33 active, 0 holddown, 1 hidden)
              Direct:     10 routes,      9 active
               Local:      9 routes,      9 active
                 BGP:     10 routes,     10 active
              Static:      5 routes,      5 active
...
```

Meaning Verify the following information:

- You can successfully log in to the device using SSH.
- The show route summary command does not display a protocol other than Direct, Local, BGP, or Static.

Related Topics For a complete description of show route summary output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying a TCP and ICMP Flood Firewall Filter

Purpose Verify the stateless firewall filter configured in "Configuring a Routing Engine Firewall Filter to Protect Against TCP and ICMP Floods" on page 531.

Action To verify that the actions of the firewall filter terms are taken, send packets to the device that match the terms. In addition, verify that the filter actions are *not* taken for packets that do not match.

- Verify that the device can establish only TCP sessions with a host at an IP address that matches 192.168.122.0/24 or 10.2.1.0/24. For example, log in to the device with the telnet *host-name* command from another host with one of these address prefixes.

- Use the ping *host-name* command to verify that the device responds only to ICMP packets (such as ping requests) that do not exceed the policer traffic rates.
- Use the ping *host-name* size *bytes* command to exceed the policer traffic rates by sending ping requests with large data payloads.

Sample Output

```
user@host> telnet 192.168.249.71
Trying 192.168.249.71...
Connected to host.acme.net.
Escape character is '^]'.

host (ttyp0)

login: user
Password:

--- JUNOS 6.4-20040521.1 built 2004-05-21 09:38:12 UTC

user@host>

user@host> ping 192.168.249.71
PING host-ge-000.acme.net (192.168.249.71): 56 data bytes
64 bytes from 192.168.249.71: icmp_seq=0 ttl=253 time=11.946 ms
64 bytes from 192.168.249.71: icmp_seq=1 ttl=253 time=19.474 ms
64 bytes from 192.168.249.71: icmp_seq=2 ttl=253 time=14.639 ms
...

user@host> ping 192.168.249.71 size 20000
PING host-ge-000.acme.net (192.168.249.71): 20000 data bytes
^C
--- host-ge-000.acme.net ping statistics ---
12 packets transmitted, 0 packets received, 100% packet loss
```

Meaning Verify the following information:

- You can successfully log in to the device using Telnet.
- The device sends responses to the ping host command.
- The device does not send responses to the ping host size 20000 command.

Related Topics For more information about the ping command, see the *JUNOS Software Administration Guide* or the *JUNOS System Basics and Services Command Reference*.

For information about using the J-Web interface to ping a host, see the *JUNOS Software Administration Guide*.

For more information about the telnet command, see the *JUNOS Software Administration Guide* or the *JUNOS System Basics and Services Command Reference*.

Verifying a Firewall Filter That Handles Fragments

Purpose Verify the firewall filter configured in “Configuring a Routing Engine Firewall Filter to Handle Fragments” on page 536.

Action To verify that the actions of the firewall filter terms are taken, send packets to the device that match the terms. In addition, verify that the filter actions are *not* taken for packets that do not match.

- Verify that packets with small fragment offsets are recorded in the router's system logging facility.
- Use the show route summary command to verify that the routing table does not contain any entries with a protocol other than Direct, Local, BGP, or Static.

Sample Output

```
user@host> show route summary
Router ID: 192.168.249.71

inet.0: 34 destinations, 34 routes (33 active, 0 holddown, 1 hidden)
              Direct:     10 routes,      9 active
               Local:      9 routes,      9 active
                 BGP:     10 routes,     10 active
              Static:      5 routes,      5 active
...
```

Meaning Verify that the show route summary command does not display a protocol other than Direct, Local, BGP, or Static.

Related Topics For a complete description of show route summary output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Part 6
Configuring Class of Service

Chapter 26

Class-of-Service Overview

When a network experiences congestion and delay, some packets must be dropped. JUNOS software class-of-service (CoS) allows you to divide traffic into classes and offer various levels of throughput and packet loss when congestion occurs. This allows packet loss to happen according to the rules you configure.

For interfaces that carry IPv4 and MPLS traffic, you can configure the JUNOS software CoS features to provide multiple classes of service for different applications. On the device, you can configure multiple forwarding classes for transmitting packets, define which packets are placed into each output queue, schedule the transmission service level for each queue, and manage congestion using a random early detection (RED) algorithm.

Traffic shaping is the allocation of the appropriate amount of network bandwidth to every user and application on an interface. The appropriate amount of bandwidth is defined as cost-effective carrying capacity at a guaranteed CoS. You can use a J-series Services Router or an SRX-series services gateway to control traffic rate by applying classifiers and shapers.

The CoS features provide a set of mechanisms that you can use to provide differentiated services when best-effort delivery is insufficient.

Using JUNOS CoS features, you can assign service levels with different delay, jitter (delay variation), and packet loss characteristics to particular applications served by specific traffic flows. CoS is especially useful for networks supporting time-sensitive video and audio applications. To configure CoS features on a device, see “Configuring Class of Service” on page 579.

NOTE: Policing, scheduling, and shaping CoS services are not supported for pre-encryption and post-encryption packets going into and coming out of an IPsec VPN tunnel.

JUNOS software supports the following RFCs for traffic classification and policing:

- RFC 2474, *Definition of the Differentiated Services Field in the IPv4 and IPv6*
- RFC 2475, *An Architecture for Differentiated Services*
- RFC 2579, *Assured Forwarding PHB Group*
- RFC 2598, *An Expedited Forwarding PHB*

- RFC 2697, *A Single Rate Three Color Marker*
- RFC 2698, A *Two Rate Three Color Marker*

This chapter contains the following topics. For more information about CoS, see the *JUNOS Class of Service Configuration Guide*.

CoS Terms

Before configuring CoS, become familiar with the terms defined in Table 174 on page 554.

Table 174: CoS Terms

Term	Definition
assured forwarding (AF)	CoS packet forwarding class that provides a group of values you can define and includes four subclasses, AF1, AF2, AF3, and AF4, each with three drop probabilities, low, medium, and high.
behavior aggregate (BA) classifier	Feature that can be used to determine the forwarding treatment for each packet. The behavior aggregate classifier maps a code point to a forwarding class and loss priority. The loss priority is used later in the work flow to select one of the two drop profiles used by random early detection (RED).
best effort (BE)	CoS packet forwarding class that provides no service profile. For the BE forwarding class, loss priority is typically not carried in a code point, and random early detection (RED) drop profiles are more aggressive.
class of service (CoS)	Method of classifying traffic on a packet-by-packet basis, using information in the type-of-service (ToS) byte to assign traffic flows to different service levels.
Differentiated Services (DiffServ)	Services based on RFC 2474, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*. The DiffServ method of CoS uses the type-of-service (ToS) byte to identify different packet flows on a packet-by-packet basis. DiffServ adds a Class Selector code point (CSCP) and a DiffServ code point (DSCP).
DiffServ code point (DSCP) values	Values for a 6-bit field defined in IP packet headers that can be used to enforce class-of-service (CoS) distinctions.
drop profile	Drop probabilities for different levels of buffer fullness that are used by random early detection (RED) to determine when to drop packets from a given J-series or SRX-series device scheduling queue.

Table 174: CoS Terms *(continued)*

Term	Definition
expedited forwarding (EF)	CoS packet forwarding class that provides end-to-end service with low loss, low latency, low jitter, and assured bandwidth.
multifield (MF) classifier	Firewall filter that scans through a variety of packet fields to determine the forwarding class and loss priority for a packet and polices traffic to a specific bandwidth and burst size. Typically, a classifier performs matching operations on the selected fields against a configured value.
network control (NC)	CoS packet forwarding class that is typically high priority because it supports protocol control.
PLP bit	Packet loss priority bit. Used to identify packets that have experienced congestion or are from a transmission that exceeded a service provider's customer service license agreement. A J-series or SRX-series device can use the PLP bit as part of a congestion control strategy. The bit can be configured on an interface or in a filter.
policer	Feature that limits the amount of traffic passing into or out of an interface. It is an essential component of firewall filters that is designed to thwart denial-of-service (DoS) attacks. A policer applies rate limits on bandwidth and burst size for traffic on a particular J-series device interface.
policing	Applying rate and burst size limits to traffic on an interface.
random early detection (RED)	Gradual drop profile for a given class, used for congestion avoidance. RED attempts to anticipate congestion and reacts by dropping a small percentage of packets from the tail of the queue to prevent congestion.
rule	Guide that the device follows when applying services. A rule consists of a match direction and one or more terms.

Benefits of CoS

IP routers normally forward packets independently, without controlling throughput or delay. This type of packet forwarding, known as best-effort service, is as good as your network equipment and links allow. Best-effort service is sufficient for many traditional IP data delivery applications, such as e-mail or Web browsing. However, newer IP applications such as real-time video and audio (or voice) require lower delay, jitter, and packet loss than simple best-effort networks can provide.

CoS features allow a J-series Services Router or an SRX-series services gateway to improve its processing of critical packets while maintaining best-effort traffic flows, even during periods of congestion. Network throughput is determined by a combination of available bandwidth and delay. CoS dedicates a guaranteed minimum bandwidth to a particular service class by reducing forwarding queue delays. (The other two elements of overall network delay, serial transmission delays determined by link speeds and propagation delays determined by media type, are not affected by CoS settings.)

Normally, packets are queued for output in their order of arrival, regardless of service class. Queuing delays increase with network congestion and often result in lost packets when queue buffers overflow. CoS packet classification assigns packets to forwarding queues by service class.

Because CoS must be implemented consistently end-to-end through the network, the CoS features on the Juniper Networks device are based on IETF Differentiated Services (DiffServ) standards, to interoperate with other vendors' CoS implementations.

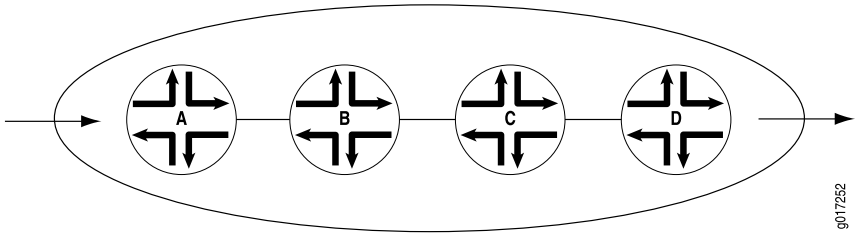
CoS Across the Network

CoS works by examining traffic entering at the edge of your network. The edge devices classify traffic into defined service groups, which allow for the special treatment of traffic across the network. For example, voice traffic can be sent across certain links, and data traffic can use other links. In addition, the data traffic streams can be serviced differently along the network path to ensure that higher-paying customers receive better service. As the traffic leaves the network at the far edge, you can reclassify the traffic.

To support CoS, you must configure each device in the network. Generally, each device examines the packets that enter it to determine their CoS settings. These settings then dictate which packets are first transmitted to the next downstream device. In addition, the devices at the edges of your network might be required to alter the CoS settings of the packets transmitting to the neighboring network.

Figure 82 on page 556 shows an example of CoS operating across an Internet Service Provider (ISP) network.

Figure 82: CoS Across the Network



In the ISP network shown in Figure 82 on page 556, Device A is receiving traffic from your network. As each packet enters, Device A examines the packet's current CoS settings and classifies the traffic into one of the groupings defined by the ISP. This definition allows Device A to prioritize its resources for servicing the traffic streams it is receiving. In addition, Device A might alter the CoS settings (forwarding class and loss priority) of the packets to better match the ISP's traffic groups. When Device B receives the packets, it examines the CoS settings, determines the appropriate traffic group, and processes the packet according to those settings. Device B then transmits the packets to Device C, which performs the same actions. Device D also examines the packets and determines the appropriate group. Because it sits at the far end of the network, the ISP might decide once again to alter the CoS settings of the packets before Device D transmits them to the neighboring network.

JUNOS CoS Components

JUNOS software supports CoS on J-series Services Routers and SRX-series services gateways as indicated in the following topics:

Code-Point Aliases

A code-point alias assigns a name to a pattern of code-point bits. You can use this name, instead of the bit pattern, when you configure other CoS components such as classifiers, drop-profile maps, and rewrite rules.

Classifiers

Packet classification refers to the examination of an incoming packet. This function associates the packet with a particular CoS servicing level. Two general types of classifiers are supported—behavior aggregate (BA) classifiers and multifield (MF) classifiers. When both BA and and MF classifications are performed on a packet, the MF classification has higher precedence.

In JUNOS software, classifiers associate incoming packets with a forwarding class (FC) and packet loss priority (PLP) and, based on the associated forwarding class, assign packets to output queues. FC and PLP associated with a packet specify the behavior of a hop, within the system, to process the packet. The per hop behavior (PHB) comprises packet forwarding, policing, scheduling, shaping, and marking. For example, a hop can put a packet in one of the priority queues according to its FC and then manage the queues by checking a packet's PLP. JUNOS software supports up to eight FCs and four PLPs.

Behavior Aggregate Classifiers

A behavior aggregate (BA) classifier operates on a packet as it enters the device. Using behavior aggregate classifiers, the device aggregates different types of traffic into a single forwarding class to receive the same forwarding treatment. The CoS value in the packet header is the single field that determines the CoS settings applied to the packet. Behavior aggregate classifiers allow you to set the forwarding class and loss priority of a packet based on the Differentiated Services (DiffServ) code point (DSCP) value, DSCP IPv4 value, IP precedence value, MPLS EXP bits, or IEEE

802.1p value. The default classifier is based on the IP precedence value. For more information, see “Default Behavior Aggregate Classifiers” on page 571.

JUNOS software performs BA classification for a packet by examining its layer 2, layer 3, and CoS-related parameters as shown in Table 175 on page 558.

Table 175: BA Classification

Layer	CoS Parameter
Layer 2	IEEE 802.1p value: User Priority
Layer 3	IPv4 precedence IPv4 Differentiated Services code point (DSCP) value

NOTE: A BA classifier evaluates Layer 2 and Layer 3 parameters independently; the results that generate from Layer 2 parameters override the results that generate from the Layer 3 parameters.

Default IP Precedence Classifier

With JUNOS software, all logical interface are automatically assigned a default IP precedence classifier when the logical interface is configured. This default traffic classifier maps IP precedence values to a forwarding class and packet loss priority as shown in Table 176 on page 558. These mapping results take effect for an ingress packet until it is further processed by another classifcation method.

Table 176: Default IP Precedence Classifier

IP Precedence CoS Values	Forwarding Class	Packet Loss Priority
000	best-effort	low
001	best-effort	high
010	best-effort	low
011	best-effort	high
100	best-effort	low
101	best-effort	high
110	network-control	low
111	network-control	high

Multifield Classifiers

A multifield (MF) classifier is a second method for classifying traffic flows. Unlike the behavior aggregate classifier, a multifield classifier can examine multiple fields in the packet—for example, the source and destination address of the packet or the source and destination port numbers of the packet. With multifield classifiers, you set the forwarding class and loss priority of a packet based on firewall filter rules.

NOTE: For a specified interface, you can configure both an MF classifier and a BA classifier without conflicts. Because the classifiers are always applied in sequential order, the BA classifier followed by the MF classifier, any BA classification result is overridden by an MF classifier, if they conflict.

JUNOS software performs MF traffic classification by directly scrutinizing multiple fields of a packet to classify a packet without having to rely upon the output of the previous BA traffic classification. JUNOS software can simultaneously check a packet's data ranging from layer 2 to layer 7 as shown in Table 177 on page 559

Table 177: MF Classification

Layer	CoS Parameter
Layer 2	IEEE 802.1Q: VLAN ID
	IEEE 802.1p: User Priority
Layer 3	IPv4: Precedence
	IPv4: DSCP
	IPv4: Source IP address
	IPv4: Destination IP address
	IPv4: Protocol
	ICMP: Code and type
Layer 4	TCP/UDP: Source port
	TCP/UDP: Destination port
	TCP: Flags
	AH/ESP: SPI
Layer 7	Not supported for this release.

Using JUNOS software, you configure an MF classifier with a firewall filter and its associated match conditions. This enables you to use any filter match criteria to locate packets that require classification. For more information on firewall filters and policies, see the *JUNOS Software Security Configuration Guide*.

Forwarding Classes

Forwarding classes allow you to group packets for transmission. Based on forwarding classes, you assign packets to output queues. The forwarding class plus the loss priority define the per-hop behavior (PHB in DiffServ) of a packet. J-series Services Routers and SRX-series services gateways support eight queues (0 through 7). For a classifier to assign an output queue (default queues 0 through 3) to each packet, it must associate the packet with one of the following forwarding classes:

- Expedited forwarding (EF)—Provides a low loss, low latency, low jitter, assured bandwidth, end-to-end service.
- Assured forwarding (AF)—Provides a group of values you can define and includes four subclasses: AF1, AF2, AF3, and AF4, each with three drop probabilities: low, medium, and high.
- Best effort (BE)—Provides no service profile. For the BE forwarding class, loss priority is typically not carried in a class-of-service (CoS) value, and random early detection (RED) drop profiles are more aggressive.
- Network Control (NC)—This class is typically high priority because it supports protocol control.

By default, the SRX-series devices support 4 queues. You can use the following CLI statement to change that setting to eight queues:

```
[edit class-of-service]
chassis {
   fpc x {
      pic y {
      max-queue-per-interface 8;
      }
      }
   }
```

The new setting will take effect when the FPC is restarted.

NOTE: Queues 4 through 7 are not mapped to forwarding classes. To use queues 4 through 7, you must create custom forwarding class names and map them to the queues. For more information, see “Forwarding Class Queue Assignments” on page 570.

In addition to BA and MF classification, the forwarding class (FC) of a packet can be directly determined by the logical interface that receives the packet. This FC of a packet can be configured using CLI commands, and if configured, this FC overrides the FC from any BA classification that was previously performed on the logical interface.

The following CLI commands can assign a forwarding class directly to packets received at a logical interface:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
forwarding-class class-name;
```

Loss Priorities

Loss priorities allow you to set the priority of dropping a packet. You can use the loss priority setting to identify packets that have experienced congestion. Typically, you mark packets exceeding some service level with a high loss priority—a greater likelihood of being dropped. You set loss priority by configuring a classifier or a policer. The loss priority is used later in the work flow to select one of the drop profiles used by random early detection (RED).

You can configure the packet loss priority (PLP) bit as part of a congestion control strategy. The PLP bit can be configured on an interface or in a filter. A packet for which the PLP bit is set has an increased probability of being dropped during congestion.

Forwarding Policy Options

CoS-based forwarding (CBF) enables you to control next-hop selection based on a packet's class of service and, in particular, the value of the IP packet's precedence bits. For example, you can specify a particular interface or next hop to carry high-priority traffic while all best-effort traffic takes some other path. CBF allows path selection based on class. When a routing protocol discovers equal-cost paths, it can pick a path at random or load-balance across the paths through either hash selection or round-robin selection.

Forwarding policy also allows you to create CoS classification overrides. For IPv4 packets, you can override the incoming CoS classification and assign the packets to a forwarding class based on their input interface, input precedence bits, or destination address. When you override the classification of incoming packets, any mappings you configured for associated precedence bits or incoming interfaces to output transmission queues are ignored.

Transmission Queues

After a packet is sent to the outgoing interface on a device, it is queued for transmission on the physical media. The amount of time a packet is queued on the device is determined by the availability of the outgoing physical media as well as the amount of traffic using the interface.

J-series Services Routers and SRX-series services gateways support queues 0 through 7. If you configure more than eight queues on a device, the commit operation fails and the device displays a detailed message stating the total number of queues available.

Schedulers

An individual device interface has multiple queues assigned to store packets temporarily before transmission. To determine the order to service the queues, the device uses a round-robin scheduling method based on priority and the queue's weighted round-robin (WRR) credits. JUNOS schedulers allow you to define the priority, bandwidth, delay buffer size, rate control status, and RED drop profiles to be applied to a particular queue for packet transmission. For more information, see "Scheduler Settings" on page 571.

You can configure per-unit scheduling (also called logical interface scheduling). Per-unit scheduling allows you to enable multiple output queues on a logical interface and associate an output scheduler with each queue.

Transmit Rate

The transmission rate determines the traffic transmission bandwidth for each forwarding class you configure. The rate is specified in bits per second (bps). Each queue is allocated some portion of the bandwidth of the outgoing interface.

This bandwidth amount can be a fixed value, such as 1 megabit per second (Mbps), a percentage of the total available bandwidth, or the rest of the available bandwidth. You can limit the transmission bandwidth to the exact value you configure, or allow it to exceed the configured rate if additional bandwidth is available from other queues (SRX-series devices do not support an exact value transmit rate). This property helps ensure that each queue receives the amount of bandwidth appropriate to its level of service.

The minimum transmit rate supported on high-speed interfaces is one-ten thousandth of the speed of that interface. For example, on a Gigabit Ethernet interface with a speed of 1000 Mbps, the minimum transmit rate is 100 Kbps (1000 Mbps x 1/10000). You can configure transmit rates in the range 3200 bps through 160,000,000,000 bps. When the configured rate is less than the minimum transmit rate, the minimum transmit rate is used instead.

NOTE: Interfaces with slower interface speeds, like T1, E1, or channelized T1/E1/ISDN PRI, cannot support minimum transmit rates because the minimum transmit rate supported on a Services Router is 3200 bps.

Transmit rate assigns the weighted round-robin (WRR) priority values within a given priority level and not between priorities. For more information, see “Transmission Scheduling” on page 575.

Delay Buffer Size

You can configure the delay buffer size to control congestion at the output stage. A delay buffer provides packet buffer space to absorb burst traffic up to a specified duration of delay. When the buffer is full, all packets are dropped.

The system calculates the buffer size for a queue based on the buffer allocation method you specify for it in the scheduler. See “Delay Buffer Size Allocation Methods” on page 648 for different buffer allocation methods and “Specifying Delay Buffer Sizes for Queues” on page 649 for buffer size calculations.

By default, all J-series Services Router interfaces other than channelized T1/E1 interfaces support a delay buffer time of 100,000 microseconds. On channelized T1/E1 interfaces, the default delay buffer time is 500,000 microseconds for clear-channel interfaces, and 1,200,000 microseconds for *N*xDS0 interfaces.

On J-series Services Routers, you can configure larger delay buffers on channelized T1/E1 interfaces. Larger delay buffers help these slower interfaces to avoid congestion

and packet dropping when they receive large bursts of traffic. For more information, see "Configuring Large Delay Buffers with a Configuration Editor" on page 647.

Scheduling Priority

Scheduling priority determines the order in which an output interface transmits traffic from the queues, thus ensuring that queues containing important traffic are provided better access to the outgoing interface.

The queues for an interface are divided into sets based on their priority. Each set contains queues of the same priority. The device examines the sets in descending order of priority. If at least one queue in a set has a packet to transmit, the device selects that set. If multiple queues in the set have packets to transmit, the device selects a queue from the set according to the weighted round-robin (WRR) algorithm that operates within the set.

The packets in a queue are transmitted based on the configured scheduling priority, the transmit rate, and the available bandwidth. For more information, see "Transmission Scheduling" on page 575.

Shaping Rate

Shaping rates control the maximum rate of traffic transmitted on an interface. You can configure the shaping rate so that the interface transmits less traffic than it is physically capable of carrying.

You can configure shaping rates on logical interfaces. By default, output scheduling is not enabled on logical interfaces. Logical interface scheduling (also called per-unit scheduling) allows you to enable multiple output queues on a logical interface and associate an output scheduler and shaping rate with the queues.

By default, the logical interface bandwidth is the average of unused bandwidth for the number of logical interfaces that require default bandwidth treatment. You can specify a peak bandwidth rate in bits per second (bps), either as a complete decimal number or as a decimal number followed by the abbreviation *k* (1000), *m* (1,000,000), or *g* (1,000,000,000). The range is from 1000 through 32,000,000,000 bps.

For low-speed interfaces, the queue-limit values might become lower than the interface MTU so that traffic with large packets can no longer pass through some of the queues. If you want larger-sized packets to flow through, set the buffer-size configuration in the scheduler to a larger value. For more accuracy, the 100-ms queue-limit values are calculated based on shaper rates and not on interface rates.

RED Drop Profiles

A drop profile is a feature of the random early detection (RED) process that allows packets to be dropped before queues are full. Drop profiles are composed of two main values—the queue fullness and the drop probability. The queue fullness represents percentage of memory used to store packets in relation to the total amount that has been allocated for that queue. The drop probability is a percentage value that correlates to the likelihood that an individual packet is dropped from the network. These two variables are combined in a graph-like format.

When a packet reaches the head of the queue, a random number between 0 and 100 is calculated by the device. This random number is plotted against the drop profile having the current queue fullness of that particular queue. When the random number falls above the graph line, the packet is transmitted onto the physical media. When the number falls below the graph line, the packet is dropped from the network.

When you configure the RED drop profile on an interface, the queue no longer drops packets from the tail of the queue (the default). Rather, packets are dropped after they reach the head of the queue.

You specify drop probabilities in the drop profile section of the class-of-service (CoS) configuration hierarchy and reference them in each scheduler configuration. For each scheduler, you can configure multiple separate drop profiles, one for each combination of loss priority (low, medium-low, medium-high, or high) and IP transport protocol (TCP or non-TCP).

You can configure a maximum of 32 different drop profiles.

To configure RED drop profiles, include the following statements at the [edit class-of-service] hierarchy level of the configuration:

```
[edit class-of-service]
drop-profiles {
   profile-name {
      fill-level percentage drop-probability percentage;
      interpolate {
         drop-probability [ values ];
         fill-level [ values ];
      }
   }
}
```

Default Drop Profiles

By default, if you configure no drop profiles, RED is still in effect and functions as the primary mechanism for managing congestion. In the default RED drop profile, when the fill level is 0 percent, the drop probability is 0 percent. When the fill level is 100 percent, the drop probability is 100 percent.

Virtual Channels

On J-series Services Routers, you can configure virtual channels to limit traffic sent from a corporate headquarters to branch offices. Virtual channels might be required when the headquarters site has an expected aggregate bandwidth higher than that of the individual branch offices. The router at the headquarters site must limit the traffic sent to each branch office router to avoid oversubscribing their links.

You configure virtual channels on a logical interface. Each virtual channel has a set of eight queues with a scheduler and an optional shaper. You can use an output firewall filter to direct traffic to a particular virtual channel. For example, a filter can direct all traffic with a destination address for branch office 1 to virtual channel 1, and all traffic with a destination address for branch office 2 to virtual channel 2.

Although a virtual channel group is assigned to a logical interface, a virtual channel is not the same as a logical interface. The only features supported on a virtual channel are queuing, packet scheduling, and accounting. Rewrite rules and routing protocols apply to the entire logical interface.

Policers for Traffic Classes

Policers allow you to limit traffic of a certain class to a specified bandwidth and burst size. Packets exceeding the policer limits can be discarded, or can be assigned to a different forwarding class, a different loss priority, or both. You define policers with firewall filters that can be associated with input or output interfaces.

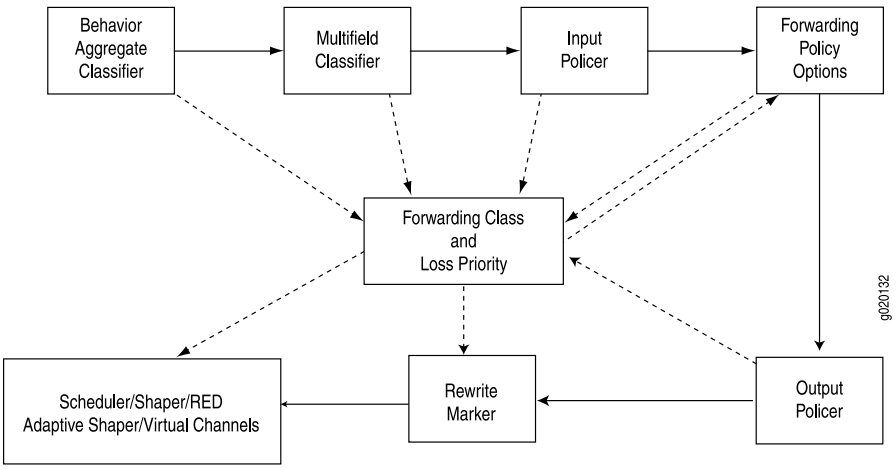
Rewrite Rules

A rewrite rule modifies the appropriate CoS bits in an outgoing packet. Modification of CoS bits allows the next downstream device to classify the packet into the appropriate service group. Rewriting or marking outbound packets is useful when the device is at the border of a network and must alter the CoS values to meet the policies of the targeted peer.

How CoS Components Work

On J-series Services Routers and SRX-series services gateways, you configure CoS functions using different components. These components are configured individually or in a combination to define particular CoS services. Figure 83 on page 565 displays the relationship of different CoS components to each other and illustrates the sequence in which they interact. “JUNOS CoS Components” on page 557 defines the components and explains their use.

Figure 83: Packet Flow Through J-series or SRX-series Device



Each box in Figure 83 on page 565 represents a CoS component. The solid lines show the direction of packet flow in a device. The upper row indicates an incoming packet, and the lower row an outgoing packet. The dotted lines show the inputs and outputs of particular CoS components. For example, the forwarding class and loss priority

are outputs of behavior aggregate classifiers and multifield classifiers and inputs for rewrite markers and schedulers.

Typically, only a combination of some components shown in Figure 83 on page 565 (not all) is used to define a CoS service offering. For example, if a packet's class is determined by a behavior aggregate classifier, it is associated with a forwarding class and loss priority and does not need further classification by the multifield classifier.

CoS Process on Incoming Packets

Classifiers and policers perform the following operations on incoming packets:

1. A classifier examines an incoming packet and assigns a forwarding class and loss priority to it.
2. Based on the forwarding class, the packet is assigned to an outbound transmission queue.
3. Input policers meter traffic to see if traffic flow exceeds its service level. Policers might discard, change the forwarding class and loss priority, or set the PLP bit of a packet. A packet for which the PLP bit is set has an increased probability of being dropped during congestion.

CoS Process on Outgoing Packets

The scheduler map and rewrite rules perform the following operations on outgoing packets:

1. Scheduler maps are applied to interfaces and associate the outgoing packets with a scheduler and a forwarding class.
2. The scheduler defines how the packet is treated in the output transmission queue based on the configured transmit rate, buffer size, priority, and drop profile.
   - The buffer size defines the period for which the packet is stored during congestion.
   - The scheduling priority and transmit rate determine the order in which the packet is transmitted.
   - The drop profile defines how aggressively to drop packets that are using a particular scheduler.
3. Output policers meter traffic and might change the forwarding class and loss priority of a packet if a traffic flow exceeds its service level.
4. The rewrite rule writes information to the packet (for example, EXP or DSCP bits) according to the forwarding class and loss priority of the packet.

Default CoS Settings

Even when you do not configure any CoS settings on your routing platform, the software performs some CoS functions to ensure that user traffic and protocol packets are forwarded with minimum delay when the network is experiencing congestion. Some default mappings are automatically applied to each logical interface that you

configure. Other default mappings, such as explicit default classifiers and rewrite rules, are in operation only if you explicitly associate them with an interface.

You can display default CoS settings by running the show class-of-service operational mode command.

This section contains the following topics:

Default CoS Values and Aliases

Table 178 on page 568 shows the default mappings between the bit values and standard aliases.

Table 178: Well-Known CoS Aliases and Default CoS Values

CoS Value Type	Alias	CoS Value
DSCP and DSCP IPv6	ef	101110
	af11	001010
	af12	001100
	af13	001110
	af21	010010
	af22	010100
	af23	010110
	af31	011010
	af32	011100
	af33	011110
	af41	100010
	af42	100100
	af43	100110
	be	000000
	cs1	001000
	cs2	010000
	cs3	011000
	cs4	100000
	cs5	101000
	nc1/cs6	110000
	nc2/cs7	111000

Table 178: Well-Known CoS Aliases and Default CoS Values *(continued)*

CoS Value Type	Alias	CoS Value
MPLS EXP	be	000
	be1	001
	ef	010
	ef1	011
	af11	100
	af12	101
	nc1/cs6	110
	nc2/cs7	111
IEEE 802.1	be	000
	be1	001
	ef	010
	ef1	011
	af11	100
	af12	101
	nc1/cs6	110
	nc2/cs7	111
IP precedence	be	000
	be1	001
	ef	010
	ef1	011
	af11	100
	af12	101
	nc1/cs6	110
	nc2/cs7	111

Forwarding Class Queue Assignments

J-series Services Routers and SRX-series services gateways have eight queues built into the hardware. By default, four queues are assigned to four forwarding classes. Table 179 on page 570 shows the four default forwarding classes and queues that Juniper Networks classifiers assign to packets based on the CoS values in arriving packet headers. Queues 4 through 7 have no default assignments to forwarding classes. To use queues 4 through 7, you must create custom forwarding class names and assign them to the queues. For more information about how to assign queues to forwarding classes, see "Configuring Class of Service" on page 579.

By default, all incoming packets, except the IP protocol control packets, are assigned to the forwarding class associated with queue 0. All IP protocol control packets are assigned to the forwarding class associated with queue 3.

Table 179 on page 570 displays the default assignments of forwarding classes to queues.

Table 179: Default Forwarding Class Queue Assignments

Forwarding Queue	Forwarding Class	Forwarding Class Description
Queue 0	best-effort (BE)	The Juniper Networks device does not apply any special CoS handling to packets with 000000 in the DiffServ field, a backward compatibility feature. These packets are usually dropped under congested network conditions.
Queue 1	expedited-forwarding (EF)	The Juniper Networks device delivers assured bandwidth, low loss, low delay, and low delay variation (jitter) end-to-end for packets in this service class. Devices accept excess traffic in this class, but in contrast to assured forwarding, out-of-profile expedited-forwarding packets can be forwarded out of sequence or dropped.
Queue 2	assured-forwarding (AF)	The Juniper Networks device offers a high level of assurance that the packets are delivered as long as the packet flow from the customer stays within a certain service profile that you define. The device accepts excess traffic, but applies a random early detection (RED) drop profile to determine whether the excess packets are dropped and not forwarded. Three drop probabilities (low, medium, and high) are defined for this service class.
Queue 3	network-control (NC)	The Juniper Networks device delivers packets in this service class with a low priority. (These packets are not delay sensitive.) Typically, these packets represent routing protocol hello or keepalive messages. Because loss of these packets jeopardizes proper network operation, delay is preferable to discard.

Scheduler Settings

Each forwarding class has an associated scheduler priority. Only two forwarding classes, best-effort and network-control (queue 0 and queue 3), are used in the JUNOS default scheduler configuration.

By default, the best-effort forwarding class (queue 0) receives 95 percent, and the network-control (queue 3) receives 5 percent of the bandwidth and buffer space for the output link. The default drop profile causes the buffer to fill and then discard all packets until it again has space.

The expedited-forwarding and assured-forwarding classes have no schedulers, because by default no resources are assigned to queue 1 and queue 2. However, you can manually configure resources for expedited-forwarding and assured-forwarding.

By default, each queue can exceed the assigned bandwidth if additional bandwidth is available from other queues. When a forwarding class does not fully use the allocated transmission bandwidth, the remaining bandwidth can be used by other forwarding classes if they receive a larger amount of offered load than the bandwidth allocated. If you do not want a queue to use any leftover bandwidth, you must configure it for strict allocation. For more information, see "Configuring Strict High Priority for Queuing with a Configuration Editor" on page 640.

The device uses the following default scheduler settings. You can modify these settings through configuration. For instructions, see "Configuring Class of Service" on page 579.

```
[edit class-of-service]
schedulers {
    network-control {
        transmit-rate percent 5;
        buffer-size percent 5;
        priority low;
        drop-profile-map loss-priority any protocol any drop-profile terminal;
    }
    best-effort {
        transmit-rate percent 95;
        buffer-size percent 95;
        priority low;
        drop-profile-map loss-priority any protocol any drop-profile terminal;
    }
}
drop-profiles {
    terminal {
        fill-level 100 drop-probability 100;
    }
}
```

Default Behavior Aggregate Classifiers

Table 180 on page 572 shows the forwarding class and packet loss priority (PLP) that are assigned by default to each well-known DSCP. Although several DSCPs map to the expedited-forwarding (ef) and assured-forwarding (af) classes, by default no resources are assigned to these forwarding classes. All af classes other than af1*x* are mapped

to best-effort, because RFC 2597, *Assured Forwarding PHB Group*, prohibits a node from aggregating classes. Assignment to best-effort implies that the node does not support that class.

You can modify the default settings through configuration. For instructions, see "Configuring Class of Service" on page 579.

Table 180: Default Behavior Aggregate Classification

DSCP and DSCP IPv6 Alias	Forwarding Class	Packet Loss Priority (PLP)
ef	expedited-forwarding	low
af11	assured-forwarding	low
af12	assured-forwarding	high
af13	assured-forwarding	high
af21	best-effort	low
af22	best-effort	low
af23	best-effort	low
af31	best-effort	low
af32	best-effort	low
af33	best-effort	low
af41	best-effort	low
af42	best-effort	low
af43	best-effort	low
be	best-effort	low
cs1	best-effort	low
cs2	best-effort	low
cs3	best-effort	low
cs4	best-effort	low
cs5	best-effort	low
nc1/cs6	network-control	low
nc2/cs7	network-control	low
other	best-effort	low

Defining BA Classifiers

You can override the default IP precedence classifier by defining a classifier and applying it to a logical interface. To define new classifiers for all code point types, include the classifiers statement at the [edit class-of-service] hierarchy level:

```
[edit class-of-service]
classifiers {
   (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) classifier-name {
      import [classifier-name | default];
      forwarding-class class-name {
         loss-priority level {
            code-points [ aliases ] [ 6-bit-patterns ];
         }
      }
   }
}
```

The map sets the forwarding class and PLP for a specific set of code-point aliases and bit patterns. The inputs of the map are code-point aliases and bit patterns. The outputs of the map are the forwarding class and the PLP.

The classifiers work as follows:

- dscp—Handles incoming IPv4 packets.
- dscp-ipv6—Handles incoming IPv6 packets. (IPv6 is not supported in this release of the software.)
- exp—Handles MPLS packets using Layer 2 headers.
- ieee-802.1—Handles Layer 2 CoS.
- inet-precedence—Handles incoming IPv4 packets. IP precedence mapping requires only the upper three bits of the DSCP field.

A classifier takes a specified bit pattern as either the literal pattern or as a defined alias and attempts to match it to the type of packet arriving on the interface. If the information in the packet's header matches the specified pattern, the packet is sent to the appropriate queue, defined by the forwarding class associated with the classifier.

The code-point aliases and bit patterns are the input for the map. The loss priority and forwarding class are outputs of the map. In other words, the map sets the PLP and forwarding class for a given set of code-point aliases and bit patterns.

Applying a BA Classifier to a Logical Interface

You can apply the classification map to a logical interface by including the classifiers statement at the [edit class-of-service interfaces *interface-name* unit *logical-unit-number*] hierarchy level:

```
[edit class-of-service interfaces interface-name interface-name unit logical-unit-number]
classifiers (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) (classifier-name |
   default);
```

You can use interface wildcards for *interface-name* and *logical-unit-number*.

CoS Value Rewrites

Typically, a device rewrites CoS values in outgoing packets on the outbound interfaces of an edge device, to meet the policies of the targeted peer. After reading the current forwarding class and loss priority information associated with the packet, the transmitting device locates the chosen CoS value from a table, and writes this CoS value into the packet header.

For instructions for configuring rewrite rules, see "Configuring and Applying Rewrite Rules" on page 611.

Sample Behavior Aggregate Classification

Table 181 on page 574 shows the device forwarding classes associated with each well-known DSCP code point and the resources assigned to their output queues for a sample DiffServ CoS implementation. This example assigns expedited forwarding to queue 1 and a subset of the assured forwarding classes (af1*x*) to queue 2, and distributes resources among all four forwarding classes.

Other DiffServ-based implementations are possible. For configuration information, see "Configuring Class of Service" on page 579.

Table 181: Sample Behavior Aggregate Classification Forwarding Classes and Queues

DSCP and DSCP IPv6 Alias	DSCP and DSCP IPv6 Bits	Forwarding Class	PLP	Queue
ef	101110	expedited-forwarding	low	1
af11	001010	assured-forwarding	low	2
af12	001100	assured-forwarding	high	2
af13	001110	assured-forwarding	high	2
af21	010010	best-effort	low	0
af22	010100	best-effort	low	0
af23	010110	best-effort	low	0
af31	011010	best-effort	low	0
af32	011100	best-effort	low	0
af33	011110	best-effort	low	0
af41	100010	best-effort	low	0
af42	100100	best-effort	low	0
af43	100110	best-effort	low	0
be	000000	best-effort	low	0

Table 181: Sample Behavior Aggregate Classification Forwarding Classes and Queues *(continued)*

DSCP and DSCP IPv6 Alias	DSCP and DSCP IPv6 Bits	Forwarding Class	PLP	Queue
cs1	0010000	best-effort	low	0
cs2	010000	best-effort	low	0
cs3	011000	best-effort	low	0
cs4	100000	best-effort	low	0
cs5	101000	best-effort	low	0
nc1/cs6	110000	network-control	low	3
nc2/cs7	111000	network-control	low	3
other	—	best-effort	low	0

Transmission Scheduling

The packets in a queue are transmitted based on their transmission priority, transmit rate, and the available bandwidth.

By default, each queue can exceed the assigned bandwidth if additional bandwidth is available from other queues. When a forwarding class does not fully use the allocated transmission bandwidth, the remaining bandwidth can be used by other forwarding classes if they receive a larger amount of offered load than the bandwidth allocated. A queue receiving traffic within its bandwidth configuration is considered to have positive bandwidth credit, and a queue receiving traffic in excess of its bandwidth allocation is considered to have negative bandwidth credit.

A queue with positive credit does not need to use leftover bandwidth, because it can use its own allocation. For such queues, packets are transmitted based on the priority of the queue, with packets from higher-priority queues transmitting first. The transmit rate is not considered during transmission. In contrast, a queue with negative credit needs a share of the available leftover bandwidth.

The leftover bandwidth is allocated to queues with negative credit in proportion to the configured transmit rate of the queues within a given priority set. The queues for an interface are divided into sets based on their priority. For more information, see “Scheduling Priority” on page 563. If no transmit rate is configured, each queue in the set receives an equal percentage of the leftover bandwidth. However, if a transmit rate is configured, each queue in the set receives the configured percentage of the leftover bandwidth.

Table 182 on page 576 shows a sample configuration of priority and transmit rate on six queues. The total available bandwidth on the interface is 100 Mbps.

Table 182: Sample Transmission Scheduling

Queue	Scheduling Priority	Transmit Rate	Incoming Traffic
0	Low	10 %	20 Mbps
1	High	20 %	20 Mbps
2	High	30 %	20 Mbps
3	Low	30 %	20 Mbps
4	Medium-high	No transmit rate configured	10 Mbps
5	Medium-high	No transmit rate configured	20 Mbps

In this example, queues are divided into three sets based on their priority:

- High priority set—Consists of queue 1 and queue 2. Packets use 40 Mbps (20 + 20) of the available bandwidth (100 Mbps) and are transmitted first. Because of positive credit, the configured transmit rate is not considered.
- Medium-high priority set—Consists of queue 4 and queue 5. Packets use 30 Mbps (10 + 20) of the remaining 60 Mbps bandwidth. Because of positive credit, the transmit rate is not considered. If the queues had negative credit, they would receive an equal share of the leftover bandwidth because no transmit rate is configured.
- Low priority set—Consists of queue 0 and queue 3. Packets share the 20 Mbps of leftover bandwidth based on the configured transmit rate. The distribution of bandwidth is in proportion to the assigned percentages. Because the total assigned percentage is 40 (10 + 30), each queue receives a share of bandwidth accordingly. Thus queue 0 receives 5 Mbps (10/40 x 20), and queue 3 receives 15 Mbps (30/40 x 20).

CoS Queuing for Tunnels

A tunnel interface in a J-series Services Router running JUNOS software supports many of the same CoS features as a physical interface. A tunnel interface is a virtual or logical interface on a J-series router. It creates a virtual point-to-point link between two J-series routers at remote points over an IP network.

For example, you can configure CoS features for generic routing encapsulation (GRE) and IP-IP tunnel interfaces. Tunneling protocols encapsulate packets inside a transport protocol.

GRE or IP-IP tunnels are used with services like IPsec and NAT to set up point-to-point VPNs. JUNOS software allows you to enable CoS queuing, scheduling, and shaping for traffic exiting through these tunnel interfaces.

Benefits of CoS Queuing on Tunnel Interfaces

On a J-series Services Router, CoS queuing enabled for tunnel interfaces allows you to

- Segregate tunnel traffic.

  Each tunnel can be shaped so that a tunnel with low-priority traffic cannot swamp other tunnels that carry high-priority traffic.

  Traffic for one tunnel does not impact traffic on other tunnels.

- Control tunnel bandwidth.

  Traffic through various tunnels is limited to not exceed a certain bandwidth.

  For example, suppose you have three tunnels to three remote sites through a single WAN interface. You can select CoS parameters for each tunnel such that traffic to some sites gets more bandwidth than traffic to other sites.

- Customize CoS policies.

  You can apply different queuing, scheduling, and shaping policies to different tunnels based on user requirements. Each tunnel can be configured with different scheduler maps, different queue depths, and so on. Customization allows you to configure granular CoS policy providing for better control over tunnel traffic.

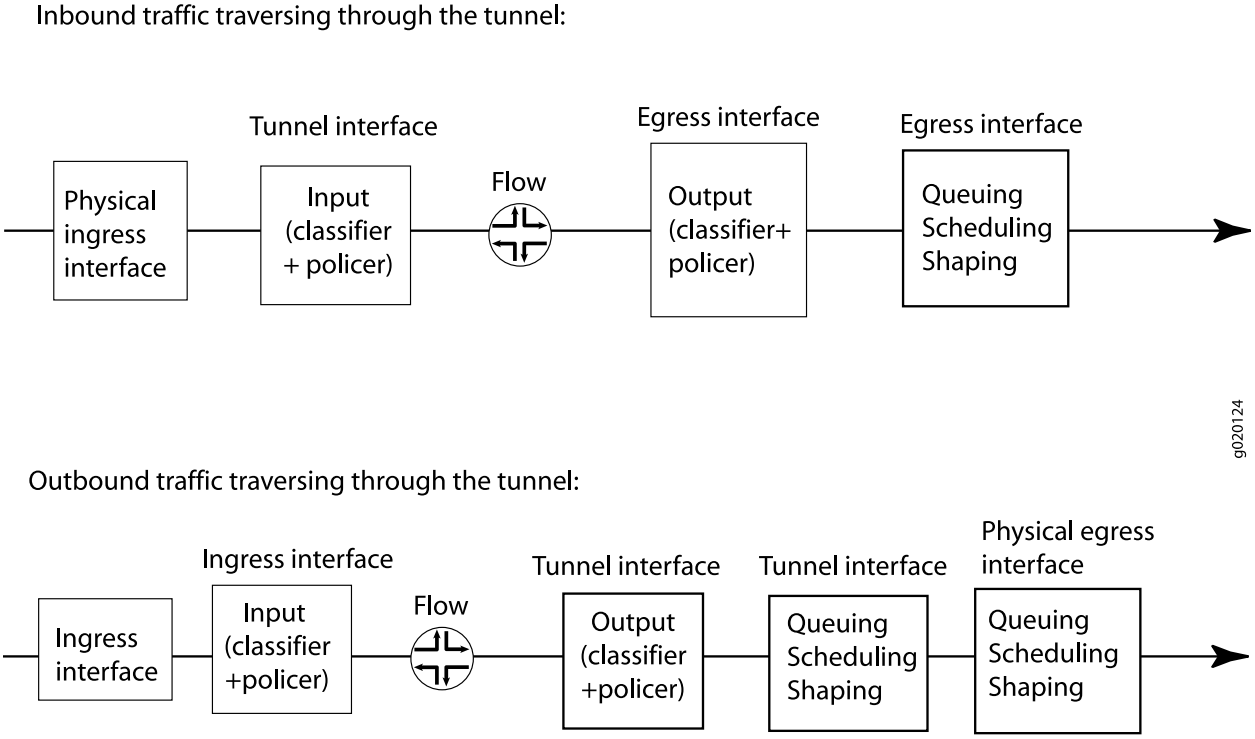
- Prioritize traffic before it enters a tunnel.

  For example, CoS queuing avoids having low-priority packets scheduled ahead of high-priority packets when the interface speed is higher than the tunnel traffic speed. This feature is most useful when combined with IPsec. Typically, IPsec processes packets in a FIFO manner. However, with CoS queuing each tunnel can prioritize high-priority packets over low-priority packets. Also, each tunnel can be shaped, so that a tunnel with low-priority traffic does not swamp tunnels carrying high-priority traffic.

How CoS Queuing Works

Figure 84 on page 578 shows CoS-related processing that occurs for traffic entering and exiting a tunnel. For information on flow-based packet processing, see the *JUNOS Software Security Configuration Guide*

Figure 84: CoS Processing for Tunnel Traffic



Limitations on CoS Shapers for Tunnel Interfaces

On a J-series Services Router, when defining a CoS shaping rate on a tunnel interface, be aware of the following restrictions:

- The shaping rate on the tunnel interface must be less than that of the physical egress interface.
- The shaping rate only measures the packet size that includes the Layer 3 packet with GRE or IP-IP encapsulation. The Layer 2 encapsulation added by the physical interface is not factored into the shaping rate measurement.
- The CoS behavior works as expected only when the physical interface carries the shaped GRE or IP-IP tunnel traffic alone. If physical interface carries other traffic, thereby lowering the available bandwidth for tunnel interface traffic, the CoS features do not work as expected.
- You cannot configure a logical interface shaper and a virtual circuit shaper simultaneously on the router. If virtual circuit shaping is desired, do not define a logical interface shaper. Instead, define a shaping rate for all the virtual circuits.

Chapter 27

Configuring Class of Service

You configure class of service (CoS) when you need to override the default packet forwarding behavior of a J-series or SRX-series device—especially in the three areas identified in Table 183 on page 579.

Table 183: Reasons to Configure Class of Service (Cos)

Default Behavior to Override with CoS	CoS Configuration Area
Packet classification—By default, the J-series or SRX-series device does not use behavior aggregate (BA) classifiers to classify packets. Packet classification applies to incoming traffic.	Classifiers
Scheduling queues—By default, the J-series or SRX-series device has only two queues enabled. Scheduling queues apply to outgoing traffic.	Schedulers
Packet headers—By default, the J-series or SRX-series device does not rewrite CoS bits in packet headers. Rewriting packet headers applies to outgoing traffic.	Rewrite rules

You can use either J-Web Quick Configuration or a configuration editor to configure CoS. This chapter contains the following topics. For more information about CoS, see the *JUNOS Class of Service Configuration Guide*.

Before You Begin

Before you begin configuring a J-series or SRX-series device for CoS, complete the following tasks:

- If you do not already have a basic understanding of CoS, read “Class-of-Service Overview” on page 553.
- Determine whether the device needs to support different traffic streams, such as voice or video. If so, CoS helps to make sure this traffic receives more than basic best-effort packet delivery service.
- Determine whether the device is directly attached to any applications that send CoS-classified packets. If no sources are enabled for CoS, you must configure and apply rewrite rules on the interfaces facing the sources.
- Determine whether the device must support assured forwarding (AF) classes. Assured forwarding usually requires random early detection (RED) drop profiles to be configured and applied.
- Determine whether the device must support expedited forwarding (EF) classes with a policer. Policers require you to apply a burst size and bandwidth limit to the traffic flow, and set a consequence for packets that exceed these limits—usually a high loss priority, so that packets exceeding the policer limits are discarded first.

Configuring CoS with Quick Configuration

The Class of Service Quick Configuration pages allow you to configure most of the JUNOS CoS components for the IPv4 and MPLS traffic on a J-series or SRX-series device. You can configure forwarding classes for transmitting packets, define which packets are placed into each output queue, schedule the transmission service level for each queue, and manage congestion using a random early detection (RED) algorithm. After defining the CoS components you must assign classifiers to the required physical and logical interfaces.

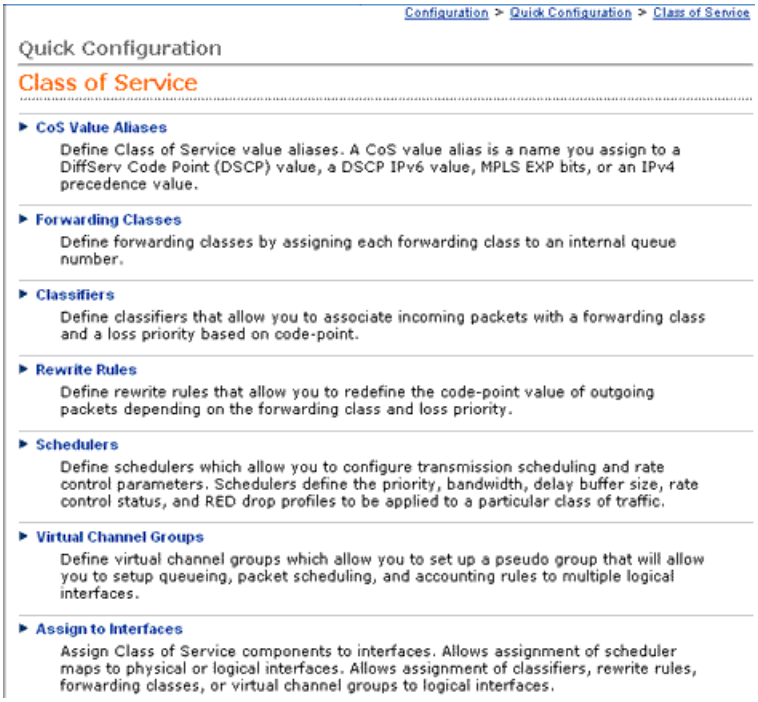
This section contains the following topics:

Defining CoS Components

Using the Class of Service Quick Configuration pages, you can configure various CoS components individually or in combination to define particular CoS services. For a description of different CoS components, see “JUNOS CoS Components” on page 557.

Figure 85 on page 581 shows the initial Quick Configuration page for CoS that displays the CoS components.

Figure 85: Initial Class of Service Quick Configuration Page



To configure CoS components with Quick Configuration:

1. In the J-Web interface, select **Configuration > Quick Configuration > Class of Service**.
2. On the Class of Service Quick Configuration page, select one of the following options depending on the CoS component that you want to define. Enter information into the pages as described in the respective table:
   - To define or edit CoS value aliases, select **CoS Value Aliases** and see "Defining CoS Value Aliases" on page 582.
   - To define or edit forwarding classes and assign queues, select **Forwarding Classes** and see "Defining Forwarding Classes" on page 584.
   - To define or edit classifiers, select **Classifiers** and see "Defining Classifiers" on page 585.
   - To define or edit rewrite rules, select **Rewrite Rules** and see "Defining Rewrite Rules" on page 587.
   - To define or edit schedulers, select **Schedulers** and see "Defining Schedulers" on page 589.
   - To define or edit virtual channel groups, select **Virtual Channel Groups** and see "Defining Virtual Channel Groups" on page 595.
3. Click one of the following buttons after completing configuration on any Quick Configuration page:

- To apply the configuration and stay in the current Quick Configuration page, click **Apply**.
- To apply the configuration and return to the previous Quick Configuration page, click **OK**.
- To cancel your entries and return to the previous Quick Configuration page, click **Cancel**.

4. Go on to one of the following procedures:
   - To assign CoS components to interfaces, see "Assigning CoS Components to Interfaces" on page 596.
   - To verify the CoS configuration, see "Verifying a CoS Configuration" on page 682.

Defining CoS Value Aliases

Figure 86 on page 582 shows the initial Quick Configuration page for defining aliases for CoS values, and Table 184 on page 583 describes the related fields. By defining aliases you can assign meaningful names to a particular set of bit values and refer to them when configuring CoS components. For more information about CoS values and aliases, see "Default CoS Values and Aliases" on page 567.

Figure 86: CoS Value Aliases Quick Configuration Page

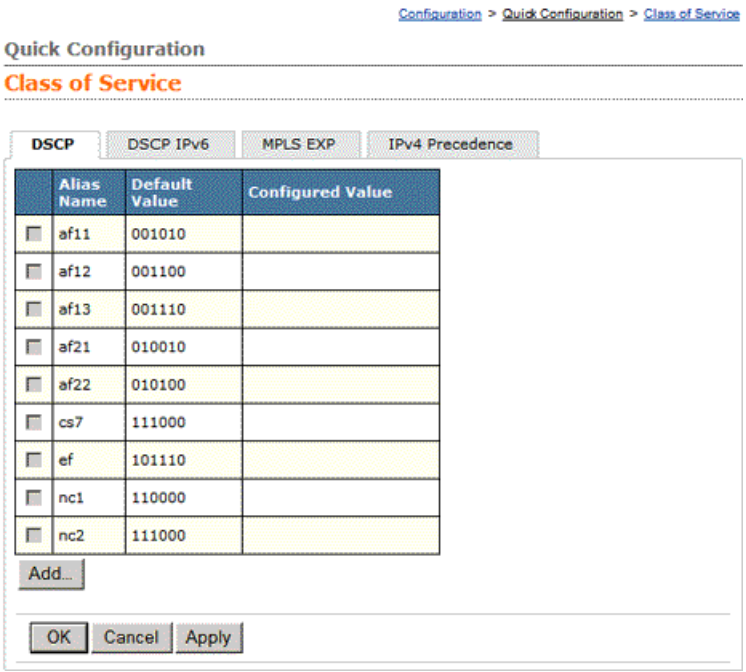


Table 184: CoS Value Aliases Quick Configuration Pages Summary

Field	Function	Your Action
CoS Value Alias Summary		
DSCP	Allows you to define aliases for DiffServ code point (DSCP) IPv4 values. You can refer to these aliases when you configure classes and define classifiers.	To define an alias for a DSCP value, click **DSCP**.
DSCP IPv6	Allows you to define aliases for DSCP IPv6 values. You can refer to these aliases when you configure classes and define classifiers.	To define an alias for a DSCP IPv6 value, click **DSCP IPv6**.
MPLS EXP	Allows you to define aliases for MPLS experimental (EXP) bits. You can map MPLS EXP bits to the device forwarding classes.	To define an alias for a set of MPLS EXP bits, click **MPLS EXP**.
IPv4 Precedence	Allows you to define aliases for IPv4 precedence values. Precedence values are modified in the IPv4 type-of-service (TOS) field and mapped to values that correspond to levels of service.	To define an alias for an IPv4 precedence value, click **IPv4 Precedence**.
Alias Name	Displays names given to CoS values—for example, af11 or be.	None.
Default Value	Displays the default values mapped to standard aliases. For example, ef (expedited forwarding) is a standard alias for DSCP bits 101110. You cannot delete default values. The check box next to these values is unavailable.	None.
Configured Value	Displays the CoS values that you have assigned to specific aliases. You can delete a configured alias.	None.
Add	Opens a page that allows you to define CoS value aliases.	To add a CoS value alias, click **Add**.
Delete	Allows you to delete a configured CoS value alias. You cannot delete a default alias.	To delete a CoS value alias, select the check box next to it and click **Delete**.
Add a CoS Value Alias		
CoS Value Alias	Assigns a name to a CoS value. A CoS value can be of different types—DSCP, DSCP IPv6, IP precedence, or MPLS EXP.	To define an alias for a CoS value, type a name—for example, my1.

Table 184: CoS Value Aliases Quick Configuration Pages Summary *(continued)*

Field	Function	Your Action
CoS Value Alias Bits	Specifies the CoS value for which an alias is defined. Changing this value alters the behavior of all classifiers that refer to this alias.	To specify a CoS value, type it in an appropriate format: ■ For DSCP and DSCP IPv6 CoS values, use the format xxxxxx, where x is 1 or 0—for example, **101110**. ■ For MPLS EXP and IP precedence CoS values, use the format xxx, where x is 1 or 0—for example, **111**.

Defining Forwarding Classes

Figure 87 on page 584 shows the initial Quick Configuration page for defining forwarding classes and assigning them to queues, and Table 185 on page 584 describes the related fields. By assigning a forwarding class to a queue number, you affect the scheduling and marking of a packet as it transits a J-series Services Router or an SRX-series services gateway. For more information about forwarding classes and queues, see "JUNOS CoS Components" on page 557.

Figure 87: Forwarding Classes Quick Configuration Page

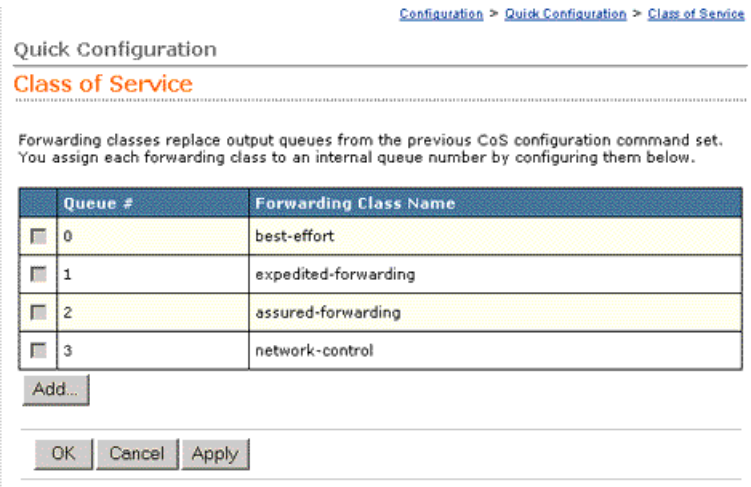


Table 185: Forwarding Classes Quick Configuration Pages Summary

Field	Function	Your Action
Forwarding Class Summary		

Table 185: Forwarding Classes Quick Configuration Pages Summary *(continued)*

Field	Function	Your Action
Queue #	Displays internal queue numbers to which forwarding classes are assigned. By default, if a packet is not classified, it is assigned to the class associated with queue 0. Allows you to edit an assigned forwarding class.	To edit an assigned forwarding class, click the queue number to which the class is assigned.
Forwarding Class Name	Displays the forwarding class names assigned to specific internal queue numbers. By default, four forwarding classes are assigned to queue numbers 0 through 3.	None.
Add	Opens a page that allows you to assign forwarding classes to internal queue numbers.	To add a forwarding class, click **Add**.
Delete	Deletes an internal queue number and the forwarding class assigned to it.	To delete a queue number, click the check box next to it and click **Delete**.
Add a Forwarding Class/Edit Forwarding Class Queue #		
Queue #	Specifies the internal queue number to which a forwarding class is assigned.	To specify an internal queue number, type an integer from 0 through 7, as supported by your platform.
Forwarding Class Name	Specifies the forwarding class name assigned to the internal queue number.	To assign a forwarding class name to a queue, type the name—for example, be-class.

Defining Classifiers

Figure 88 on page 585 shows the initial Quick Configuration page for defining classifiers, and Table 186 on page 586 describes the related fields. Classifiers examine the CoS value or alias of an incoming packet and assign it a level of service by setting its forwarding class and loss priority. For more information about classifiers, see “Default Behavior Aggregate Classifiers” on page 571.

Figure 88: Classifiers Quick Configuration Page

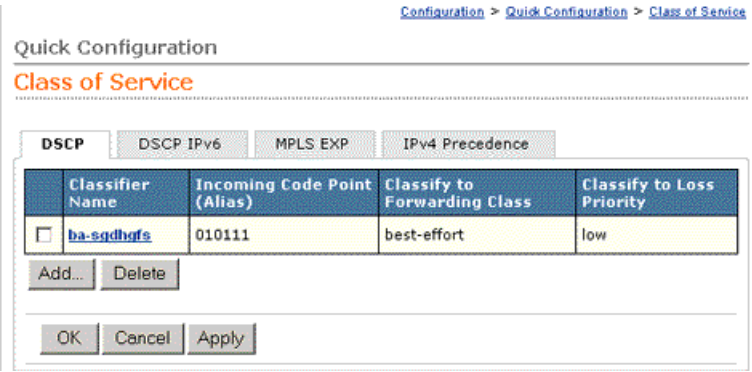


Table 186: Classifiers Quick Configuration Page Summary

Field	Function	Your Action
Classifier Summary		
DSCP	Allows you to define classifiers for DSCP IPv4 values.	To define a classifier for a DSCP code point value, click **DSCP**.
DSCP IPv6	Allows you to define classifiers for DSCP IPv6 values.	To define a classifier for a DSCP IPv6 value, click **DSCP IPv6**.
MPLS EXP	Allows you to define classifiers for MPLS experimental (EXP) bits.	To define a classifier for a set of MPLS EXP bits, click **MPLS EXP**.
IPv4 Precedence	Allows you to define classifiers for IPv4 precedence values.	To define a classifier for an IP precedence value, click **IPv4 Precedence**.
Classifier Name	Displays the names of classifiers. Allows you to edit a specific classifier.	To edit a classifier, click its name.
Incoming Code Point (Alias)	Displays CoS values and aliases to which forwarding class and loss priority are mapped.	None.
Classify to Forwarding Class	Displays forwarding classes that are assigned to specific CoS values and aliases of a classifier.	None.
Classify to Loss Priority	Displays loss priorities that are assigned to specific CoS values and aliases of a classifier.	None.
Add	Opens a page that allows you to define classifiers.	To add a classifier, click **Add**.
Delete	Deletes a specified classifier.	To delete a classifier, locate the classifier, select the check box next to it, and click **Delete**.
Add a Classifier/Edit Classifier		
Classifier Name	Specifies the name for a classifier.	To name a classifier, type the name—for example, ba-classifier.
Classifier Code Point Mapping	Sets the forwarding classes and the packet loss priorities (PLPs) for specific CoS values and aliases.	None.
Incoming Code Point	Specifies the CoS value in bits and the alias of a classifier for incoming packets.	To specify a CoS value and alias, either select preconfigured ones from the list or type new ones. For information about forwarding classes and aliases assigned to well-known DSCPs, see Table 180 on page 572.

Table 186: Classifiers Quick Configuration Page Summary *(continued)*

Field	Function	Your Action
Forwarding Class	Assigns the forwarding class to the specified CoS value and alias.	To assign a forwarding class, select either one of following default forwarding classes or one that you have configured: ■ **best-effort—**Provides no special CoS handling of packets. Typically, RED drop profile is aggressive and no loss priority is defined. ■ **expedited-forwarding—**Provides low loss, low delay, low jitter, assured bandwidth, and end-to-end service. Packets can be forwarded out of sequence or dropped. ■ **assured-forwarding—**Provides high assurance for packets within specified service profile. Excess packets are dropped. ■ **network-control—**Packets can be delayed but not dropped.
Loss Priority	Assigns a loss priority to the specified CoS value and alias.	To assign a loss priority, select one of the following: ■ **low—**Packet has a low loss priority. ■ **high—**Packet has a high loss priority. ■ **medium-low—**Packet has a medium-low loss priority. ■ **medium-high–**Packet has a medium-high loss priority.
Add	Assigns a forwarding class and loss priority to the specified CoS value and alias. A classifier examines the incoming packet's header for the specified CoS value and alias and assigns it the forwarding class and loss priority that you have defined.	To assign a forwarding class and loss priority to a specific CoS value and alias, click **Add**.
Delete	Removes the forwarding class and loss priority assignment from the classifier.	To remove the forwarding class and loss priority assignment, select it and click **Delete**.

Defining Rewrite Rules

Figure 89 on page 588 shows the initial Quick Configuration page for defining rewrite rules, and Table 187 on page 588 describes the related fields. Use the rewrite rules to alter the CoS values in outgoing packets to meet the requirements of the targeted peer. A rewrite rule examines the forwarding class and loss priority of a packet and sets its bits to a corresponding value specified in the rule.

Figure 89: Rewrite Rules Quick Configuration Page

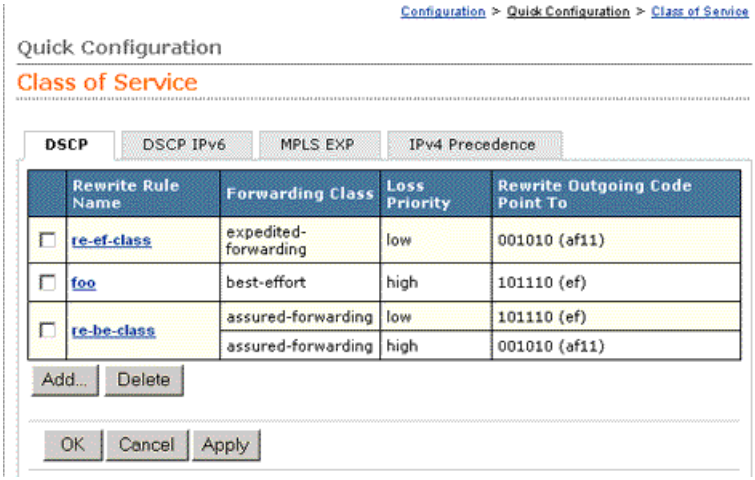


Table 187: Rewrite Rules Quick Configuration Page Summary

Field	Function	Your Action
Rewrite Rules Summary		
DSCP	Allows you to redefine DSCP IPv4 code point values of outgoing packets.	To redefine a DSCP code point value, click **DSCP**.
DSCP IPv6	Allows you to redefine DSCP IPv6 code point values.	To redefine a DSCP IPv6 code point value, click **DSCP IPv6**.
MPLS EXP	Allows you to redefine MPLS experimental (EXP) bits.	To redefine MPLS EXP bits, click **MPLS EXP**.
IPv4 Precedence	Allows you to redefine IPv4 precedence code point values.	To redefine an IPv4 precedence code point value, click **IPv4 Precedence**.
Rewrite Rule Name	Displays names of defined rewrite rules. Allows you to edit a specific rule.	To edit a rule, click its name.
Forwarding Class	Displays forwarding classes associated with a specific rewrite rule.	None.
Loss Priority	Displays loss priority values associated with a specific rewrite rule,	None.
Rewrite Outgoing Code Point To	Displays the CoS values and aliases that a specific rewrite rule has set for a specific forwarding class and loss priority.	None.
Add	Opens a page that allows you to define a new rewrite rule.	To add a rewrite rule, click **Add**.
Delete	Removes specified rewrite rules.	To remove a rule, select the check box next to it and click **Delete**.

Table 187: Rewrite Rules Quick Configuration Page Summary *(continued)*

Field	Function	Your Action
Add a Rewrite Rule/Edit Rewrite Rule		
Rewrite Rule Name	Specifies a rewrite rule name.	To name a rule, type the name—for example, rewrite-dscps.
Code Point Mapping	Rewrites outgoing CoS values of a packet, based on the forwarding class and loss priority. Allows you to remove a Code Point Mapping entry.	To configure the CoS value assignment, follow these steps: 1. From the Forwarding Class list, select a class. 2. Select a priority from the following: ■ **low—**Rewrite rule applies to packets with a low loss priority. ■ **high—**Rewrite rule applies to packets with a high loss priority. ■ **medium-low—**Rewrite rule applies to packets with a medium-low loss priority. ■ **medium-high–**Rewrite rule applies to packets with a medium-high loss priority. 3. For Rewritten Code Point, either select a predefined CoS value and alias or type a new CoS value and alias. For information about predefined CoS values and aliases, see Table 178 on page 568. 4. Click **Add**. To remove a code point mapping entry, select it and click **Delete**.

Defining Schedulers

Figure 90 on page 590 shows the initial Quick Configuration page for defining schedulers, scheduler maps, and random early detection (RED) drop profiles. Using schedulers, you can assign attributes to queues and thereby provide congestion control to a particular class of traffic. These attributes include the amount of interface bandwidth, memory buffer size, transmit rate, RED drop profiles and priority.

To configure schedulers using the Quick Configuration pages:

1. Create a drop profile by specifying the fill levels and drop probabilities. The drop profile map on the Scheduler page uses this drop profile. For a description of RED drop profile-related fields, see Table 188 on page 590.
2. Create a scheduler and specify attributes to it. For a description of scheduler-related fields, see Table 189 on page 592.

3. Associate the scheduler to a forwarding class. Because the forwarding class is assigned to a queue number, the queue inherits this scheduler's attributes. For a description of scheduler map-related fields, see Table 190 on page 594.

Figure 90: Schedulers Quick Configuration Page

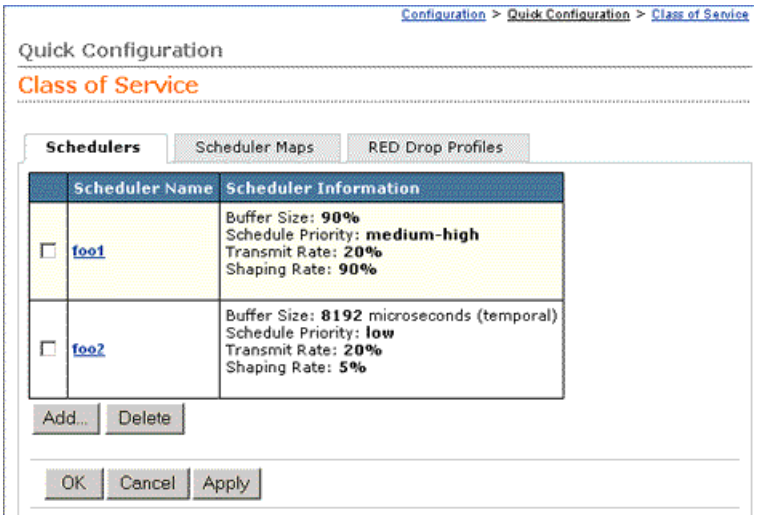


Table 188: RED Drop Profiles Quick Configuration Page Summary

Field	Function	Your Action
RED Drop Profiles Summary		
RED Drop Profile Name	Displays the configured random early detection (RED) drop profile names. RED attempts to avoid congestion by dropping packets from the head of a queue. Allows you edit a specific drop profile.	To edit a RED drop profile, click its name.
Graph RED Profile	Opens a new window and displays a graph for a specific RED drop profile.	To view the graph for a specific RED drop profile, click **Graph**.
RED Drop Profile Information (Fill Level, Drop Probability)	Displays information about the data point type, the queue buffer fill level, and the drop probability for specific RED drop profiles.	None.
Add	Opens a page that allows you to add a RED drop profile.	To add a RED drop profile, click **Add**.
Delete	Removes a RED drop profile.	To remove a RED drop profile, select it and click **Delete**.
Add a RED Drop Profile/Edit RED Drop Profile		

Table 188: RED Drop Profiles Quick Configuration Page Summary *(continued)*

Field	Function	Your Action
Graphed RED Profile	Displays a graph of RED drop profiles. Each data point in this graph is defined by a pair of x and y coordinates and represents the relationship between them. The x axis represents the queue buffer fill level, which is a percentage value of how full the queue is. The y axis represents the drop probability, which is a percentage value of the chances of a packet being dropped.	None.
Drop Profile Name	Specifies a name for a drop profile. A drop profile consists of pairs of values between 0 and 100, one for queue buffer fill level and one for drop probability, that determine the relationship between a buffer's fullness and the likelihood it will drop packets. The values you assign to each pair must increase relative to the previous pair of values. With a few value pairs the system automatically constructs a drop profile.	To name a drop profile, type the name—for example, be-normal.
RED Drop Profile Type	Specifies whether a RED drop profile type is interpolated or segmented. For more information about segmented and interpolated drop profiles, see the *JUNOS Class of Service Configuration Guide*.	To specify a RED drop profile type, select one of the following: ■ **Interpolated**—The value pairs are interpolated to produce a smooth profile. ■ **Segmented**—The value pairs are represented by line fragments, which connect each data point on the graph to produce a segmented profile.
Data Points	Specifies the points for generating the RED drop profile graph. Each data point is defined by a pair of x and y coordinates and represents the relationship between them. The x axis represents the queue buffer fill level, which is a percentage value of how full the queue is. A value of 100 means the queue is full. The y axis represents the drop probability, which is a percentage value of the chances of a packet being dropped. A value of 0 means that a packet is never dropped, and a value of 100 means that all packets are dropped.	To specify x and y coordinates for data points, type a number between 0 and 100 in the following boxes: ■ Fill level—Type the percentage value of queue buffer fullness for the x coordinate—for example, 95. ■ Drop profile—Type the percentage value of drop probability for the y coordinate—for example, 85.
Add	Adds the specified queue buffer fill level and drop probability as a data point for the graph.	To add the specified fill level and drop probability, click **Add**.
Delete	Removes a data point.	To remove a data point, select it and click **Delete**.

Table 189: Schedulers Quick Configuration Page Summary

Field	Function	Your Action
Scheduler Summary		
Scheduler Name	Displays the names of defined schedulers. Allows you to edit a specific scheduler.	To edit a scheduler, click its name.
Scheduler Information	Displays a summary of defined settings for a scheduler, such as bandwidth, delay buffer size, transmit and shaping rates, and RED drop profiles.	None.
Add	Opens a page that allows you to adds a scheduler.	To add a scheduler, click **Add**.
Delete	Removes a scheduler.	To remove a scheduler, select it and click **Delete**.
Add a Scheduler/Edit Scheduler		
Scheduler Name	Specifies the name for a scheduler.	To name a scheduler, type the name—for example, be-scheduler.
Buffer Size	Defines the size of the delay buffer. The delay buffer bandwidth provides packet buffer space to absorb burst traffic up to the specified duration of delay. By default, queues 0 through 7 have the following percentage of the total available buffer space: ■ Queue 0—95 percent ■ Queue 1—0 percent ■ Queue 2—0 percent ■ Queue 3—5 percent ■ Queue 4—0 percent ■ Queue 6—0 percent ■ Queue 7—0 percent **NOTE:** A large buffer size value means a greater possibility for delaying packets in the network. This might not be practical for sensitive traffic such as voice or video.	To define a delay buffer size for a scheduler, select the appropriate option: ■ To specify no buffer size, select **Unconfigured**. ■ To specify buffer size as a percentage of the total buffer, select **Percent** and type an integer from 1 through 100. ■ To specify buffer size as the remaining available buffer, select **Remainder**. ■ To specify buffer size in microseconds, select **Temporal**, and type an integer within the range of the buffer size available to you on your platform—for example, 8192.

Table 189: Schedulers Quick Configuration Page Summary *(continued)*

Field	Function	Your Action
Drop Profile Map	Sets the drop profile for a specific packet loss priority (PLP) and protocol type. By default, the drop profile is assigned to packets with low PLP, regardless of protocol type.	To configure a scheduler drop profile: 1. Select a loss priority from the following: ■ **low**—Drop profile applies to packets with a low loss priority. ■ **medium-low**—Drop profile applies to packets with a medium-low loss priority. ■ **high**—Drop profile applies to packets with a high loss priority. ■ **medium-high**—Drop profile applies to packets with a medium-high loss priority. ■ **any**—Drop profile applies to all packets irrespective of the loss priority. 2. From the Protocol list, select a protocol. 3. From the Drop Profile list, select a profile. 4. Click **Add**. To remove a drop profile entry, select it and click **Delete**.
Scheduling Priority	Sets the transmission priority of the scheduler, which determines the order in which an output interface transmits traffic from the queues. You can set scheduling priority at different levels in an order of increasing priority from low to high. A high-priority queue with a high transmission rate might lock out lower-priority traffic.	To specify a priority, select one of the following: ■ **high**—Packets in this queue are transmitted first. ■ **low**—Packets in this queue are transmitted last. ■ **medium-high**—Packets in this queue are transmitted after high-priority packets. ■ **medium-low**—Packets in this queue are transmitted before low-priority packets.
Shaping Rate	Defines the minimum bandwidth allocated to a queue. The default shaping rate is 100 percent, which is the same as no shaping at all.	To define a shaping rate, select the appropriate option: ■ To specify no shaping rate, select **Unconfigured**. ■ To specify shaping rate as an absolute number of bits per second, select **Absolute Rate** and type an integer from 3200 through 32000000000. ■ To specify shaping rate as a percentage, select **Percent** and type an integer from 0 through 100.

Table 189: Schedulers Quick Configuration Page Summary *(continued)*

Field	Function	Your Action
Transmit Rate	Defines the transmission rate of a scheduler. The transmit rate determines the traffic bandwidth from each forwarding class you configure. By default, queues 0 through 7 have the following percentage of transmission capacity: ■ Queue 0—95 percent ■ Queue 1—0 percent ■ Queue 2—0 percent ■ Queue 3—5 percent ■ Queue 4—0 percent ■ Queue 6—0 percent ■ Queue 7—0 percent	To define a transmit rate, select the appropriate option: ■ To not specify transmit rate, select **Unconfigured**. ■ To specify the remaining transmission capacity, select **Remainder Available**. ■ To specify a percentage of transmission capacity, select **Percent** and type an integer from 1 through 100. To enforce the exact transmission rate or percentage you configured, select the **Exact Transmit Rate** check box.

Table 190: Scheduler Maps Quick Configuration Page Summary

Field	Function	Your Action
Scheduler Maps Summary		
Scheduler Map Name	Displays the names of defined scheduler maps. Scheduler maps link schedulers to forwarding classes. Allows you to edit a scheduler map.	To edit a scheduler map, click its name.
Scheduler Map Information	For each map, displays the schedulers and the forwarding classes that they are assigned to.	None.
Add	Opens a page that allows you to add a scheduler map.	To add a scheduler map, click **Add**.
Delete	Removes a scheduler map.	To remove a scheduler map, select it and click **Delete**.
Add a Scheduler Map/Edit Scheduler Map		
Scheduler Map Name	Specifies the name for a scheduler map.	To name a map, type the name—for example, be-scheduler-map.
Scheduler Mapping	Allows you to associate a preconfigured scheduler with a forwarding class. Once applied to an interface, the scheduler maps affect the hardware queues, packet schedulers, and RED drop profiles.	To associate a scheduler with a forwarding class, locate the forwarding class and select the scheduler in the box next to it.

Defining Virtual Channel Groups

NOTE: SRX-series devices do not support Virtual Channels.

Figure 91 on page 595 shows the initial Quick Configuration page for defining virtual channel groups, and Table 191 on page 595 describes the related fields. Use virtual channels to avoid oversubscription of links by limiting traffic from a higher aggregated bandwidth to a lower one—for example, to limit traffic from a main office to branch offices. You channelize this traffic by applying queuing, packet scheduling, and accounting rules to logical interfaces.

Figure 91: Virtual Channel Group Quick Configuration Page

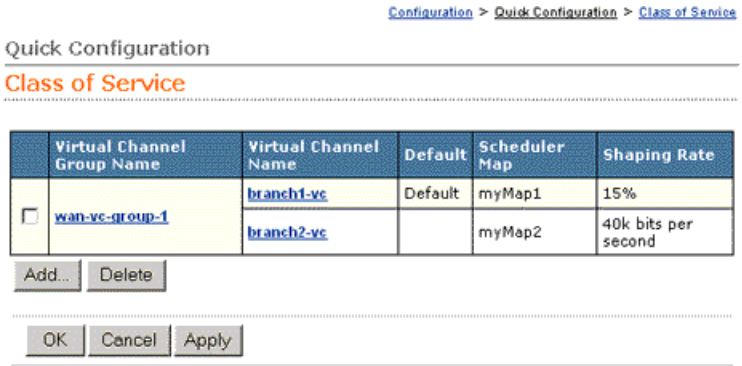


Table 191: Virtual Channel Group Quick Configuration Page Summary

Field	Function	Your Action
Virtual Channel Groups Summary		
Virtual Channel Group Name	Displays names of defined virtual channel groups. Allows you to edit a virtual channel group.	To edit a virtual channel group, click its name.
Virtual Channel Name	Displays names of defined virtual channels. Allows you to edit a virtual channel.	To edit a virtual channel, click its name.
Default	Marks the default virtual channel of a group. One of the virtual channels in a group must be configured as the default channel. Any traffic not explicitly directed to a particular channel is transmitted by this channel.	None.
Scheduler Map	Displays the scheduler map assigned to a particular virtual channel.	None.
Shaping Rate	Displays the shaping rate configured for a virtual channel.	None.

Table 191: Virtual Channel Group Quick Configuration Page Summary *(continued)*

Field	Function	Your Action
Add	Opens a page that allows you to add a virtual channel group.	To add a virtual channel group, click **Add**.
Delete	Removes a specific virtual channel group.	To remove a specific virtual channel group, locate its name, select the check box next to it, and click **Delete**.
Add a Virtual Channel Group/Edit a Virtual Channel Group		
Virtual Channel Group Name	Specifies a name for a virtual channel group.	To name a group, type the name—for example, wan-vc-group.
Add	Creates a virtual channel group. Opens a page that allows you to add a virtual channel to the specified group.	To create a virtual channel group, click **Add**.
Add a Virtual Channel/Edit Virtual Channel		
Virtual Channel Name	Specifies the name of a virtual channel to be assigned to a virtual channel group.	To name a virtual channel, either select a predefined name from the list or type a new name—for example, branch1–vc.
Scheduler Map	Specifies a predefined scheduler map to assign to a virtual channel. Scheduler maps associate schedulers with forwarding classes. For information about how to define scheduler maps, see Table 190 on page 594.	To specify a scheduler map, select it from the Scheduler Map list.
Shaping Rate	Specifies the shaping rate for a virtual channel. The shaper limits the maximum bandwidth transmitted by a virtual channel. Configuring a shaping rate is optional. If no shaping rate is configured, a virtual channel without a shaper can use the full logical interface bandwidth.	To specify a shaping rate, select one of the following options: ■ To specify no shaping rate, select **Unconfigured**. ■ To configure a shaping rate as an absolute number of bits per second, select **Absolute Rate** and type a value between 3200 and 320000000000. ■ To configure a shaping rate as a percentage, select **Percent** and type a value between 0 and 100.

Assigning CoS Components to Interfaces

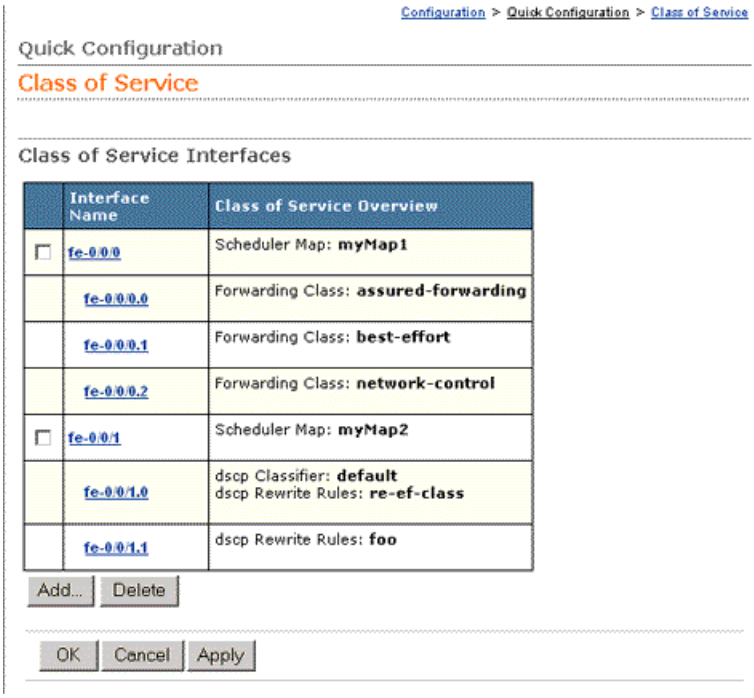
NOTE: SRX-series devices do not support WAN interfaces (including T1/E1 and channelized T1/E1.

After you have defined CoS components, you must assign them to logical or physical interfaces. The CoS Quick Configuration pages allow you to assign scheduler maps

to physical or logical interfaces and to assign forwarding classes, classifiers, rewrite rules, or virtual channel groups to logical interfaces.

Figure 92 on page 597 shows the initial Quick Configuration page for assigning CoS components to interfaces. The page displays the interfaces available for CoS component assignment and the status of existing CoS components.

Figure 92: Assignment of CoS Components to Interfaces Quick Configuration Page



To assign CoS components to interfaces with Quick Configuration:

1. In the J-Web interface, select **Configuration > Quick Configuration > Class of Service > Assign Class of Service Components to Interfaces**.
2. Enter information into these Quick Configuration pages, as described in Table 192 on page 598.
3. Click one of the following buttons after completing configuration on any Quick Configuration main page:
   - To apply the configuration and stay in current the Quick Configuration page, click **Apply**.
   - To apply the configuration and return to the previous Quick Configuration page, click **OK**.
   - To cancel your entries and return to the previous Quick Configuration page, click **Cancel**.
4. To verify the CoS configuration, see "Verifying a CoS Configuration" on page 682.

Table 192: Assigning CoS Components to Interfaces Quick Configuration Summary

Field	Function	Your Action
Class of Service Interfaces		
Interface Name (See the interface naming conventions in "Network Interface Naming" on page 16.)	Lists the names of physical and logical interfaces configured on the system. Allows you to edit CoS component assignments to physical and logical interfaces.	To edit an interface's CoS assignments, click the interface.
Class of Service Overview	Displays the CoS components assigned to a particular interface—for example, information about DSCP classifiers, EXP classifiers, or DSCP rewrite rules.	None.
Add	Allows you to add a CoS service to a physical interface.	To add a CoS service to a physical interface, click **Add**.
Delete	Removes CoS services assigned to a specific interface.	To remove CoS services assigned to a specific interface, locate the interface name, click the check box next to it, and click **Delete**.
Add CoS Service to a Physical Interface/Edit CoS Physical Interface		
Physical Interface Name	Specifies the name of a physical interface. Allows you to assign CoS components to a set of interfaces at the same time.	To specify an interface for CoS assignment, type its name in the Physical Interface Name box. To specify a set of interfaces for CoS assignment, use the wildcard character (*)—for example, ge-0/*/0.
Scheduler Map	Specifies a predefined scheduler map for the physical interface. A scheduler map enables the physical interface to have more than one set of output queues. **NOTE:** For 4-port Fast Ethernet ePIMs, if you apply a CoS scheduler map on outgoing (egress) traffic, the device does not divide the bandwidth appropriately among the CoS queues. As a workaround, configure enforced CoS shaping on the ports.	To specify a map for an interface, select it from the Scheduler Map list.
Add	Allows you to add a CoS service to a logical interface on a specified physical interface.	To add a CoS Service to a logical interface, click **Add**.
Add CoS Service to a Logical Interface Unit/Edit CoS Logical Interface Unit		
Logical Interface Unit Name	Specifies the name of a logical interface. Allows you to assign CoS components to all logical interfaces configured on a physical interface at the same time.	To specify an interface for CoS assignment, type its name in the Logical Interface Unit Name box. To assign CoS services to all logical interfaces configured on this physical interface, type the wildcard character (*).

Table 192: Assigning CoS Components to Interfaces Quick Configuration Summary *(continued)*

Field	Function	Your Action
Scheduler Map	Specifies a predefined scheduler map for this interface. **NOTE:** You can configure either a scheduler map or a virtual channel group on a logical interface, not both.	To assign a scheduler map to the interface, select it from the list.
Forwarding Class	Assigns a predefined forwarding class to incoming packets on a logical interface.	To assign a forwarding class to the interface, select it.
Virtual Channel Group	Applies a virtual channel group to a logical interface. Applying a virtual channel group creates a set of eight queues for each virtual channel in the group. **NOTE:** You can configure either a scheduler map or a virtual channel group on a logical interface, not both.	To specify a virtual channel group for the interface, select it from the list.
Classifiers	Allows you to apply classification maps to a logical interface. Classifiers assign a forwarding class and loss priority to an incoming packet based on its CoS value.	To assign a classification map to the interface, select an appropriate classifier for each CoS value type used on the interface.
Rewrite Rules	Allows you to apply rewrite rule configurations to a logical interface. Rewrite rules rewrite the CoS values in an outgoing packet based on forwarding class and loss priority. You can choose to apply your own rewrite rule or a default one. The default rewrite assignments are based on the default bit definitions of DSCP, DSCP IPv6, MPLS EXP, and IP precedence.	To apply a rewrite rule configuration to the interface, select a rule for each CoS value type used on the interface.

Configuring CoS Components with a Configuration Editor

To configure the device as a node in a network supporting CoS, read the section "Before You Begin" on page 579, determine your needs, and select the tasks you need to perform from the following list. For information about using the J-Web and CLI configuration editors, see the the *J-Web Interface User Guide* and the *JUNOS CLI User Guide*.

- Configuring a Policer for a Firewall Filter on page 600
- Configuring and Applying a Firewall Filter for a Multifield Classifier on page 601
- Assigning Forwarding Classes to Output Queues on page 604

Configuring a Policer for a Firewall Filter

You configure a policer to detect packets that exceed the limits established for expedited forwarding. The packets that exceed these limits are given a higher loss priority than packets within the bandwidth and burst size limits.

The following example shows how to configure a policer called ef-policer that identifies for likely discard expedited forwarding packets with a burst size greater than 2000 bytes and a bandwidth greater than 10 percent.

For more information about firewall filters, see "Configuring Stateless Firewall Filters (ACLs)" on page 521 and the *JUNOS Policy Framework Configuration Guide*.

To configure an expedited forwarding policer for a firewall filter for the Services Router:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 193 on page 600.
3. Go on to "Configuring and Applying a Firewall Filter for a Multifield Classifier" on page 601.

Table 193: Configuring a Policer for a Firewall Filter

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Firewall** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Firewall, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit firewall
Create the policer for expedited forwarding, and give the policer a name—for example, ef-policer.	1. Click **Add new entry** next to Policer. 2. In the Policer name box, type ef-policer.	Enter edit policer ef-policer

Table 193: Configuring a Policer for a Firewall Filter *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Set the burst limit for the policer—for example, 2k. Set the bandwidth limit or percentage for the bandwidth allowed for this type of traffic—for example, use a bandwidth percent of 10.	1. Click **Configure** next to If exceeding. 2. In the Burst size limit box, type a limit for the burst size allowed—for example, 2k. 3. From the Bandwidth list, select **bandwidth-percent**. 4. In the Bandwidth percent box, type 10. 5. Click **OK**.	Enter set if-exceeding burst-limit-size 2k set if-exceeding bandwidth-percent 10
Enter the loss priority for packets exceeding the limits established by the policer—for example, high.	1. Click **Configure** next to Then. 2. From the Loss priority list, select **high**. 3. Click **OK**.	Enter set then loss-priority high

Configuring and Applying a Firewall Filter for a Multifield Classifier

You configure a multifield (MF) classifier to detect packets of interest to CoS and assign the packet to the proper forwarding class independently of the DiffServ code point (DSCP). To configure a multifield classifier on a customer-facing or host-facing link, configure a firewall filter to classify traffic. Packets are classified as they arrive on an interface.

One common way to detect packets of CoS interest is by source or destination address. The destination address is used in this example, but many other matching criteria for packet detection are available to firewall filters.

This example shows how to configure the firewall filter mf-classifier and apply it to the Services Router's Gigabit Ethernet interface ge-0/0/0. The firewall filter consists of the rules (terms) listed in Table 194 on page 601.

Table 194: Sample mf-classifier Firewall Filter Terms

Rule (Term)	Purpose	Contents
assured forwarding	Detects packets destined for 192.168.44.55, assigns them to an assured forwarding class, and gives them a low likelihood of being dropped.	Match condition: destination address 192.168.44.55 Forwarding class: af-class Loss priority: low
expedited-forwarding	Detects packets destined for 192.168.66.77, assigns them to an expedited forwarding class, and subjects them to the EF policer configured in "Configuring a Policer for a Firewall Filter" on page 600.	Match condition: destination address 192.168.66.77 Forwarding class: ef-class Policer: ef-policer

Table 194: Sample mf-classifier Firewall Filter Terms *(continued)*

Rule (Term)	Purpose	Contents
network control	Detects packets with a network control precedence and forwards them to the network control class.	Match condition: precedence net-control Forwarding class: nc-class
best-effort-data	Detects all other packets and assigns them to the best effort class.	Forwarding class: be-class

For more information about firewalls filters see "Configuring Stateless Firewall Filters (ACLs)" on page 521 and the *JUNOS Policy Framework Configuration Guide*.

To configure a firewall filter for a multifield classifier for the Services Router:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 195 on page 602.
3. Go on to "Assigning Forwarding Classes to Output Queues" on page 604.

Table 195: Configuring and Applying a Firewall Filter for a Multifield Classifier

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Firewall** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Firewall, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit firewall
Create the multifield classifier filter and name it—for example, mf-classifier.	1. Click **Add new entry** next to Filter. 2. In the Filter name box, type mf-classifier. 3. Select the check box next to Interface specific.	Enter edit filter mf-classifier set interface-specific
Create the term for the assured forwarding traffic class, and give it a name—for example, assured-forwarding.	1. Click **Add new entry** next to Term. 2. In the Rule name box, type assured-forwarding.	Enter edit term assured-forwarding
Create the match condition for the assured forwarding traffic class. Use the destination address for assured forwarding traffic—for example, 192.168.44.55.	1. Click **Configure** next to From. 2. Click **Add new entry** next to Destination address. 3. In the Address box, type 192.168.44.55. 4. Click **OK** twice.	Enter set from destination-address 192.168.44.55

Table 195: Configuring and Applying a Firewall Filter for a Multifield Classifier *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Create the forwarding class for assured forwarding DiffServ traffic—for example, af-class. Set the loss priority for the assured forwarding traffic class—for example, low.	1. Click **Configure** next to Then. 2. In the Forwarding class box, type af-class. 3. From the Loss priority list, select **low**. 4. Click **OK** twice.	Enter set then forwarding-class af-class set then loss-priority low
Create the term for the expedited forwarding traffic class, and give it a name—for example, expedited-forwarding.	1. Click **Add new entry** next to Term. 2. In the Rule name box, type expedited-forwarding.	From the [edit firewall filter mf-classifer] hierarchy level, enter edit term expedited-forwarding
Create the match condition for the expedited forwarding traffic class. Use the destination address for expedited forwarding traffic—for example, 192.168.66.77.	1. Click **Configure** next to From. 2. Click **Add new entry** next to Destination address. 3. In the Address box, type 192.168.66.77. 4. Click **OK** twice.	Enter set from destination-address 192.168.66.77
Create the forwarding class for expedited forwarding DiffServ traffic—for example, ef-class. Apply the policer for the expedited forwarding traffic class. Use the EF policer previously configured for expedited forwarding DiffServ traffic—ef-policer. (See "Configuring a Policer for a Firewall Filter" on page 600.)	1. Click **Configure** next to Then. 2. In the Forwarding class box, type ef-class. 3. From the Policer choice list, select **Policer**. 4. In the Policer box, type ef-policer. 5. Click **OK** twice.	Enter set then forwarding-class ef-class set then policer ef-policer
Create the term for the network control traffic class, and give it a name—for example, network-control.	1. Click **Add new entry** next to Term. 2. In the Rule name box, type network-control.	From the [edit firewall filter mf-classifer] hierarchy level, enter edit term network-control
Create the match condition for the network control traffic class.	1. Click **Configure** next to From. 2. From the Precedence choice list, select **Precedence**. 3. Click **Add new entry** next to Precedence. 4. From the Value keyword list, select **net-control**. 5. Click **OK** twice.	Enter set from precedence net-control

Table 195: Configuring and Applying a Firewall Filter for a Multifield Classifier *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Create the forwarding class for the network control traffic class, and give it a name—for example, nc-class.	1. Click **Configure** next to Then. 2. In the Forwarding class box, type nc-class. 3. Click **OK** twice.	Enter set then forwarding-class nc-class
Create the term for the best-effort traffic class, and give it a name—for example, best-effort-data.	1. Click **Add new entry** next to Term. 2. In the Rule name box, type best-effort-data.	From the [edit firewall filter mf-classifer] hierarchy level, enter edit term best-effort-data
Create the forwarding class for the best-effort traffic class, and give it a name—for example, be-class. (Because this is the last term in the filter, it has no match condition.)	1. Click **Configure** next to Then. 2. In the Forwarding class box, type be-class. 3. Click **OK** four times.	Enter set then forwarding-class be-class
Navigate to the **Interfaces** level in the configuration hierarchy.	On the main Configuration page next to Interfaces, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit interfaces
Apply the multifield classifier firewall filter mf-classifier as an input filter on each customer-facing or host-facing interface that needs the filter—for example, on ge-0/0/0, unit 0.	1. Click the Interface **ge-0/0/0** and Unit **0**. 2. Click **Configure** next to Inet. 3. Click **Configure** next to Filter. 4. From the Input choice list, select **Input**. 5. In the Input box, type mf-classifier. 6. Click **OK**.	Enter set ge-0/0/0 unit 0 family inet filter input mf-classifier

Assigning Forwarding Classes to Output Queues

You must assign the forwarding classes established by the mf-classifier multifield classifier to output queues. This example assigns output queues as shown in Table 196 on page 604.

Table 196: Sample Output Queue Assignments for mf-classifier Forwarding Queues

mf-classifier Forwarding Class	For Traffic Type	Output Queue
be-class	Best-effort traffic	Queue 0
ef-class	Expedited forwarding traffic	Queue 1
af-class	Assured forwarding traffic	Queue 2
nc-class	Network control traffic	Queue 3

For multifield classifier details, see "Configuring and Applying a Firewall Filter for a Multifield Classifier" on page 601.

To assign forwarding classes to output queues:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 197 on page 605.
3. Go on to "Configuring and Applying Rewrite Rules" on page 611.

Table 197: Assigning Forwarding Classes to Output Queues

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Class of service** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit class-of-service
Assign best-effort traffic to queue 0.	1. Click **Configure** next to Forwarding classes. 2. Click **Add new entry** next to Queue. 3. In the Queue num box, type 0. 4. In the Class name box, type the previously configured name of the best-effort class—be-class. 5. Click **OK**.	Enter set forwarding-classes queue 0 be-class
Assign expedited forwarding traffic to queue 1.	1. Click **Add new entry** next to Queue. 2. In the Queue num box, type 1. 3. In the Class name box, type the previously configured name of the expedited forwarding class—ef-class. 4. Click **OK**.	Enter set forwarding-classes queue 1 ef-class
Assign assured forwarding traffic to queue 2.	1. Click **Add new entry** next to Queue. 2. In the Queue num box, type 2. 3. In the Class name box, type the previously configured name of the assured forwarding class—af-class. 4. Click **OK**.	Enter set forwarding-classes queue 2 af-class
Assign network control traffic to queue 3.	1. Click **Add new entry** next to Queue. 2. In the Queue num box, type 3. 3. In the Class name box, type the previously configured name of the network control forwarding class—nc-class. 4. Click **OK**.	Enter set forwarding-classes queue 3 nc-class

Configuring Forwarding Classes

To configure CoS forwarding classes on an SRX-series device, include the following statements at the [edit class-of-service] hierarchy level of the configuration:

```
[edit class-of-service]
forwarding-classes {
    class class-name queue-num queue-number priority (high | low);
    queue queue-number class-name priority (high | low);
}
interfaces {
    interface-name {
        unit logical-unit-number {
            forwarding-class class-name;
        }
    }
}
restricted-queues {
    forwarding-class class-name queue-number;
}
```

You cannot commit a configuration that assigns the same forwarding class to two different queues.

Assigning a Forwarding Class to an Interface

On an SRX-series device, you can configure *fixed classification* on a logical interface by specifying a forwarding class to be applied to all packets received by the logical interface, regardless of the packet contents.

To assign a forwarding class configuration to the input logical interface, include the forwarding-class statement at the [edit class-of-service interfaces *interface-name* unit *logical-unit-number*] hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
forwarding-class class-name;
```

You can include interface wildcards for *interface-name* and *logical-unit-number*.

In the following example, all packets coming into the device from the ge-3/0/0.0 interface are assigned to the assured-forwarding forwarding class:

```
[edit class-of-service]
interfaces {
    ge-3/0/0 {
        unit 0 {
            forwarding-class assured-forwarding;
        }
    }
}
```

Example: Configuring Up to Eight Forwarding Classes

By default on all platforms, four output queues are mapped to four forwarding classes as shown in Table 179 on page 570. On SRX-series devices, you can configure up to eight forwarding classes and eight queues once the eight-queue mode has been enabled. For more information on enabling up to eight queues on SRX-series devices, see "Forwarding Classes" on page 560

NOTE: The new setting takes place only after the FPC is restarted.

To configure up to eight forwarding classes, include the queue statement at the [edit class-of-service forwarding-classes] hierarchy level:

```
[edit class-of-service forwarding-classes]
queue queue-number class-name;
```

The output queue number can be from 0 through 7, and you must map the forwarding classes one-to-one with the output queues. The default scheduler transmission rate and buffer size percentages for queues 0 through 7 are 95, 0, 0, 5, 0, 0, 0, and 0 percent.

For example, to configure a one-to-one mapping between eight forwarding classes and eight queues: you would use the following configuration:

```
[edit class-of-service]
forwarding-classes {
   queue 0 be;
   queue 1 ef;
   queue 2 af;
   queue 3 nc;
   queue 4 ef1;
   queue 5 ef2;
   queue 6 af1;
   queue 7 nc1;
}
```

Defining Eight Classifiers

```
[edit class-of-service]
classifiers {
   dscp dscp-table {
      forwarding-class ef {
         loss-priority low code-points [101000, 101001];
         loss-priority high code-points [101010, 101011];
      }
      forwarding-class af {
         loss-priority low code-points [010000, 010001];
         loss-priority high code-points [010010, 010011];
      }
      forwarding-class be {
         loss-priority low code-points [000000];
      }
      forwarding-class nc {
         loss-priority low code-points [111000];
      }
```

```
        forwarding-class ef1 {
            loss-priority low code-points [101100, 101101];
            loss-priority high code-points [101110];
        }
        forwarding-class af1 {
            loss-priority high code-points [101110];
        }
        forwarding-class ef2 {
            loss-priority low code-points [101111];
        }
        forwarding-class af2 {
            loss-priority low code-points [010000];
        }
        forwarding-class nc1 {
            loss-priority low code-points [111001];
        }
    }
}
```

Adding Eight Schedulers to a Scheduler Map

Configure a custom scheduler map that applies globally to all interfaces, except those that are restricted to four queues:

```
[edit class-of-service]
scheduler-maps {
    sched {
        forwarding-class be scheduler Q0;
        forwarding-class ef scheduler Q1;
        forwarding-class af scheduler Q2;
        forwarding-class nc scheduler Q3;
        forwarding-class ef1 scheduler Q4;
        forwarding-class ef2 scheduler Q5;
        forwarding-class af1 scheduler Q6;
        forwarding-class nc1 scheduler Q7;
    }
}
schedulers {
    Q0 {
        transmit-rate percent 25;
        buffer-size percent 25;
        priority low;
        drop-profile-map loss-priority any protocol both drop-default;
    }
    Q1 {
        buffer-size temporal 2000;
        priority strict-high;
        drop-profile-map loss-priority any protocol both drop-ef;
    }
    Q2 {
        transmit-rate percent 35;
        buffer-size percent 35;
        priority low;
        drop-profile-map loss-priority any protocol both drop-default;
    }
    Q3 {
        transmit-rate percent 5;
        buffer-size percent 5;
```

```
    drop-profile-map loss-priority any protocol both drop-default;
  }
  Q4 {
    transmit-rate percent 5;
    priority high;
    drop-profile-map loss-priority any protocol both drop-ef;
  }
  Q5 {
    transmit-rate percent 10;
    priority high;
    drop-profile-map loss-priority any protocol both drop-ef;
  }
  Q6 {
    transmit-rate remainder;
    priority low;
    drop-profile-map loss-priority any protocol both drop-default;
  }
  Q7 {
    transmit-rate percent 5;
    priority high;
    drop-profile-map loss-priority any protocol both drop-default;
  }
}
```

Configuring an IP Precedence Classifier and Rewrite Tables

```
[edit class-of-service]
classifiers {
  inet-precedence inet-classifier {
    forwarding-class be {
      loss-priority low code-points 000;
    }
    forwarding-class af11 {
      loss-priority high code-points 001;
    }
    forwarding-class ef {
      loss-priority low code-points 010;
    }
    forwarding-class nc1 {
      loss-priority high code-points 011;
    }
    forwarding-class {
      loss-priority low code-points 100;
    }
    forwarding-class af12 {
      loss-priority high code-points 101;
    }
    forwarding-class ef1 {
      loss-priority low code-points 110;
    }
    forwarding-class nc2 {
      loss-priority high code-points 111;
    }
  }
}
exp exp-rw-table {
  forwarding-class be {
```

```
        loss-priority low code-point 000;
    }
    forwarding-class af11 {
        loss-priority high code-point 001;
    }
    forwarding-class ef {
        loss-priority low code-point 010;
    }
    forwarding-class nc1 {
        loss-priority high code-point 111;
    }
    forwarding-class be1 {
        loss-priority low code-point 100;
    }
    forwarding-class af12 {
        loss-priority high code-point 101;
    }
    forwarding-class ef1 {
        loss-priority low code-point 110;
    }
    forwarding-class nc2 {
        loss-priority low code-point 111;
    }
}
inet-precedence inet-rw-table {
    forwarding-class be {
        loss-priority low code-point 000;
    }
    forwarding-class af11 {
        loss-priority high code-point 001;
    }
    forwarding-class ef1 {
        loss-priority low code-point 010;
    }
    forwarding-class nc1 {
        loss-priority low code-point 111;
    }
    forwarding-class be1 {
        loss-priority low code-point 100;
    }
    forwarding-class af12 {
        loss-priority high code-point 101;
    }
    forwarding-class ef1 {
        loss-priority low code-point 111;
    }
    forwarding-class nc2 {
        loss-priority low code-point 110;
    }
}
```

Configuring and Applying Rewrite Rules

You can configure rewrite rules to replace DiffServ code points (DSCPs) on packets received from the customer or host with the values expected by other devices. You do not have to configure rewrite rules if the received packets already contain valid DSCPs. Rewrite rules apply the forwarding class information and packet loss priority used internally by the device to establish the DSCP on outbound packets. Once configured, you must apply the rewrite rules to the correct interfaces.

The following example shows how to create the rewrite rules rewrite-dscps and apply them to the device's Gibabit Ethernet interface ge-0/0/0. The rewrite rules replace the DSCPs on packets in the four forwarding classes, as shown in Table 198 on page 611.

Table 198: Sample rewrite-dscps Rewrite Rules to Replace DSCPs

mf-classifier Forwarding Class	For CoS Traffic Type	rewrite-dscps Rewrite Rules
be-class	Best-effort traffic	Low-priority code point: 000000 High-priority code point: 000001
ef-class	Expedited forwarding traffic	Low-priority code point: 101110 High-priority code point: 101111
af-class	Assured forwarding traffic	Low-priority code point: 001010 High-priority code point: 001100
nc-class	Network control traffic	Low-priority code point: 110000 High-priority code point: 110001

To configure and apply rewrite rules for the device:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 199 on page 611.
3. Go on to "Configuring and Applying Behavior Aggregate Classifiers" on page 614.

Table 199: Configuring and Applying Rewrite Rules

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Class of service** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit class-of-service

Table 199: Configuring and Applying Rewrite Rules *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure rewrite rules for DiffServ CoS.	1. Click **Configure** next to Rewrite rules. 2. Click **Add new entry** next to Dscp. 3. In the Name box, type the name of the rewrite rules—for example, rewrite-dscps.	Enter edit rewrite-rules dscp rewrite-dscps
Configure best-effort forwarding class rewrite rules.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured best-effort forwarding class—be-class. 3. Click **Add new entry** next to Loss priority. 4. From the Loss val list, select **low**. 5. In the Code point box, type the value of the low-priority code point for best-effort traffic—for example, 000000. 6. Click **OK**. 7. Click **Add new entry** next to Loss priority. 8. From the Loss val list, select **high**. 9. In the Code point box, type the value of the high-priority code point for best-effort traffic—for example, 000001. 10. Click **OK** twice.	Enter set forwarding-class be-class loss-priority low code-point 000000 set forwarding-class be-class loss-priority high code-point 000001

Table 199: Configuring and Applying Rewrite Rules *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure expedited forwarding class rewrite rules.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured expedited forwarding class—ef-class. 3. Click **Add new entry** next to Loss priority. 4. From the Loss val list, select **low**. 5. In the Code point box, type the value of the low-priority code point for expedited forwarding traffic—for example, 101110. 6. Click **OK**. 7. Click **Add new entry** next to Loss priority. 8. From the Loss val list, select **high**. 9. In the Code point box, type the value of the high-priority code point for expedited forwarding traffic—for example, 101111. 10. Click **OK** twice.	Enter set forwarding-class ef-class loss-priority low code-point 101110 set forwarding-class ef-class loss-priority high code-point 101111
Configure assured forwarding class rewrite rules.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured assured forwarding class—af-class. 3. Click **Add new entry** next to Loss priority. 4. From the Loss val list, select **low**. 5. In the Code point box, type the value of the low-priority code point for assured forwarding traffic—for example, 001010. 6. Click **OK**. 7. Click **Add new entry** next to Loss priority. 8. From the Loss val list, select **high**. 9. In the Code point box, type the value of the high-priority code point for assured forwarding traffic—for example, 001100. 10. Click **OK** twice.	Enter set forwarding-class af-class loss-priority low code-point 001010 set forwarding-class af-class loss-priority high code-point 001100

Table 199: Configuring and Applying Rewrite Rules *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure network control class rewrite rules.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured network control forwarding class—nc-class. 3. Click **Add new entry** next to Loss priority. 4. From the Loss val list, select **low**. 5. In the Code point box, type the value of the low-priority code point for network control traffic—for example, 110000. 6. Click **OK**. 7. Click **Add new entry** next to Loss priority. 8. From the Loss val list, select **high**. 9. In the Code point box, type the value of the high-priority code point for network control traffic—for example, 110001. 10. Click **OK** four times.	Enter set forwarding-class nc-class loss-priority low code-point 110000 set forwarding-class nc-class loss-priority high code-point 110001
Apply rewrite rules to an interface. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. Click **Add new entry** next to Interfaces 2. In the Interface name box, type the name of the interface—for example, ge-0/0/0. 3. Click **Add new entry** next to Unit. 4. In the Unit number box, type the logical interface unit number—0. 5. Click **Configure** next to Rewrite rules. 6. In the Rewrite rules name box, under Dscp, type the name of the previously configured rewrite rules—rewrite-dscps. 7. Click **OK**.	From the [edit class of service] hierarchy level, enter set interfaces ge-0/0/0 unit 0 rewrite-rules dscp rewrite-dscps

Configuring and Applying Behavior Aggregate Classifiers

You configure behavior aggregate classifiers to classify packets that contain valid DSCPs to appropriate queues. Once configured, you must apply the behavior aggregate classifier to the correct interfaces.

The following example shows how to configure the DSCP behavior aggregate classifier ba-classifier as the default DSCP map, and apply it to the device's Gigabit Ethernet interface ge-0/0/0. The behavior aggregate classifier assigns loss priorities, as shown in Table 200 on page 615, to incoming packets in the four forwarding classes.

Table 200: Sample ba-classifier Loss Priority Assignments

mf-classifier Forwarding Class	For CoS Traffic Type	ba-classifier Assignments
be-class	Best-effort traffic	High-priority code point: 000001
ef-class	Expedited forwarding traffic	High-priority code point: 101111
af-class	Assured forwarding traffic	High-priority code point: 001100
nc-class	Network control traffic	High-priority code point: 110001

To configure and apply behavior aggregate classifiers for the device:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 201 on page 615.
3. Go on to "Configuring RED Drop Profiles for Congestion Control" on page 620.

Table 201: Configuring and Applying Behavior Aggregate Classifiers

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Class of service** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit class-of-service
Configure behavior aggregate classifiers for DiffServ CoS.	1. Click **Configure** next to Classifiers. 2. Click **Add new entry** next to Dscp. 3. In the Name box, type the name of the behavior aggregate classifier—for example, ba-classifier. 4. In the Import box, type the name of the default DSCP map, default.	Enter edit classifiers dscp ba-classifier set import default

Table 201: Configuring and Applying Behavior Aggregate Classifiers *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure a best-effort forwarding class classifier.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured best-effort forwarding class—be-class. 3. Click **Add new entry** next to Loss priority. 4. From the Loss val list, select **high**. 5. Click **Add new entry** next to Code points. 6. In the Value box, type the value of the high-priority code point for best-effort traffic—for example, 00001. 7. Click **OK** three times.	Enter set forwarding-class be-class loss-priority high code-points 000001
Configure an expedited forwarding class classifier.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured expedited forwarding class—ef-class. 3. Click **Add new entry** next to Loss priority. 4. From the Loss val list, select **high**. 5. Click **Add new entry** next to Code points. 6. In the Value box, type the value of the high-priority code point for expedited forwarding traffic—for example, 101111. 7. Click **OK** three times.	Enter set forwarding-class ef-class loss-priority high code-points 101111

Table 201: Configuring and Applying Behavior Aggregate Classifiers *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure an assured forwarding class classifier.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured assured forwarding class—af-class. 3. Click **Add new entry** next to Loss priority. 4. From the Loss val list, select **high**. 5. Click **Add new entry** next to Code points. 6. In the Value box, type the value of the high-priority code point for assured forwarding traffic—for example, 001100. 7. Click **OK** three times.	Enter set forwarding-class af-class loss-priority high code-points 001100
Configure a network control class classifier.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured network control forwarding class—nc-class. 3. Click **Add new entry** next to Loss priority. 4. From the Loss val list, select **high**. 5. Click **Add new entry** next to Code points. 6. In the Value box, type the value of the high-priority code point for network control traffic—for example, 110001. 7. Click **OK** five times.	Enter set forwarding-class nc-class loss-priority high code-points 110001

Table 201: Configuring and Applying Behavior Aggregate Classifiers *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Apply the behavior aggregate classifier to an interface. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. Click **Add new entry** next to Interfaces. 2. In the Interface name box, type the name of the interface—for example, ge-0/0/0. 3. Click **Add new entry** next to Unit. 4. In the Unit number box, type the logical interface unit number—0. 5. Click **Configure** next to Classifiers. 6. In the Classifiers box, under Dscp, type the name of the previously configured behavior aggregate classifier—ba-classifier. 7. Click **OK**.	From the [edit class of service] hierarchy level, enter set interfaces ge-0/0/0 unit 0 classifiers dscp ba-classifier

Example: Defining Aliases for Bits

When you configure classes and define classifiers, you can refer to the markers by alias names. You can configure user-defined classifiers in terms of alias names. If the value of an alias changes, it alters the behavior of any classifier that references it.

To define a code-point alias on an SRX-series device, include the code-point-aliases statement at the [edit class-of-service] hierarchy level:

```
[edit class-of-service]
code-point-aliases {
   (dscp |exp | ieee-802.1 | inet-precedence) {
   alias-name bits;
   }
}
```

The CoS marker types are as follows:

- dscp—Handles incoming IPv4 packets.
- exp—Handles MPLS packets using Layer 2 headers.
- ieee-802.1—Handles Layer 2 CoS.
- inet-precedence—Handles incoming IPv4 packets. IP precedence mapping requires only the upper three bits of the DSCP field.

For example, you can set up the following configuration:

```
[edit class-of-service]
code-point-aliases {
   dscp {
```

```
        my1 110001;
        my2 101110;
        be 000001;
        cs7 110000;
    }
}
```

The sample configuration produces this mapping:

```
user@host>show class-of-service code-point-aliases dscp
Alias  Bit pattern
ef/my2  101110
af11   001010
af12   001100
af13   001110
af21   010010
af22   010100
af23   010110
af31   011010
af32   011100
af33   011110
af41   100010
af42   100100
af43   100110
be   000001
cs1   001000
cs2   010000
cs3   011000
cs4   100000
cs5   101000
nc1/cs6/cs7 110000
nc2   111000
my1   110001
```

The following notes explain certain results in the mapping:

- my1 110001:
  - 110001 was not mapped to anything before, and my1 is a new alias.
  - Nothing in the default mapping table is changed by this statement.
- my2 101110:
  - 101110 is now mapped to my2 as well as ef.
- be 000001:
  - be is now mapped to 000001.
  - The old value of be, 000000, is not associated with any alias. Packets with this DSCP value are now mapped to the default forwarding class.
- cs7 110000:
  - cs7 is now mapped to 110000, as well as nc1 and cs6.
  - The old value of cs7, 111000, is still mapped to nc2.

Configuring RED Drop Profiles for Congestion Control

If the device must support assured forwarding, you can control congestion by configuring random early detection (RED) drop profiles. RED drop profiles use drop probabilities for different levels of buffer fullness to determine which scheduling queue on the device is likely to drop assured forwarding packets under congested conditions. The device can drop packets when the queue buffer becomes filled to the configured percentage.

Assured forwarding traffic with the PLP (packet loss priority) bit set is more likely to be discarded than traffic without the PLP bit set. This example shows how to configure a drop probability and a queue fill level for both PLP and non-PLP assured forwarding traffic. It is only one example of how to use RED drop profiles.

The example shows how to configure the RED drop profiles listed in Table 202 on page 620.

Table 202: Sample RED Drop Profiles

Drop Profile	Drop Probability	Queue Fill Level
af-normal—For non-PLP (normal) assured forwarding traffic	Between 0 (never dropped) and 100 percent (always dropped)	Between 95 and 100 percent
af-with-plp—For PLP (aggressive packet dropping) assured forwarding traffic	Between 95 and 100 percent (always dropped)	Between 80 and 95 percent

To configure RED drop profiles for assured forwarding congestion control on the device:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 203 on page 621.
3. If you are finished configuring the device, commit the configuration.
4. Go on to one of the following tasks:
   - To assign resources, priorities, and profiles to output queues, see "Configuring Schedulers" on page 623.
   - To apply rules to logical interfaces, see "Configuring and Applying Virtual Channels" on page 631.
   - To use adaptive shapers to limit bandwidth for Frame Relay, see "Configuring and Applying Adaptive Shaping for Frame Relay" on page 635.
   - To check the configuration, see "Verifying a CoS Configuration" on page 682.

Table 203: Configuring RED Drop Profiles for Assured Forwarding Congestion Control

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Class of service** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit class-of-service
Configure the lower drop probability for normal, non-PLP traffic.	1. Click **Add new entry** next to Drop profiles. 2. In the Profile name box, type the name of the drop profile—for example, af-normal. 3. Click **Configure** next to Interpolate. 4. Click **Add new entry** next to Drop probability. 5. In the Value box, type a number for the first drop point—for example, 0. 6. Click **OK**. 7. Click **Add new entry** next to Drop probability again. 8. In the Value box, type a number for the next drop point—for example, 100. 9. Click **OK**.	Enter edit drop-profiles af-normal interpolate set drop-probability 0 set drop-probability 100
Configure a queue fill level for the lower non-PLP drop probability.	1. Click **Add new entry** next to Fill level. 2. In the Value box, type a number for the first fill level—for example, 95. 3. Click **OK**. 4. Click **Add new entry** next to Fill level. 5. In the Value box, type a number for the next fill level—for example, 100. 6. Click **OK** three times.	Enter set fill-level 95 set fill-level 100
Configure the higher drop probability for PLP traffic.	1. Click **Add new entry** next to Drop profiles. 2. In the Profile name box, type the name of the drop profile—for example, af-with-plp. 3. Click **Configure** next to Interpolate. 4. Click **Add new entry** next to Drop probability. 5. In the Value box, type a number for the first drop point—for example, 95. 6. Click **OK**. 7. Click **Add new entry** next to Drop probability. 8. In the Value box, type a number for the next drop point—for example, 100. 9. Click **OK**.	From the [edit class of service] hierarchy level, enter edit drop-profiles af-with-PLP interpolate set drop-probability 95 set drop-probability 100

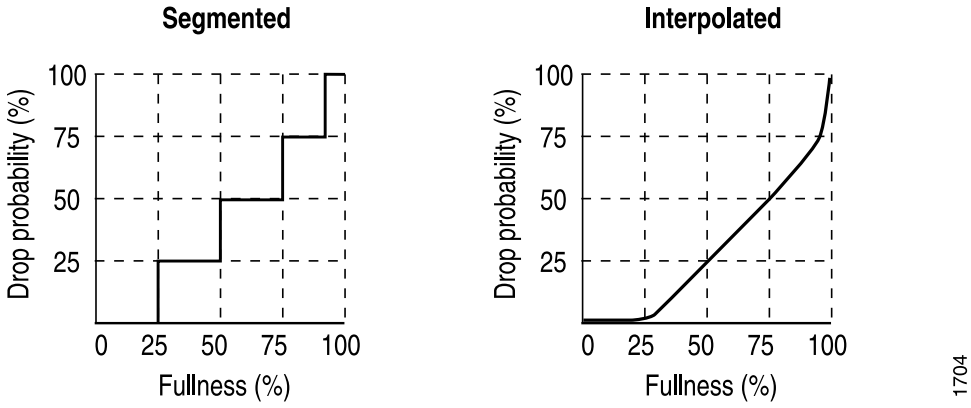
Table 203: Configuring RED Drop Profiles for Assured Forwarding Congestion Control *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure a queue fill level for the higher PLP drop probability.	1. Click **Add new entry** next to Fill level. 2. In the Value box, type a number for the first fill level—for example, 80. 3. Click **OK**. 4. Click **Add new entry** next to Fill level. 5. In the Value box, type a number for the next fill level—for example, 95. 6. Click **OK**.	Enter set fill-level 80 set fill-level 95

Example: Configuring RED Drop Profiles

Create a segmented configuration and an interpolated configuration that correspond to the graphs in Figure 93 on page 622. The values defined in the configuration are matched to represent the data points in the graph line. In this example, the drop probability is 25 percent when the queue is 50 percent full. The drop probability increases to 50 percent when the queue is 75 percent full.

Figure 93: Segmented and Interpolated Drop Profiles



Segmented

```
class-of-service {
   drop-profiles {
      segmented-style-profile {
         fill-level 25 drop-probability 25;
         fill-level 50 drop-probability 50;
         fill-level 75 drop-probability 75;
         fill-level 95 drop-probability 100;
      }
   }
}
```

To create the profile's graph line, the software begins at the bottom-left corner, representing a 0 percent fill level and a 0 percent drop probability. This configuration

draws a line directly to the right until it reaches the first defined fill level, 25 percent for this configuration. The software then continues the line vertically until the first drop probability is reached. This process is repeated for all of the defined levels and probabilities until the top-right corner of the graph is reached.

Create a smoother graph line by configuring the profile with the interpolate statement. This allows the software to automatically generate 64 data points on the graph beginning at (0, 0) and ending at (100, 100). Along the way, the graph line intersects specific data points, which you define as follows:

Interpolated

```
class-of-service {
   drop-profiles {
      interpolated-style-profile {
         interpolate {
            fill-level [ 50 75 ];
            drop-probability [ 25 50 ];
         }
      }
   }
}
```

Configuring Schedulers

You configure schedulers to assign resources, priorities, and drop profiles to output queues. By default, only queues 0 and 3 have resources assigned.

NOTE: SRX-series devices support hierarchical schedulers, including per-unit-schedulers. For more information, see "Configuring CoS Hierarchical Schedulers" on page 652.

This example creates the schedulers listed in Table 204 on page 623.

Table 204: Sample Schedulers

Scheduler	For CoS Traffic Type	Assigned Priority	Allocated Portion of Queue Buffer	Assigned Bandwidth (Transmit Rate)
be-scheduler	Best-effort traffic	Low	40 percent	10 percent
ef-scheduler	Expedited forwarding traffic	High	10 percent	10 percent
af-scheduler	Assured forwarding traffic	High	45 percent	45 percent
nc-scheduler	Network control traffic	Low	5 percent	5 percent

To configure schedulers for the device:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 205 on page 624.

3. Go on to "Configuring and Applying Scheduler Maps" on page 627.

Table 205: Configuring Schedulers

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Class of service** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit class-of-service
Configure a best-effort scheduler.	1. Click **Add new entry** next to Schedulers. 2. In the Scheduler name box, type the name of the best-effort scheduler—for example, be-scheduler.	Enter edit schedulers be-scheduler
Configure a best-effort scheduler priority and buffer size.	1. In the Priority box, type low. 2. Click **Configure** next to Buffer size. 3. From the Buffer size choice list, select the basis for the buffer allocation method—for example, **Percent**. 4. In the Percent box, type the percentage of the buffer to be used by the best-effort scheduler—for example, 40. 5. Click **OK**.	Enter set priority low set buffer-size percent 40
Configure a best-effort scheduler transmit rate.	1. Click **Configure** next to Transmit rate. 2. From the Transmit rate choice list, select the basis for the transmit rate method—for example, **Percent**. 3. In the Percent box, type the percentage of the bandwidth to be used by the best-effort scheduler—for example, 10. 4. Click **OK** twice.	Enter set transmit-rate percent 10
Configure an expedited forwarding scheduler.	1. Click **Add new entry** next to Schedulers. 2. In the Scheduler name box, type the name of the expedited forwarding scheduler—for example, ef-scheduler.	From the [edit class of service] hierarchy level, enter edit schedulers ef-scheduler
Configure an expedited forwarding scheduler priority and buffer size.	1. In the Priority box, type high. 2. Click **Configure** next to Buffer size. 3. From the Buffer size choice list, select the basis for the buffer allocation method—for example, **Percent**. 4. In the Percent box, type the percentage of the buffer to be used by the expedited forwarding scheduler—for example, 10. 5. Click **OK**.	Enter set priority high set buffer-size percent 10

Table 205: Configuring Schedulers *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure an expedited forwarding scheduler transmit rate.	1. Click **Configure** next to Transmit rate. 2. From the Transmit rate choice list, select the basis for the transmit rate method—for example, **Percent**. 3. In the Percent box, type the percentage of the bandwidth to be used by the expedited forwarding scheduler—for example, 10. 4. Click **OK** twice.	Enter set transmit-rate percent 10
Configure an assured forwarding scheduler.	1. Click **Add new entry** next to Schedulers. 2. In the Scheduler name box, type the name of the assured forwarding scheduler—for example, af-scheduler.	From the [edit class of service] hierarchy level, enter edit schedulers af-scheduler
Configure an assured forwarding scheduler priority and buffer size.	1. In the Priority box, type high. 2. Click **Configure** next to Buffer size. 3. From the Buffer size choice list, select the basis for the buffer allocation method—for example, **Percent**. 4. In the Percent box, type the percentage of the buffer to be used by the assured forwarding scheduler—for example, 45. 5. Click **OK**.	Enter set priority high set buffer-size percent 45
Configure an assured forwarding scheduler transmit rate.	1. Click **Configure** next to Transmit rate. 2. From the Transmit rate choice list, select the basis for the transmit rate method—for example, **Percent**. 3. In the Percent box, type the percentage of the bandwidth to be used by the assured forwarding scheduler—for example, 45. 4. Click **OK**.	Enter set transmit-rate percent 45
(Optional) Configure a drop profile map for assured forwarding low and high priority. (DiffServ can have a RED drop profile associated with assured forwarding.)	1. Click **Add new entry** next to Drop profile map. 2. From the Loss priority box, select **Low**. 3. From the Protocol box, select **Any**. 4. In the Drop profile box, type the name of the drop profile—for example, af-normal. 5. Click **OK**. 6. Click **Add new entry** next to Drop profile map. 7. From the Loss priority box, select **High**. 8. From the Protocol box, select **Any**. 9. In the Drop profile box, type the name of the drop profile—for example, af-with-PLP. 10. Click **OK** twice.	Enter set drop-profile-map loss-priority low protocol any drop-profile af-normal set drop-profile-map loss-priority high protocol any drop-profile af-with-PLP

Table 205: Configuring Schedulers *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure a network control scheduler.	1. Click **Add new entry** next to Schedulers. 2. In the Scheduler name box, type the name of the network control scheduler—for example, nc-scheduler.	From the [edit class of service] hierarchy level, enter edit schedulers nc-scheduler
Configure a network control scheduler priority and buffer size.	1. In the Priority box, type low. 2. Click **Configure** next to Buffer size. 3. From the Buffer size choice list, select the basis for the buffer allocation method—for example, **Percent**. 4. In the Percent box, type the percentage of the buffer to be used by the network control scheduler—for example, 5. 5. Click **OK**.	Enter set priority low set buffer-size percent 5
Configure a network control scheduler transmit rate.	1. Click **Configure** next to Transmit rate. 2. From the Transmit rate choice list, select the basis for the transmit rate method—for example, **Percent**. 3. In the Percent box, type the percentage of the bandwidth to be used by the network control scheduler—for example, 5. 4. Click **OK**.	Enter set transmit-rate percent 5

Example: Configuring Priority Scheduling

The JUNOS software supports multiple levels of transmission priority, which in order of increasing priority are low, medium-low, medium-high, and high, and strict-high. This allows the software to service higher-priority queues before lower-priority queues.

Priority scheduling determines the order in which an output interface transmits traffic from the queues, thus ensuring that queues containing important traffic are provided better access to the outgoing interface. This is accomplished through a procedure in which the software examines the priority of the queue. In addition, the software determines if the individual queue is within its defined bandwidth profile. This binary decision, which is reevaluated on a regular time cycle, compares the amount of data transmitted by the queue against the amount of bandwidth allocated to it by the scheduler. When the transmitted amount is less than the allocated amount, the queue is considered to be in profile. A queue is out of profile when its transmitted amount is larger than its allocated amount.

The queues for a given output physical interface (or output logical interface if per-unit scheduling is enabled on that interface) are divided into sets based on their priority. Any such set contains queues of the same priority.

The software traverses the sets in descending order of priority. If at least one of the queues in the set has a packet to transmit, the software selects that set. A queue

from the set is selected based on the weighted round robin (WRR) algorithm, which operates within the set.

You can configure priority scheduling, as shown in the following example:

1. Configure a scheduler, be-sched, with medium-low priority.

```
[edit class-of-service]
schedulers {
   be-sched {
      priority medium-low;
   }
}
```

2. Configure a scheduler map, be-map, that associates be-sched with the best-effort forwarding class.

```
[edit class-of-service]
scheduler-maps {
   be-map {
      forwarding-class best-effort scheduler be-sched;
   }
}
```

3. Assign be-map to a Gigabit Ethernet interface, ge-0/0/0.

```
[edit class-of-service]
interfaces {
   ge-0/0/0 {
      scheduler-map be-map;
   }
}
```

Configuring and Applying Scheduler Maps

You configure a scheduler map to assign a forwarding class to a scheduler, then apply the scheduler map to any interface that must enforce DiffServ CoS.

The following example shows how to create the scheduler map diffserv-cos-map and apply it to the device's Ethernet interface ge-0/0/0. The map associates the mf-classifier forwarding classes configured in "Configuring and Applying a Firewall Filter for a Multifield Classifier" on page 601 to the schedulers configured in "Configuring Schedulers" on page 623, as shown in Table 206 on page 627.

Table 206: Sample diffserv-cos-map Scheduler Mapping

mf-classifier Forwarding Class	For CoS Traffic Type	diffserv-cos-map Scheduler
be-class	Best-effort traffic	be-scheduler
ef-class	Expedited forwarding traffic	ef-scheduler
af-class	Assured forwarding traffic	af-scheduler

Table 206: Sample diffserv-cos-map Scheduler Mapping *(continued)*

mf-classifier Forwarding Class	For CoS Traffic Type	diffserv-cos-map Scheduler
nc-class	Network control traffic	nc-scheduler

To configure and apply scheduler maps:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 207 on page 628.
3. If you are finished configuring the device, commit the configuration.
4. Go on to one of the following tasks:
   - To apply rules to logical interfaces, see "Configuring and Applying Virtual Channels" on page 631.
   - To use adaptive shapers to limit bandwidth for Frame Relay, see "Configuring and Applying Adaptive Shaping for Frame Relay" on page 635.
   - To check the configuration, see "Verifying a CoS Configuration" on page 682.

Table 207: Configuring Scheduler Maps

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Class of service** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit class-of-service
Configure a scheduler map for DiffServ CoS.	1. Click **Add new entry** next to Scheduler maps. 2. In the Map name box, type the name of the scheduler map—for example, diffserv-cos-map.	Enter edit scheduler-maps diffserv-cos-map
Configure a best-effort forwarding class and scheduler.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured best-effort forwarding class—be-class. 3. In the Scheduler box, type the name of the previously configured best-effort scheduler—be-scheduler. 4. Click **OK**.	Enter set forwarding-class be-class scheduler be-scheduler

Table 207: Configuring Scheduler Maps *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure an expedited forwarding class and scheduler.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured expedited forwarding class—ef-class. 3. In the Scheduler box, type the name of the previously configured expedited forwarding scheduler—ef-scheduler. 4. Click **OK**.	Enter set forwarding-class ef-class scheduler ef-scheduler
Configure an assured forwarding class and scheduler.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured assured forwarding class—af-class. 3. In the Scheduler box, type the name of the previously configured assured forwarding scheduler—af-scheduler. 4. Click **OK**.	Enter set forwarding-class af-class scheduler af-scheduler
Configure a network control class and scheduler.	1. Click **Add new entry** next to Forwarding class. 2. In the Class name box, type the name of the previously configured network control class—nc-class. 3. In the Scheduler box, type the name of the previously configured network control scheduler—nc-scheduler. 4. Click **OK** twice.	Enter set forwarding-class nc-class scheduler nc-scheduler
Apply the scheduler map to an interface. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. Click **Add new entry** next to Interfaces. 2. In the Interface name box, type the name of the interface—for example, ge-0/0/0. 3. Click **Add new entry** next to Unit. 4. In the Unit number box, type the logical interface unit number—0. 5. In the Scheduler map box, type the name of the previously configured scheduler map—diffserv-cos-map. 6. Click **OK**.	From the [edit class of service] hierarchy level, enter set interfaces ge-0/0/0 scheduler-map diffserv-cos-map

Scheduler Maps: Sample Configuration

Once you define a scheduler, you can include it in a *scheduler map*, which maps a specified forwarding class to a scheduler configuration. To do this, include the scheduler-maps statement at the [edit class-of-service] hierarchy level:

```
[edit class-of-service]
scheduler-maps {
    map-name {
        forwarding-class class-name scheduler scheduler-name;
    }
}
```

After you have defined the scheduler map, you can associate it with an output interface. To do this, include the scheduler-map statement at the [edit class-of-service interfaces *interface-name*] hierarchy level:

```
[edit class-of-service interfaces interface-name]
scheduler-map map-name;
```

Interface wildcards are supported.

Schedulers: Sample Configuration

You use *schedulers* to define the properties of output queues. These properties include the amount of interface bandwidth assigned to the queue, the size of the memory buffer allocated for storing packets, the priority of the queue, and the random early detection (RED) drop profiles associated with the queue.

You associate the schedulers with forwarding classes by means of *scheduler maps*. You can then associate each scheduler map with an interface, thereby configuring the hardware queues, packet schedulers, and RED processes that operate according to this mapping.

To configure class-of-service (CoS) schedulers, use the following sample configuration at the [edit class-of-service] hierarchy level:

```
[edit class-of-service]
interfaces {
    interface-name {
        scheduler-map map-name;
        scheduler-map-chassis map-name;
        schedulers number;
        shaping-rate rate;
        unit {
            output-traffic-control-profile profile-name;
            scheduler-map map-name;
            shaping-rate rate;
        }
    }
}
fabric {
    scheduler-map {
        priority (high | low) scheduler scheduler-name;
```

```
    }
  }
  scheduler-maps {
    map-name {
      forwarding-class class-name scheduler scheduler-name;
    }
  }
  schedulers {
    scheduler-name {
      buffer-size (percent percentage | remainder | temporal microseconds );
      drop-profile-map loss-priority (any | low | medium-low | medium-high | high) protocol
        (any | non-tcp | tcp) drop-profile profile-name;
      priority priority-level;
      transmit-rate (rate  | percent percentage remainder) <exact | rate-limit>;
    }
  }
  traffic-control-profiles profile-name {
    delay-buffer-rate (percent percentage | rate);
    guaranteed-rate (percent percentage | rate);
    scheduler-map map-name;
    shaping-rate (percent percentage | rate);
  }
```

Configuring and Applying Virtual Channels

You configure a virtual channel to set up queuing, packet scheduling, and accounting rules to be applied to one or more logical interfaces. You then must apply the virtual channel to a particular logical interface. Virtual channels can be applied in different ways. In the example here, an output firewall filter is used for directing traffic to a particular virtual channel.

The following example shows how to create the virtual channels branch1–vc, branch2–vc, and branch3–vc and apply them in the firewall filter choose-vc to the Services Router's T3 interface t3-1/0/0.

To configure and apply virtual channels for the Services Router:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 208 on page 632.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following tasks:
   - To assign resources, priorities, and profiles to output queues, see "Configuring Schedulers" on page 623.
   - To use adaptive shapers to limit bandwidth for Frame Relay, see "Configuring and Applying Adaptive Shaping for Frame Relay" on page 635.
   - To check the configuration, see "Verifying a CoS Configuration" on page 682.

Table 208: Configuring and Applying Virtual Channels

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Class of service** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit class-of-service
Define the virtual channels branch1–vc, branch2–vc, branch3–vc, and the default virtual channel. You must specify a default virtual channel.	1. Click **Add new entry** next to Virtual channels. 2. In the Channel name box, type the name of the virtual channel—for example, branch1–vc. 3. Click **OK**. 4. Create additional virtual channels for branch2–vc, branch3–vc, and default-vc.	1. Enter set virtual-channels branch1–vc 2. Repeat this statement for branch2–vc, branch3–vc, and default-vc.
Define the virtual channel group wan-vc-group to include the four virtual channels, and assign each virtual channel the scheduler map bestscheduler.	1. Click **Add new entry** next to Virtual channel groups. 2. In the Group name box, type the name of the virtual channel group—wan-vc-group. 3. Click **Add new entry** next to Channel. 4. In the Channel name box, type the name of the previously configured virtual channels—branch1–vc. 5. In the Scheduler map box, type the name of the previously configured scheduler map—bestscheduler. 6. Click **OK**. 7. Add the virtual channels branch2–vc, branch3–vc, and default-vc. Select the **Default** box when adding the virtual channel default-vc.	1. Enter set virtual-channel-groups wan-vc-group branch1–vc scheduler-map bestscheduler 2. Repeat this statement for branch2–vc, branch3–vc, and default-vc. 3. Enter set virtual-channel-groups wan-vc-group default–vc default

Table 208: Configuring and Applying Virtual Channels *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Specify a shaping rate of 2 Mbps for each virtual channel within the virtual channel group.	1. Click **branch1–vc** in the list of virtual channels. 2. Select the **Shaping rate** box. 3. Click **Configure**. 4. Select **Absolute rate** from the Rate choice box. 5. In the Absolute rate box, type the shaping rate—2m. 6. Add the shaping rate for the branch2–vc and branch3–vc virtual channels. 7. Click **OK** three times.	1. Enter set virtual-channel-groups wan-vc-group branch1–vc shaping-rate 2m 2. Repeat this statement for branch2–vc and branch3–vc.
Apply the virtual channel group to the logical interface t3–1/0/0.0. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. Click **Add new entry** next to Interfaces. 2. In the Interface name box, type the name of the interface—t3–1/0/0. 3. Click **Add new entry** next to Unit. 4. In the Unit number box, type the logical interface unit number—0. 5. In the Virtual channel group box, type the name of the previously configured virtual channel group—wan-vc-group. 6. Click **OK**.	From the [edit class of service] hierarchy level, enter set interfaces t3–1/0/0 unit 0 virtual-channel-group wan-vc-group

Table 208: Configuring and Applying Virtual Channels *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor

Create the firewall filter choose-vc to select the traffic that is transmitted on a particular virtual channel.	1. On the main Configuration page next to Firewall, click **Configure** or **Edit**.
	2. Click **Add new entry** next to Filter.
	3. In the Filter name box, type the name of the firewall filter—choose-vc.
	4. Click **Add new entry** next to Term.
	5. In the Rule name box, type the name of the firewall term—branch1.
	6. Click **Configure** next to From.
	7. Click **Add new entry** next to Destination address.
	8. In the Address box, type the IP address of the destination host—192.168.10.0/24.
	9. Click **OK** twice.
	10. On the firewall term page, click **Configure** next to Then.
	11. Select **Accept** from the Designation box.
	12. In the Virtual channel box, type the name of the previously configured virtual channel—branch1–vc.
	13. Click **OK**.
	14. Repeat these steps for the virtual channels branch2–vc and branch3–vc.
		1. From the [edit] hierarchy level, enter
		`edit firewall`
		2. Enter
		`set family inet filter choose-vc term branch1 from destination 192.168.10.0/24`
		3. Enter
		`set family inet filter choose-vc term branch1 then accept`
		4. Enter
		`set family inet filter choose-vc term branch1 then virtual-channel branch1–vc`
		5. Repeat these steps for virtual channels branch2–vc and branch3–vc.

Apply the firewall filter choose-vc to output traffic on the t3–1/0/0.0 interface.	1. On the main Configuration page next to Interfaces, click **Configure** or **Edit**.
	2. Click **t3–1/0/0** in the list of configured interfaces.
	3. Click **0** in the list of configured logical units for the interface.
	4. Click **Edit** next to Inet.
	5. Click **Configure** next to Filter.
	6. In the Output box, type the name of the previously configured firewall filter—choose–vc.
	7. Click **OK**.
		1. From the [edit] hierarchy level, enter
		`edit interfaces`
		2. Enter
		`set t3–1/0/0 unit 0 family inet filter output choose-vc`

Configuring and Applying Adaptive Shaping for Frame Relay

You can use adaptive shaping to limit the bandwidth of traffic flowing on a Frame Relay logical interface. If you configure and apply adaptive shaping, the device checks the backward explicit congestion notification (BECN) bit within the last inbound (ingress) packet received on the interface. If the BECN bit is set, the device limits the transmit bandwidth on the interface to the configured adaptive shaper maximum transmit rate. If the BECN bit is not set, the transmit bandwidth is not limited and is allowed to exceed the adaptive shaper rate.

For more information about adaptive shapers for a Frame Relay interface, see the *JUNOS Class of Service Configuration Guide*.

The following example shows how to create adaptive shaper fr-shaper and apply it to the device's T1 interface t1-0/0/2. The adapter shaper limits the transmit bandwidth on the interface to 64 Kbps.

To configure and apply an adaptive shaper for the device:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 209 on page 635.
3. If you are finished configuring the device, commit the configuration.
4. Go on to one of the following tasks:
   - To assign resources, priorities, and profiles to output queues, see "Configuring Schedulers" on page 623.
   - To apply rules to logical interfaces, see "Configuring and Applying Virtual Channels" on page 631.
   - To check the configuration, see "Verifying a CoS Configuration" on page 682.

Table 209: Configuring and Applying an Adaptive Shaper

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Class of service** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit class-of-service

Table 209: Configuring and Applying an Adaptive Shaper *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define the adaptive shaper name and maximum transmit rate.	1. Next to Adaptive Shapers, click **Add new entry**. 2. In the Adaptive shaper name box, type fr-shaper. 3. Next to Trigger, click **Add new entry**. 4. Next to Becn, select the check box. 5. Next to Shaping rate, select the check box and click **Configure**. 6. From the Rate choice list, select **Absolute rate**. 7. In the Absolute rate box, type 64k. 8. Click **OK** three times.	Enter set adaptive-shapers fr-shaper trigger becn shaping-rate 64k
Apply the adaptive shaper to the logical interface t1-0/0/2.0. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. Next to Interfaces, click **Add new entry**. 2. In the Interface name box, type the name of the interface—t1-0/0/2. 3. Next to Unit, click **Add new entry**. 4. In the Unit number box, type the logical interface unit number—0. 5. In the Adaptive shaper box, type the name of the adaptive shaper—fr-shaper. 6. Click **OK**.	Enter set interfaces t1-0/0/2 unit 0 adaptive-shaper fr-shaper

Configuring CoS Queuing for Tunnels with a Configuration Editor

CoS queuing, scheduling, and shaping allow you to control and improve the flow of traffic through tunnel interfaces like GRE and IP-IP interfaces. The GRE and IP-IP interfaces on a J-series router are internal, configurable interfaces named gr-0/0/0 and ip-0/0/0.

To configure CoS for a GRE or IP-IP tunnel, you must first enable tunnel queuing on the router. If tunnel queuing is not enabled, the router continues to send traffic through the tunnel but ignores any configured CoS schedulers and shapers.

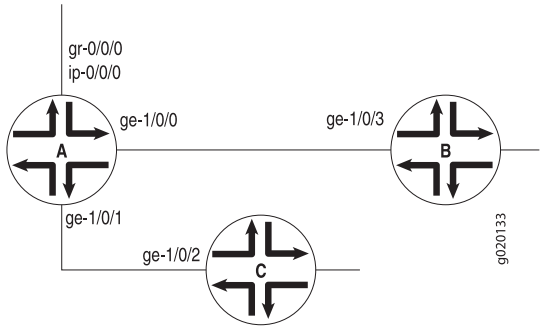
NOTE: You cannot enable tunnel queuing on J-series interfaces other than tunnel interfaces, although the router allows you to commit such a configuration.

You then define the GRE or IP-IP tunnel interface and its per-unit scheduler and set a line rate for the tunnel with the CoS shaper.

Configuring CoS for GRE Tunnels

In the network shown in Figure 94 on page 637, Router A acts as a tunnel ingress device. The link between GRE tunnel interfaces ge-1/0/0 in Router A and ge-1/0/3 in Router B is the GRE tunnel. Router A monitors the traffic received from interface ge-1/0/3. By way of interface ge-1/0/2, Router C generates traffic to Router B.

Figure 94: Configuring CoS Queing for GRE Tunnels



To configure COS queuing for GRE or IP-IP tunnels:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 210 on page 638 to configure CoS queuing for tunnel interfaces.
   a. Enable tunnel queuing on the router.
   b. Define the GRE or IP-IP tunnel interface.
   c. Define the per-unit scheduler for the GRE or IP-IP tunnel interface.
   d. Define the tunnel's line rate using the shaper definition.
3. Configure forwarding classes and schedulers.

   For information on configuring forwarding classes, see "Assigning Forwarding Classes to Output Queues" on page 604. For information on configuring schedulers, see "Configuring Schedulers" on page 623.
4. Configure a scheduler map and apply the scheduler map to the tunnel interface. For information on configuring a scheduler map, see "Configuring and Applying Scheduler Maps" on page 627.
5. Configure classifiers and apply them to the tunnel interface.

   For information on configuring classifiers, see "Configuring and Applying Behavior Aggregate Classifiers" on page 614.
6. Create rewrite rules and apply them to the tunnel interface.

   For information on configuring rewrite rules, see "Configuring and Applying Rewrite Rules" on page 611.

7. If you are finished configuring the router, commit the configuration.
8. Go on to one of the following tasks:
   - To configure other CoS components, see "Configuring CoS Components with a Configuration Editor" on page 599.
   - To check the configuration, see "Verifying a CoS Configuration" on page 682.

Table 210: Configuring CoS for GRE Tunnels

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Chassis** level in the configuration hierarchy, and enable tunnel queuing on the router.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration.** 2. Next to Chassis, click **Configure** or **Edit**. 3. Next to Fpc, click **Add new entry.** 4. Next to Slot, type **0.** 5. Next to Pic, click **Add New Entry.** 6. Next to Slot, type **0.** 7. Select the check box next to Tunnel queuing. 8. Click **OK** until you return to the main Configuration page.	From the [edit] hierarchy level, enter `edit chassis fpc 0 pic 0 tunnel-queuing`
Navigate to the **Interfaces** level in the configuration hierarchy, and define the GRE tunnel interface gr-0/0/0.	1. On the main Configuration page, next to Interfaces click **Configure** or **Edit**. 2. In the Interfaces name box, type gr-0/0/0. 3. Next to Unit, click **Add new entry**. 4. In the Interfaces unit number box, type 0. 5. Click **OK**.	From the [edit] hierarchy level, enter `edit interfaces gr-0/0/0 unit 0`
Define the per-unit scheduler for the GRE tunnel interface.	1. From the Scheduler type list, select **Per unit scheduler**. 2. Click **OK** until you return to the main Configuration page.	From the [edit] hierarchy level, enter `set interfaces gr-0/0/0 per-unit-scheduler`

Table 210: Configuring CoS for GRE Tunnels *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the Class of service level in the configuration hierarchy, and define the GRE tunnel's line rate (for example, 100 Mbps) using the shaper definition.	1. On the main configuration page next to Class of service, click **Configure** or **Edit** 2. Next to Interfaces, click **Add new entry.** 3. In the Interface name box, type the name of the interface gr-0/0/0. 4. Next to unit, click **Add New Entry.** 5. In the Interface unit number box, type the logical interface unit number 0. 6. Select the Shaping rate check box, and click **Configure** 7. Next to Shaping Rate choice, select **Rate**. 8. In the Rate box, type 100m. 9. Click **OK** until you return to the main Class of Service configuration page.	From the [edit] hierarchy level, enter set class-of-service interfaces gr-0/0/0 unit 0 shaping-rate 100m

Preserving the ToS Value of a Tunneled Packet

To ensure that the tunneled packet continues to have the same CoS treatment even in the physical interface, you must preserve the type-of-service (ToS) value from the inner IP header to the outer IP header.

For transit traffic, JUNOS software preserves the CoS value of the tunnel packet for both GRE and IP-IP tunnel interfaces. The inner IPv4 or IPv6 ToS bits are copied to the outer IPv4 ToS header for both types of tunnel interfaces.

For Routing Engine traffic, however, the router handles GRE tunnel interface traffic differently from IP-IP tunnel interface traffic. Unlike for IP-IP tunnels, the IPv4 ToS bits are not copied to the outer IPv4 header by default. You have a configuration option to copy the ToS value from the packet's inner IPv4 header to the outer IPv4 header.

To copy the inner ToS bits to the outer IP header (which is required for some tunneled routing protocols) on packets sent by the Routing Engine, include the copy-tos-to-outer-ip-header statement at the logical unit hierarchy level of a GRE interface.

NOTE: For IPv6 traffic, the inner ToS value is not copied to the outer IPv4 header for both GRE and IP-IP tunnel interfaces even if the copy-tos-to-outer-ip-header statement is specified.

This example copies the inner ToS bits to the outer IP header on a GRE tunnel:

```
[edit interfaces]
gr-0/0/0 {
```

```
    unit 0 {
      copy-tos-to-outer-ip-header;
      family inet;
    }
  }
```

Configuring Strict High Priority for Queuing with a Configuration Editor

You can configure one queue per interface to have strict high priority, which causes delay-sensitive traffic, such as voice traffic, to be removed and forwarded with minimum delay. Packets that are queued in a strict-priority queue are removed before packets in other queues, including high-priority queues.

The strict high-priority queuing feature allows you to configure traffic policing that prevents lower-priority queues from being starved. The strict-priority queue does not cause starvation of other queues because the configured policer allows the queue to exceed the configured bandwidth only when other queues are not congested. If the interface is congested, the software polices strict-priority queues to the configured bandwidth.

To prevent queue starvation of other queues, you must configure an output (egress) policer that defines a limit for the amount of traffic that the queue can service. The software services all traffic in the strict-priority queue that is under the defined limit. When strict-priority traffic exceeds the limit, the policer marks the traffic in excess of the limit as out-of-profile. If the output port is congested, the software drops out-of-profile traffic.

You can also configure a second policer with an upper limit. When strict-priority traffic exceeds the upper limit, the software drops the traffic in excess of the upper limit, regardless of whether the output port is congested. This upper-limit policer is not a requirement for preventing starvation of the lower-priority queues. The policer for the lower limit, which marks the packets as out-of-profile, is sufficient to prevent starvation of other queues.

The sample strict-high priority queuing configuration does the following:

1. Uses a behavior aggregate (BA) classifier to classify traffic based on the IP precedence of the packet. The classifier defines IP precedence value 101 as voice traffic and 000 as data traffic.
2. To minimize delay, assigns all delay-sensitive packets to the strict-priority queue.
3. Configures two policers on the output interface that identify excess voice traffic belonging to the voice-class forwarding class. If the traffic exceeds 1 Mbps, a policer marks the traffic in excess of 1 Mbps as out-of-profile. If the traffic exceeds 2 Mbps, the second policer discards the traffic in excess of 2 Mbps.

To configure strict-priority queuing and prevent starvation of other queues:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 211 on page 641.
3. If you are finished configuring the router, commit the configuration.

Table 211: Configuring Strict-High Priority Queuing and Starvation Prevention

Task	J-Web Configuration Editor	CLI Configuration Editor
Configuring a BA Classifier		
Use a BA classifier to classify traffic based on the IP precedence of the packet. The classifier defines IP precedence value 101 as voice traffic and 000 as data traffic.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**. 3. Next to Classifiers, click **Configure** or **Edit**. 4. Next to Inet precedence, click **Add new entry**. 5. Enter corp-traffic in the Name box. 6. Next to Forwarding class, click **Add new entry**. 7. Enter voice-class in the Class name box. 8. Next to Loss priority, click **Add new entry**. 9. Enter low in the Loss val box. 10. Next to Code points, click **Add new entry**. 11. Enter 101 in the Value box. 12. Click **OK** three times. 13. In the Inet precedence forwarding class page, enter voice-class in the Class name box. 14. Next to Loss priority, click **Add new entry**. 15. Enter high in the Loss val box. 16. Next to Code points, click **Add new entry**. 17. Enter 000 in the Value box. 18. Click **OK** five times.	From the [edit] hierarchy level, enter edit Class of service classifiers inet-precedence corp-traffic forwarding-class voice-class loss-priority low Enter set code-points 101 From the [edit] hierarchy level, enter edit class-of-service classifiers inet-precedence corp-traffic forwarding-class data-class loss-priority high Enter set code-points 000
Configuring the Forwarding Classes		

Table 211: Configuring Strict-High Priority Queuing and Starvation Prevention *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Assign priority queuing to voice and data traffic.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**. 3. Next to Forwarding classes, click **Configure** or **Edit**. 4. Next to Queue, click **Add new entry**. 5. Enter 0 in the Queue num box. 6. Enter voice-class in the Class name box. 7. Click **OK** to return to the Forwarding Classes page. 8. Next to Queue, click **Add new entry**. 9. Enter 1 in the Queue num box. 10. Enter data-class in the Class name box. 11. Click **OK** three times.	From the [edit] hierarchy level, enter edit class-of-service forwarding-classes queue 0 voice-class enter edit class-of-service forwarding-classes queue 1 data-class
Configuring the Scheduler Map and Schedulers		
Configure the scheduler map and voice scheduler.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**. 3. Next to Scheduler maps, click **Add new entry**. 4. In the Map name box, type corp-map. 5. Next to Forwarding class, click **Add new entry**. 6. In the Class name box, type voice-class. 7. In the Scheduler name box, type voice-sched. 8. Click **OK** three times.	From the [edit] hierarchy level, enter edit class-of-service scheduler-maps corp-map forwarding-class voice-class Enter set scheduler voice-sched

Table 211: Configuring Strict-High Priority Queuing and Starvation Prevention *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Define the voice and data traffic schedulers, and set the priority.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Class of service, click **Configure** or **Edit**. 3. Next to Schedulers, click **Add new entry**. 4. In the Scheduler name box, type voice-sched. 5. In the Priority box, type strict-high. 6. Click **OK**. 7. Next to Schedulers, click **Add new entry**. 8. In the Scheduler name box, type data-sched. 9. In the Priority box, type low. 10. Click **OK** twice.	From the [edit] hierarchy level, enter edit class-of-service schedulers voice-sched Enter set priority strict-high From the [edit] hierarchy level, enter edit class-of-service schedulers data-sched Enter set priority low
Applying the BA Classifier to an Input Interface and Scheduler Map to an Output Interface		

Table 211: Configuring Strict-High Priority Queuing and Starvation Prevention *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Apply the BA classifier to an input interface—for example, ge-0/0/0. Apply the scheduler map to and output interface—for example, e1-1/0/0. (See the interface naming conventions in "Network Interface Naming" on page 16.)	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Interfaces, click **Configure** or **Edit**. 3. Next to Interface, click **Add new entry**. 4. In the Interface name box, type ge-0/0/0. 5. Click **OK** three times. 6. In the Edit Configuration page, next to Class of service, click **Configure** or **Edit**. 7. Next to Classifiers, click **Edit**. 8. Next to Inet precedence, click **Add new entry**. 9. In the Name box, type corp-traffic. 10. Click **OK** three times. 11. In the Edit Configuration page, next to Interfaces, click **Configure** or **Edit**. 12. Next to Interface name, type e1-1/0/1. 13. Click **OK** twice. 14. In the Edit Configuration page, next to Class of service, click **Configure** or **Edit**. 15. Next to Scheduler maps, click **Add new entry**. 16. In the Map name box, type corp-map. 17. Click **OK** twice.	From the [edit] hierarchy level, enter edit interfaces ge-0/0/0 unit 0 From the [edit] hierarchy level, enter edit class of service classifiers inet-precedence corp-traffic From the [edit] hierarchy level, enter edit interfaces e1-1/0/0 unit 0 From the [edit] hierarchy level, enter edit class-of-service scheduler-maps corp-map
Configuring Two Policers		

Table 211: Configuring Strict-High Priority Queuing and Starvation Prevention *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Configure two policers: one as voice-drop and second as voice-excess.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Firewall, click **Configure** or **Edit**. 3. Next to Policer, click **Add new entry**. 4. In the Policer name box, type voice-drop. 5. Next to If Exceeding, select the check box and click **Configure**. 6. In the Burst size limit box, type 200000. 7. In the Bandwidth list, select **Bandwidth limit**. 8. In the Bandwidth limit box, type 2000000. 9. Click **OK**. 10. On the Policer page, next to Then, click **Configure**. 11. Next to Discard, select the check box. 12. Click Ok twice. 13. In the Firewall Configuration page next to Policer, click **Add new entry**. 14. In the Policer name box, type voice-excess. 15. Next to If Exceeding, select the check box and click **Configure**. 16. In the Burst size limit box, type 200000. 17. In the Bandwidth list, select **Bandwidth limit**. 18. In the Bandwidth limit box, type 1000000. 19. Click **OK**. 20. On the Policer page, next to Then, click **Configure**. 21. Next to Out of profile, select the check box. 22. Click **OK** twice.	From the [edit] hierarchy level, enter edit firewall policer voice-drop if-exceeding Enter set burst-size-limit 200000 bandwidth-limit 2000000 Enter set then discard From the [edit] hierarchy level, enter edit firewall policer voice-excess if-exceeding Enter set burst-size-limit 200000 bandwidth-limit 1000000 Enter set then out-of-profile

Table 211: Configuring Strict-High Priority Queuing and Starvation Prevention *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Create a firewall filter voice-term that includes the new policers. First, add the policer voice-drop to the term.	1. In the Firewall Configuration page next to Filter, click **Add new entry**. 2. In the Filter name box, type voice-term. 3. Next to Term click **Add new entry**. 4. In the Rule name box, type term 01. 5. Next to Term, click **Add new entry**. 6. Next to From, click **Configure**. 7. Next to Forwarding class choice, select **forwarding-class**. 8. Next to Forwarding class, click **Add new entry**. 9. In the String box, type voice-class. 10. Click **OK** twice. 11. In the Term Filter page, next to Then, click **Configure**. 12. Next to Policer choice, select **policer**. 13. In the Policer box, type voice-drop. 14. Next to Designation, select **Next**. 15. In the Next box, select term. 16. Click **OK** twice.	From the [edit] hierarchy level, enter edit firewall filter voice-term term 01 from forwarding-class voice-class then policer voice-drop next term
Then add the policer voice-excess to the term.	1. In the Firewall Filter page, next to Term, click **Add new entry**. 2. In the Rule name box, type term 02. 3. Next to From, click **Configure**. 4. Next to Forwarding class choice, select **forwarding-class**. 5. Next to Forwarding class, click **Add new entry**. 6. In the String box, type voice-class. 7. Click **OK** twice. 8. In the Term Filter page, next to Then, click **Configure**. 9. Next to Policer choice, select **policer**. 10. In the Policer box, type voice-excess. 11. Next to Designation, select **Accept**. 12. Click **OK** four times.	Enter edit firewall filter voice-term term 02 from forwarding-class voice-class then policer voice-excess accept
Applying the Filter to the Output Interface		

Table 211: Configuring Strict-High Priority Queuing and Starvation Prevention *(continued)*

Task	J-Web Configuration Editor	CLI Configuration Editor
Apply filter voice-term to e1-1/0/0 using the CLI.		From the [edit] hierarchy level, enter edit interfaces e1-1/0/1 unit 0 family inet filter output voice-term Enter set family inet address 11.1.1.1/24

Configuring Large Delay Buffers with a Configuration Editor

Large bursts of traffic from faster interfaces can cause congestion and dropped packets on slower interfaces that have small delay buffers. For example, a J-series Services Router operating at the edge of the network can drop a portion of the burst traffic it receives on a channelized T1/ E1 interface from a Fast Ethernet or Gigabit Ethernet interface on a router at the network core.

To ensure that traffic is queued and transmitted properly on slower interfaces, you can configure a buffer size larger than the default maximum. On J-series Services Routers, you can configure large delay buffers on channelized T1/E1 interfaces only.

This section contains the following topics:

Maximum Delay Buffer Sizes Available to Interfaces

When you enable the large delay buffer feature on interfaces, a larger buffer is available for allocation to scheduler queues. The maximum delay buffer size that is available for an interface depends on the maximum available delay buffer time and the speed of the interface.

On channelized T1/E1 interfaces, the maximum delay buffer time varies by the number of DS0 channels configured on the interface as shown in Table 212 on page 648. The default values are as follows:

- Clear-channel interface—The default delay buffer time is 500,000 microseconds (0.5 seconds).
- *N*xDS0 interface—The default delay buffer time is 1,200,000 microseconds (1.2 seconds).

Table 212: Maximum Available Delay Buffer Time by Channels

Channelized (*N*xDS0) Interfaces	Maximum Available Delay Buffer Time
1xDS0 through 3xDS0	4,000,000 microseconds (4 seconds)
4xDS0 through 7xDS0	2,000,000 microseconds (2 seconds)
8xDS0 through 15xDS0	1,000,000 microseconds (1 second)
16xDS0 through 32xDS0	500,000 microseconds (0.5 second)

You can calculate the maximum delay buffer size available for an interface, with the following formula:

interface speed x *maximum delay buffer time = maximum available delay buffer size*

For example, the following maximum delay buffer sizes are available to 1xDS0 and 2xDS0 interfaces:

1xDS0—64 kilobits per second x 4 seconds = 256 kilobits (32 kilobytes)

2xDS0—128 kilobits per second x 4 seconds = 512 kilobits (64 kilobytes)

If you configure a delay buffer size larger than the new maximum, the system allows you to commit the configuration but displays a system log warning message and uses the default buffer size setting instead of the configured maximum setting.

Delay Buffer Size Allocation Methods

You can specify delay buffer sizes for each queue using schedulers. The queue buffer can be specified as a period of time (microseconds) or as a percentage of the total buffer or as the remaining buffer. Table 213 on page 648 shows different methods that you can specify for buffer allocation in queues.

Table 213: Delay Buffer Size Allocation Methods

Buffer Size Allocation Method	Description
Percentage	A percentage of the total buffer.
Temporal	A period of time, value in microseconds. When you configure a temporal buffer, you must also configure a transmit rate. The system calculates the queue buffer size by multiplying the available bandwidth of the interface times the configured temporal value and transmit rate. When you specify a temporal method, the drop profile is assigned a static buffer and the system starts dropping packets once the queue buffer size is full. By default, the other buffer types are assigned dynamic buffers that use surplus transmission bandwidth to absorb bursts of traffic.

Table 213: Delay Buffer Size Allocation Methods *(continued)*

Buffer Size Allocation Method	Description
Remainder	The remaining buffer available. The remainder is the percentage buffer that is not assigned to other queues. For example, if you assign 40 percent of the delay buffer to queue 0, allow queue 3 to keep the default allotment of 5 percent, and assign the remainder to queue 7, then queue 7 uses approximately 55 percent of the delay buffer.

Specifying Delay Buffer Sizes for Queues

You specify delay buffer sizes for queues using schedulers. The system calculates the buffer size of a queue based on the buffer allocation method you specify for it in the scheduler. See Table 213 on page 648 for different buffer allocation methods and Table 214 on page 649 for buffer size calculations.

Table 214: Delay Buffer Allocation Method and Queue Buffer

Buffer Size Allocation Method	Queue Buffer Calculation	Example
Percentage	*available interface bandwidth* x *configured buffer size percentage* x *maximum delay buffer time* = *queue buffer*	Suppose you configure a queue on a 1xDS0 interface to use 30 percent of the available delay buffer size. The system uses the maximum available delay buffer time (4 seconds) and allocates the queue 9600 bytes of delay buffer: 64 Kbps x 0.3 x 4 seconds = 76800 bits = 9600 bytes
Temporal	*available interface bandwidth* x *configured transmit rate percentage* x *configured temporal buffer size* = *queue buffer*	Suppose you configure a queue on a 1xDS0 interface to use 300,000 microseconds (3 seconds) of delay buffer, and you configure the transmission rate to be 20 percent. The queue receives 4800 bytes of delay buffer: 64 Kbps x 0.2 x 3 seconds = 38400 bits = 4800 bytes When you configure a temporal value that is greater than the maximum available delay buffer time, the system allocates this queue the remaining buffer after other queues are allocated buffer. Suppose you configure a temporal value of 6,000,000 microseconds on a 1xDS0 interface. Because this value is greater than the maximum allowed value of 4,000,000 microseconds, the queue is allocated the remaining delay buffer.

When you specify the buffer size as a percentage, the system ignores the transmit rate and calculates the buffer size based only on the buffer size percentage.

Configuring a Large Delay Buffer on a Channelized T1 interface

On J-series Services Routers you can configure large delay buffers on channelized T1/E1 interfaces only. To configure large-delay buffer sizes, you must first enable the large buffer feature on the channelized T1/E1 PIM and then configure a buffer size for each queue in the CoS scheduler.

Each channelized T1/E1 interface can be configured as a single clear channel, or for channelized (*N*xDS0) operation, where *N* denotes channels 1 to 32 for an E1 interface and channels 1 to 24 for a T1 interface.

In this configuration, you enable the large delay buffer option on a channelized T1 PIM with an interface speed of 1.5 Mbps and a maximum delay buffer time of 500,000 microseconds. Based on the interface speed and the maximum delay buffer time, you can calculate the available delay buffer size for the interface. For more information, see "Maximum Delay Buffer Sizes Available to Interfaces" on page 647.

Next, you specify a queue buffer of 30 percent in a scheduler be-scheduler and associate the scheduler to a defined forwarding class be-class using a scheduler map large-buf-sched-map. Finally, you apply the scheduler map to the channelized T1 interface t1-3/0/0. As a result, a buffer of 9600 bytes is assigned to the queue associated with forwarding class be-class (see Table 214 on page 649). You can specify a delay buffer size for other queues following the instructions in this example.

To configure large delay buffers for channelized T1/E1 interfaces:

1. Navigate to the top of the configuration hierarchy in either the J-Web or CLI configuration editor.
2. Perform the configuration tasks described in Table 215 on page 650.
3. If you are finished configuring the router, commit the configuration.
4. Go on to one of the following tasks:
   - To configure other CoS components, see "Configuring CoS Components with a Configuration Editor" on page 599.
   - From the CLI, enter the show class of service command, to check your configuration.

Table 215: Configuring a Large Delay Buffer

Task	J-Web Configuration Editor	CLI Configuration Editor
Navigate to the **Chassis** level in the configuration hierarchy.	1. In the J-Web interface, select **Configuration > View and Edit > Edit Configuration**. 2. Next to Chassis, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit chassis

Table 215: Configuring a Large Delay Buffer *(continued)*

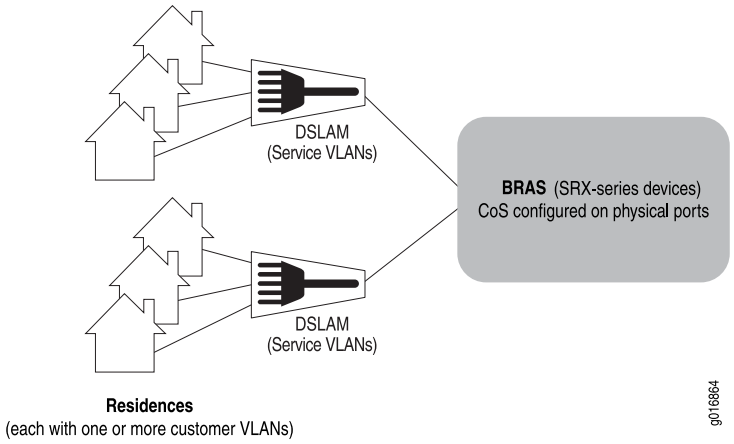
Task	J-Web Configuration Editor	CLI Configuration Editor
Enable the large buffer size feature on the channelized T1/E1 PIM in slot 3.	1. Next to Fpc, click **Add new entry**. 2. In the Slot box, type the slot number 3. 3. Next to Pic, click **Add new entry**. 4. In the Slot box, type 0. 5. Next to Q pic large buffer, select the check box. 6. Click **OK**.	Enter set fpc 3 pic 0 q-pic-large-buffer
Navigate to the **Class-of-service** level in the configuration hierarchy.	On the main Configuration page next to Class of service, click **Configure** or **Edit**.	From the [edit] hierarchy level, enter edit class-of-service
Create be-scheduler and specify a buffer size of 30 percent for it.	1. Next to Schedulers, click **Add new entry**. 2. In the Scheduler name box, type the name of the scheduler—be-scheduler. 3. Next to Buffer size, click **Configure**. 4. From the Buffer size choice list, select percent. 5. In the Percent box, type 30. 6. Click **OK**.	Enter set schedulers be-scheduler buffer-size percent 30
Configure the scheduler map large-buf-scheduler-map to associate schedulers with defined forwarding classes. For information about configuring forwarding classes, see "Assigning Forwarding Classes to Output Queues" on page 604.	1. On the Class of service page, next to Scheduler maps, click **Add new entry**. 2. In the Map name box, type the name of the scheduler map—large-buf-sched-map. 3. Next to Forwarding class, click **Add new entry**. 4. In the Class name box, type the name of the forwarding class to be associated with the scheduler—be-class. 5. In the Scheduler box, type the name of the scheduler to be associated with the forwarding class—be-scheduler. 6. Click **OK**.	From the [edit class-of-service] hierarchy level, enter set scheduler-maps large-buf-sched-map forwarding-class be-class scheduler be-scheduler
Apply the scheduler map to the channelized T1 interface. **NOTE:** For information about configuring channelized T1/E1 interfaces, see "Configuring Channelized T1/E1/ISDN PRI Interfaces" on page 109.	1. On the Class of service page, next to Interfaces, click **Add new entry**. 2. In the Interface name box, type the name of the interface to which the scheduler map is to be applied—t1-3/0/0. 3. Next to Unit, click **Add new entry**. 4. In the Unit number box, type 0. 5. In the Scheduler map box, type the name of the scheduler map—large-buf-sched-map. 6. Click **OK**.	From the [edit class-of-service] hierarchy level, type set interfaces t1-3/0/0 unit 0 scheduler-map large-buf-sched-map

Configuring CoS Hierarchical Schedulers

In metro Ethernet environments, a VLAN typically corresponds to a customer premises equipment (CPE) device and the VLANs are identified by an inner VLAN tag on Ethernet frames (called the customer VLAN, or C-VLAN, tag). A set of VLANs can be grouped at the DSL access multiplexer (DSLAM) and identified by using the same outer VLAN tag (called the service VLAN, or S-VLAN, tag). The service VLANs are typically gathered at the Broadband Remote Access Server (BRAS) level, which can be (among other devices) an SRX-series device. On SRX 5600 and SRX 5800 devices, hierarchical schedulers let you provide shaping and scheduling at the service VLAN level as well as other levels, such as the physical interface. In other words, you can group a set of logical interfaces and then apply scheduling and shaping parameters to the logical interface set as well as other levels.

This basic architecture is shown in Figure 95 on page 652. You can apply class-of-service (CoS) parameters at the premises on the CPE, on the customer or service VLANs, at the BRAS level, or at all levels.

Figure 95: An SRX-series Device in a Hierarchical Scheduler Architecture



On SRX 5600 and SRX 5800 devices, you can apply CoS shaping and scheduling at one of four different levels, including the VLAN set level.

The supported scheduler hierarchy is as follows:

- The physical interface (level 1)
- The service VLAN (level 2 is unique to SRX-series devices)
- The logical interface or customer VLAN (level 3)
- The queue (level 4)

You can specify a traffic control profile (output-traffic-control-profile) that can specify a shaping rate, a guaranteed rate, and a scheduler map with transmit rate and buffer delay. The scheduler map contains the mapping of queues (forwarding classes) to their respective schedulers (schedulers define the properties for the queue). Queue properties can specify a transmit rate and buffer management parameters such as

buffer size and drop profile. For more information, see "Defining Schedulers" on page 589.

To configure CoS hierarchical schedulers, include the following statements at the [edit class-of-service interfaces] and [edit interfaces] hierarchy levels:

```
[edit class-of-service interfaces]
interface-set interface-set-name {
    excess-bandwith-share (proportional value | equal);
    internal-node;
    output-traffic-control-profile profile-name;
    output-traffic-control-profile-remaining profile-name;
}

[edit interfaces]
hierarchical-scheduler;
interface-set interface-set-name {
    ethernet-interface-name {
        (interface-parameters);
    }
}
```

Hierarchical Scheduler Terminology

Hierarchical schedulers introduce some new terms into a discussion of CoS capabilities. They also use some familiar terms in different contexts. This section presents a complete overview of the terms used with hierarchical schedulers.

The following terms are important for hierarchical schedulers:

- Customer VLAN (C-VLAN)—A C-VLAN, defined by IEEE 802.1ad, . A stacked VLAN contains an outer tag corresponding to the S-VLAN, and an inner tag corresponding to the C-VLAN. A C-VLAN often corresponds to CPE. Scheduling and shaping is often used on a C-VLAN to establish minimum and maximum bandwidth limits for a customer. See also *S-VLAN*.
- Interface set—A logical group of interfaces that describe the characteristics of set of service VLANs, logical interfaces, or customer VLANs. Interface sets establish the set and name the traffic control profiles. See also *Service VLAN*.
- Scheduler— A scheduler defines the scheduling and queuing characteristics of a queue. Transmit rate, scheduler priority, and buffer size can be specified. In addition, a drop profile may be referenced to describe WRED congestion control aspects of the queue. See also *Scheduler map*.
- Scheduler map—A scheduler map is referenced by traffic control profiles to define queues. The scheduler map establishes the queues that comprise a scheduler node and associates a forwarding class with a scheduler. See also *Scheduler*.
- Stacked VLAN—An encapsulation on an S-VLAN with an outer tag corresponding to the S-VLAN, and an inner tag corresponding to the C-VLAN. See also *Service VLAN* and *Customer VLAN*.
- Service VLAN (S-VLAN)—An S-VLAN, defined by IEEE 802.1ad, often corresponds to a network aggregation device such as a DSLAM. Scheduling and shaping is

often established for an S-VLAN to provide CoS for downstream devices with little buffering and simple schedulers. See also *Customer VLAN*.

- Traffic control profile—Defines the characteristics of a scheduler node. Traffic control profiles are used at several levels of the CLI, including the physical interface, interface set, and logical interface levels. Scheduling and queuing characteristics can be defined for the scheduler node using the shaping-rate, guaranteed-rate, and delay-buffer-rate statements. Queues over these scheduler nodes are defined by referencing a scheduler map. See also *Scheduler* and *Scheduler map*.
- VLAN—Virtual LAN, defined on an Ethernet logical interface.

These terms are especially important when applied to a scheduler hierarchy. Scheduler hierarchies are composed of nodes and queues. Queues terminate the CLI hierarchy. Nodes can be either root nodes, leaf nodes, or internal (non-leaf) nodes. Internal nodes are nodes that have other nodes as "children" in the hierarchy. For example, if an interface-set statement is configured with a logical interface (such as unit 0) and queue, then the interface-set is an internal node at level 2 of the hierarchy. However, if there are no traffic control profiles configured on logical interfaces, then the interface set is at level 3 of the hierarchy.

Table 216 on page 654 shows how the configuration of an interface set or logical interface affects the terminology of hierarchical scheduler nodes.

Table 216: Hierarchical Scheduler Nodes

Root Node (Level 1)	Level 2	Level 3	Queue (Level 4)
Physical interface	Interface set	Logical interfaces	One or more queues
Physical interface		Interface set	One or more queues
Physical interface		Logical interfaces	One or more queues

SRX 3400 and SRX 3600 Hardware Capabilities and Limitations

The following list describes the hardware capabilities and limitations for the SRX 3400 and SRX 3600 series devices:

- For SRX 3400 and SRX 3600 series devices, each Input/Output Card (IOC) Flexible PIC Concentrator (FPC) or IOC slot has only one Physical Interface Card (PIC), which contains either two 10-Gigabit or sixteen 1-Gigabit Ethernet ports. Table 217 on page 655 shows the maximum number of cards and ports allowed in an SRX 3400 and SRX 3500 device.

Table 217: Available NPCs and IO Ports for SRX 3400 and SRX 3600 Devices

System	IOCs	IO Ports	NPCs
SRX 3600	7	108 (16 x 6 + 12)	3
SRX 3400	5	76 (16 x 4 + 12)	2

NOTE: The number of ports the Network Processing Unit (NPU) needs to handle may be different than the fixed 10:1 port to NPU ratio for 1G IOC, or the 1:1 ratio for the 10G IOC that is needed on the SRX 5600 and SRX 5800 devices, leading to oversubscription on the SRX 3400 and SRX 3600 devices.

- SRX 3400 and SRX 3600 series devices allow you to install up to three Network Processing Cards (NPC). In a single-NPC configuration, the NPC has to process all of the packets to and from each IOC. However, when there is more than one NPC available, an IOC will only exchange packets with a pre-assigned NPC. You can use the set chassis ioc-npc-connectivity CLI statement to configure the IOC-to-NPC mapping. By default, the mapping is assigned so that the load is shared equally among all NPCs. When the mapping is changed, for example, an IOC or NPC is removed, or you have mapped a specific NPC to an IOC, then the device has to be restarted. For more information, see the JUNOS Software Administration Guide.
- For SRX 3400 and SRX 3600 series devices, the IOC supports the following hierarchical scheduler characteristics:

  Level 1- Shaping at the physical interface (ifd)

  Level 2- Shaping and scheduling at the logical interface level (ifl)

  Level 3- Scheduling at the queue level

NOTE: Interface set (iflset) is not supported for the SRX 3400 and SRX 3600 devices.

- Shaping at the port level—In SRX 5600 and SRX 5800 devices, an NPC supports 32 port-level shaping profiles at level 1, such that each front port can have its own shaping profile.

  In SRX 3400 and SRX 3600 devices, an NPC supports only 16 port-level shaping profiles in the hardware, including two profiles that are predefined for 10-Gbps

and 1-Gbps shaping rates. The user can configure up to 14 different levels of shaping rates. If more levels are configured, then the closest match found in the 16 profiles will be used instead.

For example, assume that a system is already configured with the following rates for ifds:

10Mbps, 20Mbps, 40Mbps, 60Mbps, 80Mbps, 100Mbps, 200Mbps, 300Mbps, 400Mbps, 500Mbps, 600Mbps, 700Mbps, 800Mbps, 900Mbps, 1Gbps (predefined), 10Gbps (predefined)

Each of these 16 rates is programmed into one of the 16 profiles in the hardware, then consider the following two scenarios.

1. If the user changes one port's shaping rate from 1Gbps to 100Mbps, which is already programmed in one of the 16 profiles, the profile with 100Mbps shaping rate will be used by the port.

2. If the user changes another port's shaping rate from 1Gbps to 50Mbps, which is not in the shaping profiles, the closest matching profile with 60Mbps shaping rate will be used instead.

When scenario 2) happens, not all of the user-configured rates can be supported by the hardware. If more than 14 different rates are specified, only 14 will be programmed in the hardware. Which 14 rates are programmed in the hardware depends on many factors. For this reason, we recommend that you plan carefully and use no more than 14 levels of port-level shaping rates.

- Weighed Random Early Discard (WRED) at the port level—Each NPU has 512 MB of frame memory. Also, 10-Gigabit Ethernet ports get more buffers than the 1-Gigabit Ethernet ports. Buffer availability depends on how much bandwidth (number of NPCs, ports, 1-Gigabit or 10-Gigabit, and so forth) the device has to support. The more the bandwidth that the device has to support, the less buffer is available. When two NPCs are available, the amount of frame buffer available is doubled.

Configuring an Interface Set

To configure an interface set, include the following statement at the [edit class-of-service interfaces] hierarchy level of the configuration:

```
[edit class-of-service interfaces]
interface-set interface-set-name {
    (interface-cos-parameters);
}
```

To apply the interface set to interfaces, include the following statements at the [edit interfaces] hierarchy level of the configuration:

```
interface-set interface-set-name {
    ethernet-interface-name {
        (interface-cos-parameters);
    }
}
```

Interface sets can be defined as a list of logical interfaces (unit 100, unit 200, and so on). Service providers can use these statements to group interfaces to apply scheduling parameters such as guaranteed rate and shaping rate to the traffic in the groups.

All traffic heading downstream must be gathered into an interface set with the interface-set statement at the [edit class-of-service interfaces] hierarchy level.

Interface sets are currently only used by CoS, but they are applied at the [edit interfaces] hierarchy level so that they might be available to other services.

```
[edit interfaces]
interface-set interface-set-name {
   ethernet-interface-name {
      unit unit-number {
         ...
      }
   }
}
```

The logical interface naming option lists Ethernet interfaces:

```
[edit interfaces]
interface-set unitl-set-ge-0 {
   ge-0/0/0 {
      unit 0;
      unit 1;
      ...
   }
}
```

NOTE: Ranges are not supported; you must list each logical interface separately.

Applying an Interface Set

Although the interface set is applied at the [edit interfaces] hierarchy level, the CoS parameters for the interface set are defined at the [edit class-of-service interfaces] hierarchy level, usually with the output-traffic-control-profile *profile-name* statement.

This example applies a traffic control profile called tcp–set1 to an interface set called set-ge-0:

```
[edit interfaces]
interface-set set-ge-0 {
   output-traffic-control-profile tcp-set1;
}
```

Interface Set Caveats

You cannot specify an interface set mixing the logical interface, S-VLAN, or VLAN outer tag list forms of the interface-set statement.

A logical interface can only belong to one interface set. If you try to add the same logical interface to different interface sets, the commit will fail.

This example will generate a commit error:

```
[edit interfaces]
interface-set set-one {
    ge-2/0/0 {
        unit 0;
        unit 2;
    }
}
interface-set set-two {
    ge-2/0/0 {
        unit 1;
        unit 3;
        unit 0; # COMMIT ERROR! Unit 0 already belongs to -set-one.
    }
}
```

Members of an interface set cannot span multiple physical interfaces. Only one physical interface is allowed to appear in an interface set.

This configuration is not supported:

```
[edit interfaces]
interface-set set-group {
    ge-0/0/1 {
        unit 0;
        unit 1;
    }
    ge-0/0/2 { # This type of configuration is NOT supported in the same interface set!
        unit 0;
        unit 1;
    }
}
```

Introduction to Hierarchical Schedulers

When used, the interface set level of the hierarchy falls between the physical interface level (level 1) and the logical interface (level 3). Queues are always level 4 of the hierarchy.

Hierarchical schedulers add CoS parameters to the new interface set level of the configuration. They use traffic control profiles to set values for parameters such as shaping rate (the peak information rate [PIR]), guaranteed rate (the committed information rate [CIR] on these interfaces), scheduler maps (assigning queues and resources to traffic), and so on.

The following CoS configuration places the following parameters in traffic control profiles at various levels:

- Traffic control profile at the port level (tcp-port-level1):
  - A shaping rate (PIR) of 100 Mbps
  - A delay buffer rate of 100 Mbps
- Traffic control profile at the interface set level (tcp-interface-level2):

- A shaping rate (PIR) of 60 Mbps
- A guaranteed rate (CIR) of 40 Mbps

- Traffic control profile at the logical interface level (tcp-unit-level3):
  - A shaping rate (PIR) of 50 Mbps
  - A guaranteed rate (CIR) of 30 Mbps
  - A scheduler map called smap1 to hold various queue properties (level 4)
  - A delay buffer rate of 40 Mbps

For more information on scheduler maps, see "Defining and Applying Scheduler Maps" on page 259.

In this case, the traffic control profiles look like this:

```
[edit class-of-service traffic-control-profiles]
tcp-port-level1 { # This is the physical port level
    shaping-rate 100m;
    delay-buffer-rate 100m;
}
tcp-interface-level2 { # This is the interface set level
    shaping-rate 60m;
    guaranteed-rate 40m;
}
tcp-unit-level3 { # This is the logical interface level
    shaping-rate 50m;
    guaranteed-rate 30m;
    scheduler-map smap1;
    delay-buffer-rate 40m;
}
```

Once configured, the traffic control profiles must be applied to the proper places in the CoS interfaces hierarchy.

```
[edit class-of-service interfaces]
interface-set level-2 {
    output-traffic-control-profile tcp-interface-level-2;
}
ge-0/1/0 {
    output-traffic-control-profile tcp-port-level-1;
    unit 0 {
        output-traffic-control-profile tcp-unit-level-3;
    }
}
```

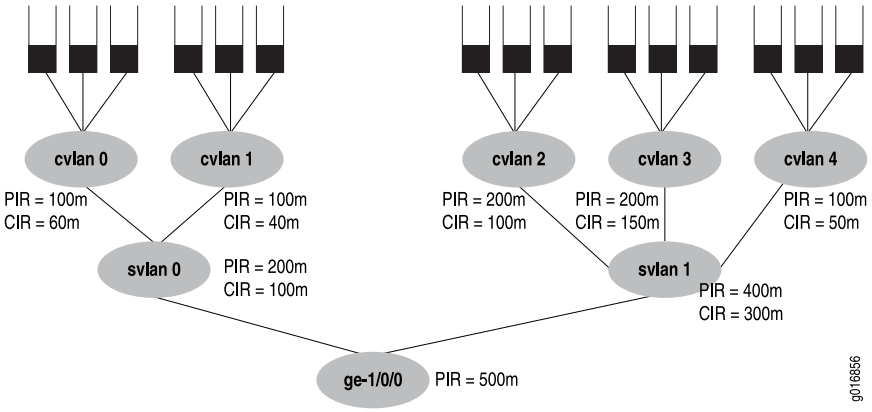
In all cases, the properties for level 4 of the hierarchical schedulers are determined by the scheduler map.

Scheduler Hierarchy Example

This section provides a more complete example of building a 4-level hierarchy of schedulers. The configuration parameters are shown in Figure 96 on page 660. The

queues are shown at the top of the figure with the other three levels of the hierarchy below.

Figure 96: Building a Scheduler Hierarchy



The figure's PIR values will be configured as the shaping rates, and the CIRs will be configured as the guaranteed rate on the Ethernet interface ge-1/0/0. The PIR can be oversubscribed (that is, the sum of the children PIRs can exceed the parent's, as in svlan 1, where 200 + 200 + 100 exceeds the parent rate of 400). However, the sum of the children node level's CIRs must never exceed the parent node's CIR, as shown in all the service VLANs (otherwise, the guaranteed rate could never be provided in all cases).

This configuration example will present all details of the CoS configuration for the interface in the figure (ge-1/0/0), including:

- Interface Sets for the Hierarchical Example on page 660
- Interfaces for the Hierarchical Example on page 661
- Traffic Control Profiles for the Hierarchical Example on page 661
- Schedulers for the Hierarchical Example on page 662
- Drop Profiles for the Hierarchical Example on page 663
- Scheduler Maps for the Hierarchical Example on page 663
- Applying Traffic Control Profiles for the Hierarchical Example on page 663

Interface Sets for the Hierarchical Example

```
[edit interfaces]
interface-set svlan-0 {
    interface ge-1/0/0 {
        unit 0;
        unit 1;
    }
}
interface-set svlan-1 {
    interface ge-1/0/0 {
        unit 2;
```

```
        unit 3;
        unit 4;
    }
}
```

Interfaces for the Hierarchical Example

The keyword to configure hierarchical schedulers is at the physical interface level, as are VLAN tagging and the VLAN IDs. In this example, the interface sets are defined by logical interfaces (units) and not outer VLAN tags. All VLAN tags in this example are customer VLAN tags.

```
[edit interface ge-1/0/0]
hierarchical-scheduler;
vlan-tagging;
unit 0 {
    vlan-id 100;
}
unit 1 {
    vlan-id 101;
}
unit 2 {
    vlan-id 102;
}
unit 3 {
    vlan-id 103;
}
unit 4 {
    vlan-id 104;
}
```

Traffic Control Profiles for the Hierarchical Example

The traffic control profiles hold parameters for levels above the queue level of the scheduler hierarchy. This section defines traffic control profiles for both the service VLAN level (logical interfaces) and the customer VLAN (VLAN tag) level.

```
[edit class-of-service traffic-control-profiles]
tcp-500m-shaping-rate {
    shaping-rate 500m;
}
tcp-svlan0 {
    shaping-rate 200m;
    guaranteed-rate 100m;
    delay-buffer-rate 300m; # This parameter is not shown in the figure
}
tcp-svlan1 {
    shaping-rate 400m;
    guaranteed-rate 300m;
    delay-buffer-rate 100m; # This parameter is not shown in the figure
}
tcp-cvlan0 {
    shaping-rate 100m;
    guaranteed-rate 60m;
    scheduler-map tcp-map-cvlan0; # This example applies scheduler maps to customer
        VLANs
```

```
}
tcp-cvlan1 {
    shaping-rate 100m;
    guaranteed-rate 40m;
    scheduler-map tcp-map-cvlan1; # This example applies scheduler maps to customer
        VLANs
}
tcp-cvlan2 {
    shaping-rate 200m;
    guaranteed-rate 100m;
    scheduler-map tcp-map-cvlanx; # This example applies scheduler maps to customer
        VLANs
}
tcp-cvlan3 {
    shaping-rate 200m;
    guaranteed-rate 150m;
    scheduler-map tcp-map-cvlanx; # This example applies scheduler maps to customer
        VLANs
}
tcp-cvlan4 {
    shaping-rate 100m;
    guaranteed-rate 50m;
    scheduler-map tcp-map-cvlanx; # This example applies scheduler maps to customer
        VLANs
}
```

Schedulers for the Hierarchical Example

The schedulers hold the information about the queues, the last level of the hierarchy. Note the consistent naming schemes applied to repetitive elements in all parts of this example.

```
[edit class-of-service schedulers]
sched-cvlan0-qx {
    priority low;
    transmit-rate 20m;
    buffer-size temporal 100ms;
    drop-profile-map loss-priority low dp-low;
    drop-profile-map loss-priority high dp-high;
}
sched-cvlan1-q0 {
    priority high;
    transmit-rate 20m;
    buffer-size percent 40;
    drop-profile-map loss-priority low dp-low;
    drop-profile-map loss-priority high dp-high;
}
sched-cvlanx-qx {
    transmit-rate percent 30;
    buffer-size percent 30;
    drop-profile-map loss-priority low dp-low;
    drop-profile-map loss-priority high dp-high;
}
sched-cvlan1-qx {
    transmit-rate 10m;
    buffer-size temporal 100ms;
```

```
    drop-profile-map loss-priority low dp-low;
    drop-profile-map loss-priority high dp-high;
}
```

Drop Profiles for the Hierarchical Example

This section configures the drop profiles for the example. For more information about drop profiles, see "Configuring RED Drop Profiles for Congestion Control" on page 620.

```
[edit class-of-service drop-profiles]
dp-low {
    interpolate fill-level 80 drop-probability 80;
    interpolate fill-level 100 drop-probability 100;
}
dp-high {
    interpolate fill-level 60 drop-probability 80;
    interpolate fill-level 80 drop-probability 100;
}
```

Scheduler Maps for the Hierarchical Example

This section configures the scheduler maps for the example. Each one references a scheduler configured in "Schedulers for the Hierarchical Example" on page 662.

```
[edit class-of-service scheduler-maps]
tcp-map-cvlan0 {
    forwarding-class voice scheduler sched-cvlan0-qx;
    forwarding-class video scheduler sched-cvlan0-qx;
    forwarding-class data scheduler sched-cvlan0-qx;
}
tcp-map-cvlan1 {
    forwarding-class voice scheduler sched-cvlan1-q0;
    forwarding-class video scheduler sched-cvlan1-qx;
    forwarding-class data scheduler sched-cvlan1-qx;
}
tcp-map-cvlanx {
    forwarding-class voice scheduler sched-cvlanx-qx;
    forwarding-class video scheduler sched-cvlanx-qx;
    forwarding-class data scheduler sched-cvlanx-qx;
}
```

Applying Traffic Control Profiles for the Hierarchical Example

This section applies the traffic control profiles to the proper levels of the hierarchy.

NOTE: Although a shaping rate can be applied directly to the physical interface, hierarchical schedulers must use a traffic control profile to hold this parameter, as shown in "Controlling Remaining Traffic" on page 664.

```
[edit class-of-service interfaces]
ge-1/0/0 {
    output-traffic-control-profile tcp-500m-shaping-rate;
    unit 0 {
```

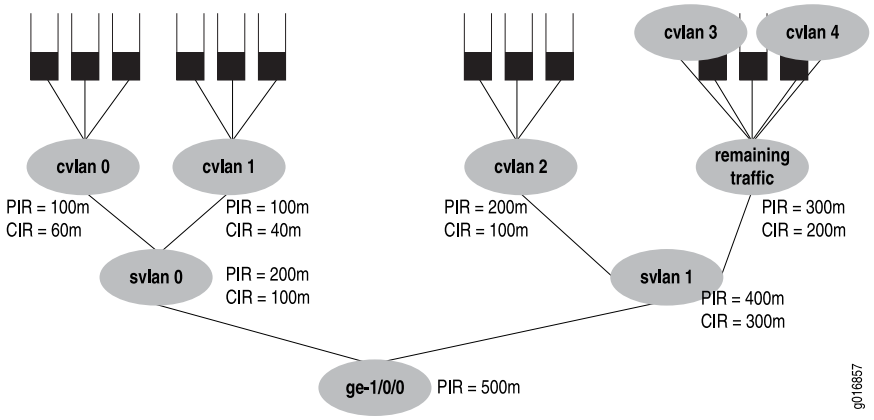
```
        output-traffic-control-profile tcp-cvlan0;
    }
    unit 1 {
        output-traffic-control-profile tcp-cvlan1;
    }
    unit 2 {
        output-traffic-control-profile tcp-cvlan2;
    }
    unit 3 {
        output-traffic-control-profile tcp-cvlan3;
    }
    unit 4 {
        output-traffic-control-profile tcp-cvlan4;
    }
}
interface-set svlan0 {
    output-traffic-control-profile tcp-svlan0;
}
interface-set svlan1 {
    output-traffic-control-profile tcp-svlan1;
}
```

Controlling Remaining Traffic

You can configure many logical interfaces under an interface. However, only a subset of them might have a traffic control profile attached. For example, you can configure three logical interfaces (units) over the same service VLAN, but you can apply a traffic control profile specifying best-effort and voice queues to only one of the logical interface units. Traffic from the two remaining logical interfaces is considered *remaining traffic*. To configure transmit rate guarantees for the remaining traffic, you configure the output-traffic-control-profile-remaining statement specifying a guaranteed rate for the remaining traffic. Without this statement, the remaining traffic gets a default, minimal bandwidth. In the same way, the shaping-rate and delay-buffer-rate statements can be specified in the traffic control profile referenced with the output-traffic-control-profile-remaining statement in order to shape and provide buffering for remaining traffic.

Consider the interface shown in Figure 97 on page 665. Customer VLANs 3 and 4 have no explicit traffic control profile. However, the service provider might want to establish a shaping and guaranteed transmit rate for aggregate traffic heading for those customer VLANs. The solution is to configure and apply a traffic control profile for all remaining traffic on the interface.

Figure 97: Handling Remaining Traffic



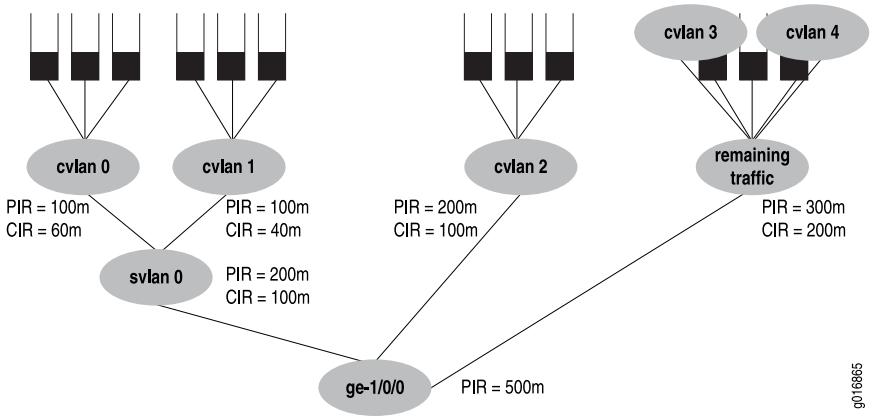
This example considers the case where customer VLANs 3 and 4 have no explicit traffic control profile, yet need to establish a shaping and guaranteed transmit rate for traffic heading for those customer VLANs. The solution is to add a traffic control profile to the svlan1 interface set. This example builds on the example used in "Scheduler Hierarchy Example" on page 659 and so this does not repeat all configuration details, only those at the service VLAN level.

```
[edit class-of-service interfaces]
interface-set svlan0 {
    output-traffic-control-profile tcp-svlan0;
}
interface-set svlan1 {
    output-traffic-control-profile tcp-svlan1;
    output-traffic-control-profile-remaining tcp-svlan1-remaining; # For all remaining traffic
}


[edit class-of-service traffic-control-profiles]
tcp-svlan1 {
    shaping-rate 400m;
    guaranteed-rate 300m;
}
tcp-svlan1-remaining {
    shaping-rate 300m;
    guaranteed-rate 200m;
    scheduler-map smap-remainder; # this smap is not shown in detail
}
```

Next, consider the example shown in Figure 98 on page 666.

Figure 98: Another Example of Handling Remaining Traffic



In this example, ge-1/0/0 has five logical interfaces (cvlan 0, 1, 2, 3 and 4), and svlan0, which are covered by the interface set:

- Scheduling for the interface set svlan0 is specified by referencing an output-traffic-control-profile statement, which specifies the guaranteed-rate, shaping-rate, and delay-buffer-rate statement values for the interface set. In this example, the output traffic control profile called tcp-svlan0 guarantees 100 Mbps and shapes the interface set svlan0 to 200 Mbps.
- Scheduling and queuing for remaining traffic of svlan0 is specified by referencing an output-traffic-control-profile-remaining statement, which references a scheduler-map statement that establishes queues for the remaining traffic. The specified traffic control profile can also configure guaranteed, shaping, and delay-buffer rates for the remaining traffic. In this example, output-traffic-control-profile-remaining tcp-svlan0-rem references scheduler-map smap-svlan0-rem, which calls for a best-effort queue for remaining traffic (that is, traffic on unit 3 and unit 4, which is not classified by the svlan0 interface set). The example also specifies a guaranteed-rate of 200 Mbps and a shaping-rate of 300 Mbps for all remaining traffic.
- Scheduling and queuing for logical interface ge-1/0/0 unit 1 is configured "traditionally" and uses an output-traffic-control-profile specified for that unit. In this example, output-traffic-control-profile tcp-ifl1 specifies scheduling and queuing for ge-1/0/0 unit 1.

This example does not include the [edit interfaces] configuration.

```
[edit class-of-service interfaces]
interface-set {
   svlan0 {
      output-traffic-control-profile tcp-svlan0; # Guarantee & shaper for svlan0
   }
}
ge-1/0/0 {
   output-traffic-control-profile-remaining tcp-svlan0-rem
      # Unit 3 and 4 are not explicitly configured, but captured by "remaining'
      unit 1 {
      output-traffic-control-profile tcp-ifl1; # Unit 1 be & ef queues
```

```
    }
}
```

Here is how the traffic control profiles for this example are configured:

```
[edit class-of-service traffic-control-profiles]
tcp-svlan0 {
    shaping-rate 200m;
    guaranteed-rate 100m;
}
tcp-svlan0-rem {
    shaping-rate 300m;
    guaranteed-rate 200m;
    scheduler-map smap-svlan0-rem; # This specifies queues for remaining traffic
}
tcp-ifl1 {
    scheduler-map smap-ifl1;
}
```

Finally, here are the scheduler maps and queues for the example:

```
[edit class-of-service scheduler-maps]
smap-svlan0-rem {
    forwarding-class best-effort scheduler sched-foo;
}
smap-ifl1 {
    forwarding-class best-effort scheduler sched-bar;
    forwarding-class assured-forwarding scheduler sched-baz;
}
```

The configuration for the referenced schedulers is not given for this example.

Internal Scheduler Nodes

A node in the hierarchy is considered internal if either of the following conditions apply:

- Any one of its children nodes has a traffic control profile configured and applied.
- You configure the internal-node statement.

Why would it be important to make a certain node internal? Generally, there are more resources available at the logical interface (unit) level than at the interface set level. Also, it might be desirable to configure all resources at a single level, rather than spread over several levels. The internal-node statement provides this flexibility. This can be a helpful configuration device when interface-set queuing without logical interfaces is used exclusively on the interface.

The internal-node statement can be used to raise the interface set without children to the same level as the other configured interface sets with children, allowing them to compete for the same set of resources.

In summary, using the internal-node statement allows statements to all be scheduled at the same level with or without children.

The following example makes the interfaces sets if-set-1 and if-set-2 internal:

```
[edit class-of-service interfaces ]
interface-set {
    if-set-1 {
        internal-node;
        output-traffic-control-profile tcp-200m-no-smap;
    }
    if-set-2 {
        internal-node;
        output-traffic-control-profile tcp-100m-no-smap;
    }
}
```

If an interface set has logical interfaces configured with a traffic control profile, then the use of the internal-node statement has no effect.

Internal nodes can specify a traffic-control-profile-remaining statement.

PIR-only and CIR Mode

The actual behavior of many CoS parameters, especially the shaping rate and guaranteed rate, depend on whether the physical interface is operating in PIR-only (peak information rate) or CIR (committed information rate) mode.

In PIR-only mode, one or more nodes perform shaping. The physical interface is in the PIR-only mode if no child (or grandchild) node under the port has a guaranteed rate configured.

The mode of the port is important because in PIR-only mode, the scheduling across the child nodes is in proportion to their shaping rates (PIRs) and not the guaranteed rates (CIRs). This can be important if the observed behavior is not what is anticipated.

In CIR mode, one or more nodes applies a guaranteed rate and might perform shaping. A physical interface is in CIR mode if at least one child (or grandchild) node has a guaranteed rate configured. In addition, any child or grandchild node under the physical interface can have a shaping rate configured.

Only the guaranteed rate matters. In CIR mode, nodes that do not have a guaranteed rate configured are assumed to have a very small guaranteed rate (queuing weight).

Priority Propagation

SRX 5600 and SRX 5800 devices with input/output cards (IOCs) perform priority propagation. Priority propagation is useful for mixed traffic environments when, for example, you want to make sure that the voice traffic of one customer does not suffer due to the data traffic of another customer. Nodes and queues are always serviced in the order of their priority. The priority of a queue is decided by configuration (the default priority is low) in the scheduler. However, not all elements of hierarchical schedulers have direct priorities configured. Internal nodes, for example, must determine their priority in other ways.

The priority of any internal node is decided by:

- The highest priority of an active child (interface sets only take the highest priority of their active children
- Whether the node is above its configured guaranteed rate (CIR) or not (this is relevant only if the physical interface is in CIR mode)

Each queue will have a configured priority and a hardware priority. The usual mapping between the configured priority and the hardware priority as shown in Table 218 on page 669.

Table 218: Queue Priority

Configured Priority	Hardware Priority
Strict-high	0
High	0
Medium-high	1
Medium-low	1
Low	2

In CIR mode, the priority for each internal node depends on whether the highest active child node is above or below the guaranteed rate. The mapping between the highest active child's priority and the hardware priority below and above the guaranteed rate is shown in Table 219 on page 669.

Table 219: Internal Node Queue Priority for CIR Mode

Configured Priority of Highest Active Child Node	Hardware Priority Below Guaranteed Rate	Hardware Priority Above Guaranteed Rate
Strict-high	0	0
High	0	3
Medium-high	1	3
Medium-low	1	3
Low	2	3

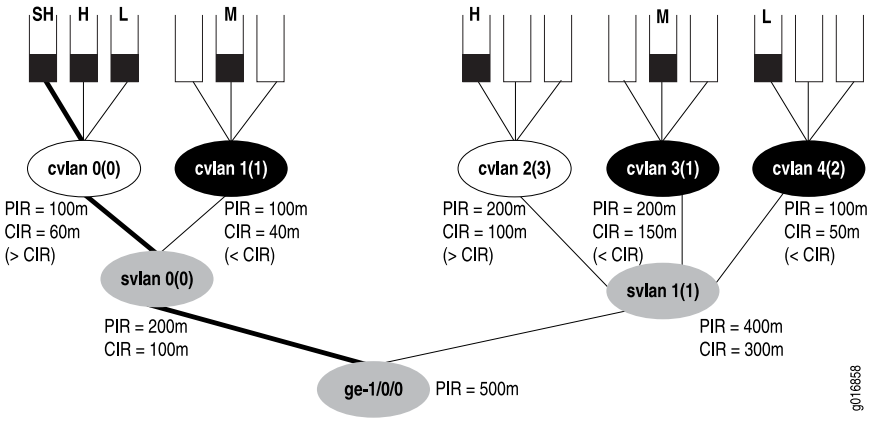
In PIR-only mode, nodes cannot send if they are above the configured shaping rate. The mapping between the configured priority and the hardware priority is for PIR-only mode is shown in Table 220 on page 670.

Table 220: Internal Node Queue Priority for PIR-Only Mode

Configured Priority	Hardware Priority
Strict-high	0
High	0
Medium-high	1
Medium-low	1
Low	2

A physical interface with hierarchical schedulers configured is shown in Figure 99 on page 670. The configured priorities are shown for each queue at the top of the figure. The hardware priorities for each node are shown in parentheses. Each node also shows any configured shaping rate (PIR) or guaranteed rate (CIR) and whether or not the queues are above or below the CIR. The nodes are shown in one of three states: above the CIR (clear), below the CIR (dark), or in a condition where the CIR does not matter (gray).

Figure 99: Hierarchical Schedulers and Priorities



In the figure, the strict high queue for customer VLAN 0 (cvlan 0) receives service first, even though the customer VLAN is above the configured CIR (see Table 219 on page 669 for the reason: strict-high always has hardware priority 0 regardless of CIR state). Once that queue has been drained, and the priority of the node has become 3 instead of 0 (due to the lack of strict-high traffic), the system moves on to the medium queues next (cvlan 1 and cvlan 3), draining them in a round robin fashion (empty queues lose their hardware priority). The low queue on cvlan 4 (priority 2) will be sent next, because that mode is below the CIR. Then the high queues on cvlan 0 and cvlan2 (both now with priority 3) are drained in a round-robin fashion, and finally the low queue on cvlan 0 is drained (because svlan 0 has a priority of 3).

IOC Hardware Properties

On SRX 5600 and SRX 5800 devices, two IOCs (40x1GE IOC and 4x10GE IOC) are supported on which you can configure schedulers and queues. You can configure 15 VLAN sets per Gigabit Ethernet (40x1GE IOC) port and 255 VLAN sets per 10 Gigabit Ethernet (4x10GE IOC) port. The IOC performs priority propagation from one hierarchy level to another, and drop statistics are available on the IOC per color per queue instead of just per queue.

SRX 5600 and SRX 5800 devices with IOCs have Packet Forwarding Engines that can support up to 512 MB of frame memory, and packets are stored in 512–byte frames. Table 221 on page 671 compares the major properties of the the Packet Forwarding Engine within the IOC.

Table 221: Forwarding Engine Properties within 40x1GE IOC and 4x10GE IOC

Feature	PFE Within 40x1GE IOC and 4x10GE IOC
Number of usable queues	16,000
Number of shaped logical interfaces	2,000 with 8 queues each, or 4,000 with 4 queues each.
Number of hardware priorities	4
Priority propagation	Yes
Dynamic mapping	Yes: schedulers/port are not fixed.
Drop statistics	Per queue per color (PLP high, low)

Additionally, the IOC features also support hierarchical weighted random early detection (WRED).

The IOC supports the following hierarchical scheduler characteristics:

- Shaping at the physical interface level
- Shaping and scheduling at the service VLAN interface set level
- Shaping and scheduling at the customer VLAN logical interface level
- Scheduling at the queue level

The IOC supports the following features for scalability:

- 16,000 queues per PFE
- 4 Packet Forwarding Engines per IOC
  - 4000 schedulers at logical interface level (level 3) with 4 queues each
  - 2000 schedulers at logical interface level (level 3) with 8 queues each
- 255 schedulers at the interface set level (level 2) per 1-port PFE on a 10-Gigabit Ethernet IOC (4x10GE IOC)

- 15 schedulers at the interface set level (level 2) per 10-port PFE on a 1-Gigabit Ethernet IOC (40x1GE IOC)
- About 400 milliseconds of buffer delay (this varies by packet size and if large buffers are enabled)
- 4 levels of priority (strict-high, high, medium, and low)

NOTE: The exact option for a transmit-rate (transmit-rate rate exact) is not supported on the IOCs on SRX-series devices.

The manner in which the IOC maps a queue to a scheduler depends on whether 8 queues or 4 queues are configured. By default, a scheduler at level 3 has 4 queues. Level 3 scheduler X controls queue X*4 to X*4 + 3, so that scheduler 100 (for example) controls queues 400 to 403. However, when 8 queues per scheduler are enabled, the odd-numbered schedulers are disabled, allowing twice the number of queues per subscriber as before. With 8 queues, level 3 scheduler X controls queue X*4 to X*4 + 7, so that scheduler 100 (for example) now controls queues 400 to 407.

You configure the max-queues-per-interface statement to set the number of queues at 4 or 8 at the FPC level of the hierarchy. Changing this statement will result in a restart of the FPC. For more information about the max-queues-per-interface statement, see “Example: Configuring Up to Eight Forwarding Classes” on page 607 and the *JUNOS Software CLI Reference*.

The IOC maps level 3 (customer VLAN) schedulers in groups to level 2 (service VLAN) schedulers. Sixteen contiguous level 3 schedulers are mapped to level 2 when 4 queues are enabled, and 8 contiguous level 3 schedulers are mapped to level 2 when 8 queues are enabled. All the schedulers in the group should use the same queue priority mapping. For example, if the queue priorities of one scheduler are high, medium, low, and low, all members of the group should have the same queue priority.

Groups at level 3 to level 2 can be mapped at any time. However, a group at level 3 can only be unmapped from a level 2 scheduler, and only if all the schedulers in the group are free. Once unmapped, a level 3 group can be remapped to any level 2 scheduler. There is no restriction on the number of level 3 groups that can be mapped to a particular level 2 scheduler. There can be 256 level 3 groups, but fragmentation of the scheduler space can reduce the number of schedulers available. In other words, there are scheduler allocation patterns that might fail even though there are free schedulers.

In contrast to level 3 to level 2 mapping, the IOC maps level 2 (service VLAN) schedulers in a fixed mode to level 1 (physical interface) schedulers. On 40-port Gigabit Ethernet IOCs, there are 16 level 1 schedulers, and 10 of these are used for the physical interfaces. There are 256 level 2 schedulers, or 16 per level 1 scheduler. A level 1 scheduler uses level schedulers X*16 through X*16 + 15. Therefore level 1 scheduler 0 uses level 2 schedulers 0 through 15, level 1 scheduler 1 uses level 2 schedulers 16 through 31, and so on. On 4-port 10 Gigabit Ethernet PICs, there is one level 1 scheduler for the physical interface, and 256 level 2 schedulers are mapped to the single level 1 scheduler.

The maximum number of level 3 (customer VLAN) schedulers that can be used is 4076 (4 queues) or 2028 (8 queues) for the 10-port Gigabit Ethernet Packet Forwarding Engine and 4094 (4 queues) or 2046 (8 queues) for the 10 Gigabit Ethernet Packet Forwarding Engine.

WRED on the IOC

Shaping to drop out-of-profile traffic is done on the IOC at all levels except the queue level. However, weighed random early discard (WRED) is done at the queue level with much the same result. With WRED, the decision to drop or send the packet is made before the packet is placed in the queue.

WRED shaping on the IOC involves two levels. The probabilistic drop region establishes a minimum and a maximum queue depth. Below the minimum queue depth, the drop probability is 0 (send). Above the maximum level, the drop probability is 100 (certainty).

There are four drop profiles associated with each queue. These correspond to each of four loss priorities (low, medium-low, medium-high, and high). Sixty-four sets of four drop profiles are available (32 for ingress and 32 for egress). In addition, there are eight WRED scaling profiles in each direction.

An IOC drop profile for expedited forwarding traffic might look like this:

```
[edit class-of-service drop-profiles]
drop-ef {
   fill-level 20 drop-probability 0; # Minimum Q depth
   fill-level 100 drop-probability 100; # Maximum Q depth
}
```

Note that only two fill levels can be specified for the IOC. You can configure the **interpolate** statement, but only two fill levels are used. The **delay-buffer-rate** statement in the traffic control profile determines the maximum queue size. This delay buffer rate is converted to a packet delay buffers, where one buffer is equal to 512 bytes. For example, at 10 Mbps, the IOC will allocate 610 delay buffers when the delay buffer rate is set to 250 milliseconds. The WRED threshold values are specified in terms of absolute buffer values.

The WRED scaling factor multiples all WRED thresholds (both minimum and maximum) by the value specified. There are eight values in all: 1, 2, 4, 8, 16, 32, 64, and 128. The WRED scaling factor is chosen to best match the user-configured drop profiles. This is done because the hardware supports only certain values of thresholds (all values must be a multiple of 16). So if the configured value of a threshold is 500 (for example), the multiple of 16 is 256 and the scaling factor applied is 2, making the value 512, which allows the value of 500 to be used. If the configured value of a threshold is 1500, the multiple of 16 is 752 and the scaling factor applied is 2, making the value 1504, which allows the value of 1500 to be used.

Hierarchical RED is used to support the oversubscription of the delay buffers (WRED is configured only at the queue, physical interface, and PIC level). Hierarchical RED works with WRED as follows:

- If any level accepts the packet (the queue depth is less than the minimum buffer level), this level accepts the packet.

- If any level probabilistically drops the packet, then this level drops the packet.

However, these rules might lead to the accepting of packets under loaded conditions that might otherwise have been dropped. In other words, the logical interface will accept packets if the physical interface is not congested.

Due to the limits placed on shaping thresholds used in the hierarchy, there is a granularity associated with the IOCs. The shaper accuracies differ at various levels of the hierarchy, with shapers at the logical interface level (level 3) being more accurate than shapers at the interface set level (level 2) or the port level (level 1). Table 222 on page 674 shows the accuracy of the logical interface shaper at various speeds for Ethernet ports operating at 1 Gbps.

Table 222: Shaper Accuracy of 1-Gbps Ethernet at the Logical Interface Level

Range of Logical Interface Shaper	Step Granularity
Up to 4.096 Mbps	16 Kbps
4.096 to 8.192 Mbps	32 Kbps
8.192 to 16.384 Mbps	64 Kbps
16.384 to 32.768 Mbps	128 Kbps
32.768 to 65.535 Mbps	256 Kbps
65.535 to 131.072 Mbps	512 Kbps
131.072 to 262.144 Mbps	1024 Kbps
262.144 to 1 Gbps	4096 Kbps

Table 223 on page 674 shows the accuracy of the logical interface shaper at various speeds for Ethernet ports operating at 10 Gbps.

Table 223: Shaper Accuracy of 10-Gbps Ethernet at the Logical Interface Level

Range of Logical Interface Shaper	Step Granularity
Up to 10.24 Mbps	40 Kbps
10.24 to 20.48 Mbps	80 Kbps
10.48 to 40.96 Mbps	160 Kbps
40.96 to 81.92 Mbps	320 Kbps
81.92 to 163.84 Mbps	640 Kbps
163.84 to 327.68 Mbps	1280 Kbps
327.68 to 655.36 Mbps	2560 Kbps

Table 223: Shaper Accuracy of 10-Gbps Ethernet at the Logical Interface Level *(continued)*

Range of Logical Interface Shaper	Step Granularity
655.36 to 2611.2 Mbps	10240 Kbps
2611.2 to 5222.4 Mbps	20480 Kbps
5222.4 to 10 Gbps	40960 Kbps

Table 224 on page 675 shows the accuracy of the interface set shaper at various speeds for Ethernet ports operating at 1 Gbps.

Table 224: Shaper Accuracy of 1-Gbps Ethernet at the Interface Set Level

Range of Interface Set Shaper	Step Granularity
Up to 20.48 Mbps	80 Kbps
20.48 Mbps to 81.92 Mbps	320 Kbps
81.92 Mbps to 327.68 Mbps	1.28 Mbps
327.68 Mbps to 1 Gbps	20.48 Mbps

Table 225 on page 675 shows the accuracy of the interface set shaper at various speeds for Ethernet ports operating at 10 Gbps.

Table 225: Shaper Accuracy of 10-Gbps Ethernet at the Interface Set Level

Range of Interface Set Shaper	Step Granularity
Up to 128 Mbps	500 Kbps
128 Mbps to 512 Mbps	2 Mbps
512 Mbps to 2.048 Gbps	8 Mbps
2.048 Gbps to 10 Gbps	128 Mbps

Table 226 on page 675 shows the accuracy of the physical port shaper at various speeds for Ethernet ports operating at 1 Gbps.

Table 226: Shaper Accuracy of 1-Gbps Ethernet at the Physical Port Level

Range of Physical Port Shaper	Step Granularity
Up to 64 Mbps	250 Kbps

Table 226: Shaper Accuracy of 1-Gbps Ethernet at the Physical Port Level *(continued)*

Range of Physical Port Shaper	Step Granularity
64 Mbps to 256 Mbps	1 Mbps
256 Mbps to 1 Gbps	4 Mbps

Table 227 on page 676 shows the accuracy of the physical port shaper at various speeds for Ethernet ports operating at 10 Gbps.

Table 227: Shaper Accuracy of 10-Gbps Ethernet at the Physical Port Level

Range of Physical Port Shaper	Step Granularity
Up to 640 Mbps	2.5 Mbps
640 Mbps to 2.56 Gbps	10 Mbps
2.56 Gbps to 10 Gbps	40 Mbps

For more information about configuring RED drop profiles, see “Configuring RED Drop Profiles for Congestion Control” on page 620.

MDRR on the IOC

The guaranteed rate (CIR) at the interface set level is implemented by using modified deficit round-robin (MDRR). The IOC hardware provides four levels of strict priority. There is no restriction on the number of queues for each priority. MDRR is used among queues of the same priority. Each queue has one priority when it is under the guaranteed rate and another priority when it is over the guaranteed rate but still under the shaping rate (PIR). The IOC hardware implements the priorities with 256 service profiles. Each service profile assigns eight priorities for eight queues. One set is for logical interfaces under the guaranteed rate and another set is for logical interfaces over the guaranteed rate but under the shaping rate. Each service profile is associated with a group of 16 level 3 schedulers, so there is a unique service profile available for all 256 groups at level 3, giving 4,096 logical interfaces.

JUNOS software provides three priorities for traffic under the guaranteed rate and one reserved priority for traffic over the guaranteed rate that is not configurable. JUNOS software provides three priorities when there is no guaranteed rate configured on any logical interface.

The relationship between JUNOS software priorities and the IOC hardware priorities below and above the guaranteed rate (CIR) is shown in Table 228 on page 677.

Table 228: JUNOS Priorities Mapped to IOC Hardware Priorities

JUNOS Software Priority	IOC Hardware Priority Below Guaranteed Rate	IOC Hardware Priority Above Guaranteed Rate
Strict-high	High	High
High	High	Low
Medium-high	Medium-high	Low
Medium-low	Medium-high	Low
Low	Medium-low	Low

The JUNOS software parameters are set in the scheduler map:

```
[edit class-of-service schedulers]
best-effort-scheduler {
   transmit-rate percent 30; # if no shaping rate
   buffer-size percent 30;
   priority high;
}
expedited-forwarding-scheduler {
   transmit-rate percent 40; # if no shaping rate
   buffer-size percent 40;
   priority strict-high;
}
```

NOTE: The use of both shaping rate and a guaranteed rate at the interface set level (level 2) is not supported.

MDRR is provided at three levels of the scheduler hierarchy of the IOC with a granularity of 1 through 255. There are 64 MDRR profiles at the queue level, 16 at the interface set level, and 32 at the physical interface level.

Queue transmit rates are used for queue-level MDRR profile weight calculation. The queue MDRR weight is calculated differently based on the mode set for sharing excess bandwidth. If you configure the equal option for excess bandwidth, then the queue MDRR weight is calculated as:

Queue weight = (255 * Transmit-rate-percentage) / 100

If you configure the proportional option for excess bandwidth, which is the default, then the queue MDRR weight is calculated as:

Queue weight = Queue-transmit-rate / Queue-base-rate, where

Queue-transmit-rate = (Logical-interface-rate * Transmit-rate-percentage) / 100, and

Queue-base-rate = Excess-bandwidth-propotional-rate / 255

To configure the way that the IOC should handle excess bandwidth, configure the excess-bandwith-share statement at the [edit interface-set *interface-set-name*] hierarchy level. By default, the excess bandwidth is set to proportional with a default value of 32.64 Mbps. In this mode, the excess bandwidth is shared in the ratio of the logical interface shaping rates. If set to equal, the excess bandwidth is shared equally among the logical interfaces.

This example sets the excess bandwidth sharing to proportional at a rate of 100 Mbps with a shaping rate of 80 Mbps.

```
[edit interface-set example-interface-set]
excess-bandwidth-share proportional 100m;
output-traffic-control-profile PIR-80Mbps;
```

Shaping rates established at the logical interface level are used to calculate the MDRR weights used at the interface set level. The 16 MDRR profiles are set to initial values, and the closest profile with rounded values is chosen. By default, the physical port MDRR weights are preset to the full bandwidth on the interface.

Configuring Excess Bandwidth Sharing

When using the IOC (40x1GE IOC or 4x10GE IOC) on an SRX-series device, there are circumstances when you should configure excess bandwidth sharing and minimum logical interface shaping. This section details some of the guidelines for configuring excess bandwidth sharing.

Excess Bandwidth Sharing and Minimum Logical Interface Shaping

The default excess bandwidth sharing proportional rate is 32.65 Mbps (128 Kbps x 255). In order to have better weighed fair queuing (WFQ) accuracy among queues, the shaping rate configured should be larger than the excess bandwidth sharing proportional rate. Some examples are shown in Table 229 on page 678.

Table 229: Shaping Rates and WFQ Weights

Shaping Rate	Configured Queue Transmit Rate	WFQ Weight	Total Weights
10 Mbps	(30, 40, 25, 5)	(22, 30, 20, 4)	76
33 Mbps	(30, 40, 25, 5)	(76, 104, 64, 13)	257
40 Mbps	(30, 40, 25, 5)	(76, 104.64, 13)	257

With a 10-Mbps shaping rate, the total weights are 76. This is divided among the four queues according to the configured transmit rate. Note that when the shaping rate is larger than the excess bandwidth sharing proportional rate of 32.65 Mbps, the total weight on the logical interface is 257 and the WFQ accuracy will be the same.

Selecting Excess Bandwidth Sharing Proportional Rates

To determine a good excess bandwidth-sharing proportional rate to configure, choose the largest CIR (guaranteed rate) among all the logical interfaces (units). If the logical units have PIRs (shaping rates) only, then choose the largest PIR rate. However, this is not ideal if a single logical interface has a large WRR rate. This method can skew the distribution of traffic across the queues of the other logical interfaces. To avoid this issue, set the excess bandwidth-sharing proportional rate to a lower value on the logical interfaces where the WRR rates are concentrated. This improves the bandwidth sharing accuracy among the queues on the same logical interface. However, the excess bandwidth sharing for the logical interface with the larger WRR rate is no longer proportional.

As an example, consider five logical interfaces on the same physical port, each with four queues, all with only PIRs configured and no CIRs. The WRR rate is the same as the PIR for the logical interface. The excess bandwidth is shared proportionally with a rate of 40 Mbps. The traffic control profiles for the logical interfaces are shown in Table 230 on page 679.

Table 230: Example Shaping Rates and WFQ Weights

Shaping Rate	Configured Queue Transmit Rate	WFQ Weight	Total Weights
(Unit 0) 10 Mbps	(95, 0, 0, 5)	(60, 0, 0, 3)	63
(Unit 1) 20 Mbps	(25, 25, 25, 25)	(32, 32, 32, 32)	128
(Unit 2) 40 Mbps	(40, 30, 20, 10)	(102, 77, 51, 26)	255
(Unit 3) 200 Mbps	(70, 10, 10, 10)	(179, 26, 26, 26)	255
(Unit 4) 2 Mbps	(25, 25, 25, 25)	(5, 5, 5, 5)	20

Even though the maximum transmit rate for the queue on logical interface unit 3 is 200 Mbps, the excess bandwidth-sharing proportional rate is kept at a much lower value. Within a logical interface, this method provides a more accurate distribution of weights across queues. However, the excess bandwidth is now shared equally between unit 2 and unit 3 (total weights = 255).

Mapping Calculated Weights to Hardware Weights

The calculated weight in a traffic control profile is mapped to hardware weight, but the hardware only supports a limited WFQ profile. The weights are rounded to the nearest hardware weight according to the values in Table 231 on page 680.

Table 231: Rounding Configured Weights to Hardware Weights

Traffic Control Profile Number	Number of Traffic Control Profiles	Weights	Maximum Error
1–16	16	1–16 (interval of 1)	50.00%
17–29	13	18–42 (interval of 2)	6.25%
30–35	6	45–60 (interval of 3)	1.35%
36–43	8	64–92 (interval of 4)	2.25%
44–49	6	98–128 (interval of 6)	3.06%
50–56	7	136–184 (interval of 8)	3.13%
57–62	6	194–244 (interval of 10)	2.71%
63–63	1	255–255 (interval of 11)	2.05%

From the table, as an example, the calculated weight of 18.9 is mapped to a hardware weight of 18, because 18 is closer to 18.9 than 20 (an interval of 2 applies in the range 18–42).

Allocating Weight with Only Shaping Rates or Unshaped Logical Interfaces

Logical interfaces with only shaping rates (PIRs) or unshaped logical interfaces (units) are given a weight of 10. A logical interface with a small guaranteed rate (CIR) might get an overall weight less than 10. In order to allocate a higher share of the excess bandwidth to logical interfaces with a small guaranteed rate in comparison to the logical interfaces with only shaping rates configured, a minimum weight of 20 is given to the logical interfaces with guaranteed rates configured.

For example, consider a logical interface configuration with five units, as shown in Table 232 on page 680.

Table 232: Allocating Weights with PIR and CIR on Logical Interfaces

Logical Interface (Unit)	Traffic Control Profile	WRR Percentages	Weights
Unit 1	PIR 100 Mbps	95, 0, 0, 5	10, 1, 1, 1
Unit 2	CIR 20 Mbps	25, 25, 25, 25	64, 64, 64, 64
Unit 3	PIR 40 Mbps, CIR 20 Mbps	50, 30, 15, 5	128, 76, 38, 13
Unit 4	Unshaped	95, 0, 0, 5	10, 1, 1, 1
Unit 5	CIR 1 Mbps	95, 0, 0, 5	10, 1, 1, 1

The weights for these units are calculated as follows:

- Select the excess bandwidth-sharing proportional rate to be the maximum CIR among all the logical interfaces: 20 Mbps (unit 2).
- Unit 1 has a PIR and unit 4 is unshaped. The weight for these units is 10.
- The weight for unit 1 queue 0 is 9.5 (10 x 95%), which translates to a hardware weight of 10.
- The weight for unit 1 queue 1 is 0 (0 x 0%), but although the weight is zero, a weight of 1 is assigned to give minimal bandwidth to queues with zero WRR.
- Unit 5 has a very small CIR (1 Mbps), and a weight of 20 is assigned to units with a small CIR.
- The weight for unit 5 queue 0 is 19 (20 x 95%), which translates to a hardware weight of 18.
- Unit 3 has a CIR of 20 Mbps, which is the same as the excess bandwidth-sharing proportional rate, so it has a total weight of 255.
- The weight of unit 3 queue 0 is 127.5 (255 x 50%), which translates to a hardware weight of 128.

Sharing Bandwidth Among Logical Interfaces

As a simple example showing how bandwidth is shared among the logical interfaces, assume that all traffic is sent on queue 0. Assume also that there is a 40-Mbps load on all of the logical interfaces. Configuration details are shown in Table 233 on page 681.

Table 233: Sharing Bandwidth Among Logical Interfaces

Logical Interface (Unit)	Traffic Control Profile	WRR Percentages	Weights
Unit 1	PIR 100 Mbps	95, 0, 0, 5	10, 1, 1, 1
Unit 2	CIR 20 Mbps	25, 25, 25, 25	64, 64, 64, 64
Unit 3	PIR 40 Mbps, CIR 20 Mbps	50, 30, 15, 5	128, 76, 38, 13
Unit 4	Unshaped	95, 0, 0, 5	10, 1, 1, 1

1. When the port is shaped at 40 Mbps, because units 2 and 3 have a guaranteed rate (CIR) configured, both units 2 and 3 get 20 Mbps of shared bandwidth.
2. When the port is shaped at 100 Mbps, because units 2 and 3 have a guaranteed rate (CIR) configured, each of them can transmit 20 Mbps. On units 1, 2, 3, and 4, the 60 Mbps of excess bandwidth is shaped according to the values shown in Table 234 on page 682.

Table 234: First Example of Bandwidth Sharing

Logical Interface (Unit)	Calculation	Bandwidth
1	10 / (10 + 64 + 128 + 10) x 60 Mbps	2.83 Mbps
2	64 / (10 + 64 + 128 + 10) x 60 Mbps	18.11 Mbps
3	128 / (10 + 64 + 128 + 10) x 60 Mbps	36.22 Mbps
4	10 (10 + 64 + 128 + 10) x 60 Mbps	2.83 Mbps

However, unit 3 only has 20 Mbps extra (PIR and CIR) configured. This means that the leftover bandwidth of 16.22 Mbps (36.22 Mbps – 20 Mbps) is shared among units 1, and 2, and 4. This is shown in Table 235 on page 682.

Table 235: Second Example of Bandwidth Sharing

Logical Interface (Unit)	Calculation	Bandwidth
1	10 / (10 + 64 + 128 + 10) x 16.22 Mbps	1.93 Mbps
2	64 / (10 + 64 + 128 + 10) x 16.22 Mbps	12.36 Mbps
4	10 (10 + 64 + 128 + 10) x 16.22 Mbps	1.93 Mbps

Finally, Table 236 on page 682 shows the resulting allocation of bandwidth among the logical interfaces when the port is configured with a 100-Mbps shaping rate.

Table 236: Final Example of Bandwidth Sharing

Logical Interface (Unit)	Calculation	Bandwidth
1	2.83 Mbps + 1.93 Mbps	4.76 Mbps
2	20 Mbps + 18.11 Mbps + 12.36 Mbps	50.47 Mbps
3	20 Mbps + 20 Mbps	40 Mbps
4	2.83 Mbps + 1.93 Mbps	4.76 Mbps

Verifying a CoS Configuration

To verify a CoS configuration on a Services Router, perform the tasks relevant to your CoS configuration from the following:

- Verifying Multicast Session Announcements on page 683
- Verifying a Virtual Channel Configuration on page 683

Verifying Multicast Session Announcements

Purpose Verify that the Services Router is listening to the appropriate groups for multicast Session Announcement Protocol (SAP) session announcements.

Action From the CLI, enter the show sap listen command.

Sample Output

```
user@host> show sap listen
Group Address   Port
224.2.127.254   9875
```

Meaning The output shows a list of the group addresses and ports that SAP and SDP listen on. Verify the following information:

- Each group address configured, especially the default 224.2.127.254, is listed.
- Each port configured, especially the default 9875, is listed.

Related Topics For a complete description of the show sap listen command and output, see the *JUNOS Routing Protocols and Policies Command Reference*.

Verifying a Virtual Channel Configuration

Purpose Verify the virtual channel configuration on a logical interface. Verify the class-of-service (CoS) configuration associated with an interface on a Services Router.

Action From the CLI, enter the show class-of-service virtual-channel command.

Sample Output

```
user@host> show class-of-service virtual-channel
Virtual channel: vc-1 Index: 1
```

Meaning Verify that the name of the configured virtual channel is displayed in the output.

Related Topics For a complete description of the show class-of-service virtual-channel command and output, see the *JUNOS System Basics and Services Command Reference*.

Verifying a Virtual Channel Group Configuration

Purpose Verify the virtual channel group configuration on a logical interface. Verify the class-of-service (CoS) configuration associated with an interface on a Services Router.

Action From the CLI, enter the show class-of-service virtual-channel-group command.

Sample Output

```
user@host> show class-of-service virtual-channel-group
Virtual channel group: vc-group, Index: 16321        Virtual channel: vc-1
   Scheduler map: sc-map
```

Meaning Verify that the name of the configured virtual channel group is displayed in the output.

Related Topics For a complete description of the show class-of-service virtual-channel-group command and output, see the *JUNOS System Basics and Services Command Reference*.

Verifying an Adaptive Shaper Configuration

Purpose Verify the adaptive shaper trigger point and its associated transmit rate. Verify the class-of-service (CoS) configuration associated with an interface on a Services Router.

Action From the CLI, enter the show class-of-service adaptive-shaper and show class-of-service interface t1-0/0/2 commands.

Sample Output

```
user@host> show class-of-service adaptive-shaper
Adaptive shaper: fr-shaper, Index: 35320
  Trigger type    Shaping rate
        BECN         64000 bps

user@host> show class-of-service interface t1-0/0/2
Physical interface: t1-0/0/2, Index: 137
Queues supported: 8, Queues in use: 4
  Scheduler map: <default>, Index: 2

  Logical interface: t1-0/0/2.0, Index: 69
    Object            Name                   Type                    Index
    Adaptive-shaper   fr-shaper                                      35320
    Classifier        ipprec-compatibility   ip                         11
```

Meaning Verify the following information:

- The trigger type and shaping rate are consistent with the configured adaptive shaper.
- The adaptive shaper applied to the logical interface is displayed under Name.

Related Topics For a complete description of the show class-of-service adaptive-shaper and show class-of-service interface commands and output, see the *JUNOS System Basics and Services Command Reference*.

Displaying CoS Tunnel Configurations

Purpose Verify the configuration of the CoS tunnel queuing on a Services Router. You can analyze the flow of traffic by displaying the entire configuration. The following output is specific to CoS queuing configuration.

Action From the CLI on Router A and B, enter the following show commands.

Router B

```
user@host# show interfaces gr-0/0/0
per-unit-scheduler
   unit 0 {
      tunnel
      source 70.0.0.1;
      destination 70.0.0.2;
      family inet {
         address 10.80.0.1/24;
      }
```

```
    }
  }

  user@host#show class-of-service interfaces gr-0/0/0
  unit 0 {
    scheduler-map SMAP;
    shaping-rate 200m;
  }
```

Router A

```
  user@host# show chassis
  fpc 0 {
    pic 0 {
      tunnel-queuing
    }
  }
  [edit]
```

Verifying a CoS GRE Tunnel Queuing Configuration

Purpose Verify that the Services Router is configured properly for tunnel configuration.

Action From the CLI, enter the show interfaces queue gr-0/0/0.0 command.

NOTE: If you enter gr-0/0/0 only, queue information for all tunnels is displayed. If you enter gr-0/0/0 unit *logical_unit_number* queue information for the specific tunnel is displayed.

Sample Output

```
user@host> show interfaces queue gr-0/0/0.0
Logical interface gr-0/0/0.0 (Index 68) (SNMP ifIndex 112)
Forwarding classes: 8 supported, 4 in use
Egress queues: 8 supported, 4 in use Burst size: 0
Queue: 0, Forwarding classes: VOICE
  Queued:
    Packets              :              7117734                 7998 pps
    Bytes                :            512476848              4606848 bps
  Transmitted:
    Packets              :              4548146                 3459 pps
    Bytes                :            327466512              1992912 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :              2569421                 4537 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :              2569421                 4537 pps
    RED-dropped bytes    :            184998312              2613640 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :            184998312              2613640 bps
Queue: 1, Forwarding classes: GOLD
  Queued:
    Packets              :               117600                    0 pps
    Bytes                :              8467200                    0 bps
  Transmitted:
    Packets              :               102435                    0 pps
```

```
    Bytes               :              7375320                0 bps
    Tail-dropped packets :                    0                0 pps
    RED-dropped packets  :                15165                0 pps
     Low                 :                    0                0 pps
     Medium-low          :                    0                0 pps
     Medium-high         :                    0                0 pps
     High                :                15165                0 pps
    RED-dropped bytes    :              1091880                0 bps
     Low                 :                    0                0 bps
     Medium-low          :                    0                0 bps
     Medium-high         :                    0                0 bps
     High                :              1091880                0 bps
Queue: 2, Forwarding classes: SILVER
  Queued:
    Packets              :                    0                0 pps
    Bytes                :                    0                0 bps
  Transmitted:
    Packets              :                    0                0 pps
    Bytes                :                    0                0 bps
    Tail-dropped packets :                    0                0 pps
    RED-dropped packets  :                    0                0 pps
     Low                 :                    0                0 pps
     Medium-low          :                    0                0 pps
     Medium-high         :                    0                0 pps
     High                :                    0                0 pps
    RED-dropped bytes    :                    0                0 bps
     Low                 :                    0                0 bps
     Medium-low          :                    0                0 bps
     Medium-high         :                    0                0 bps
     High                :                    0                0 bps
Queue: 3, Forwarding classes: BRONZE
  Queued:
    Packets              :                    0                0 pps
    Bytes                :                    0                0 bps
  Transmitted:
    Packets              :                    0                0 pps
    Bytes                :                    0                0 bps
    Tail-dropped packets :                    0                0 pps
    RED-dropped packets  :                    0                0 pps
     Low                 :                    0                0 pps
     Medium-low          :                    0                0 pps
     Medium-high         :                    0                0 pps
     High                :                    0                0 pps
    RED-dropped bytes    :                    0                0 bps
     Low                 :                    0                0 bps
     Medium-low          :                    0                0 bps
     Medium-high         :                    0                0 bps
     High                :                    0                0 bps
```

Meaning The output lists the egress queues and the corresponding forwarding classes. Assuming that traffic is correctly classified using appropriate classifiers on the ingress port, verify the traffic flow in various queues.

Related Topics For a complete description of the show interfaces queue command and output, see the *JUNOS Interfaces Command Reference*.

Verifying a CoS IP-IP Tunnel Configuration

Purpose Verify that the Services Router is configured properly for tunnel configuration.

Action From the CLI, enter the show interfaces queue ip-0/0/0.0 command.

NOTE: If you enter ip-0/0/0 only, queue information for all tunnels is displayed. If you enter ip-0/0/0 unit *logical_unit_number* queue information for the specific tunnel is displayed.

Sample Output

```
user@host> show interfaces queue ip-0/0/0.0
Logical interface ip-0/0/0.0 (Index 70) (SNMP ifIndex 56
Forwarding classes: 8 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: q0
Queued:
  Packets: 38802710 22687 pps
  Bytes: 10942364220 51183344 bps
```

Meaning The output lists the egress queues and the corresponding forwarding classes. Assuming that traffic is correctly classified using appropriate classifiers on the ingress port, verify the traffic flow in various queues.

Related Topics For a complete description of the show interfaces queue command and output, see the *JUNOS Interfaces Command Reference*.

Part 7
Index

Index

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X