



**JUNOS™ Internet Software
for M-series and T-series Routing Platforms**

Baseline Operations Guide

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

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986–1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by The Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, The Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

Juniper Networks, the Juniper Networks logo, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. JUNOS and JUNOSE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

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

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

Copyright © 2006, Juniper Networks, Inc.
All rights reserved. Printed in USA.

JUNOS Internet Software for M-series and T-series Routing Platforms Baseline Network Operations Guide
Writer: Merisha Wazna
Editor: Sonia Saruba
Covers and template design: Edmonds Design

Thanks to Robert Kim for his outstanding contribution to this book, and Siew Ng for his help.

Revision History
31 March 2004—Revision 1.
10 April 2006—Revision 2.
27 September 2006—Revision 3.

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 Juniper Networks, Inc. and its subsidiaries (collectively "Juniper"), and 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, and updates and releases of such software, for which Customer has paid the applicable license or support fees to Juniper or an authorized Juniper reseller. "Embedded Software" means Software which Juniper has embedded in the Juniper 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 the 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 software on multiple computers requires multiple licenses, regardless of whether such computers 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 the Embedded Software on non-Juniper equipment; (j) use the 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 NON-INFRINGEMENT. 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 for the Software are exclusive of taxes, withholdings, duties, or levies (collectively "Taxes"). Customer shall be responsible for paying Taxes arising from the purchase of the license, or importation or use of the Software.

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 présentes confirment leur volonté que cette convention de même que tous les documents y compris tout avis qui s'y rattache, soient rédigés 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

	About This Guide	xi
Part 1	Juniper Networks Hardware and Software	
Chapter 1	Juniper Networks Router Overview	3
Chapter 2	Cheat Sheet for the CLI Commands	17
Chapter 3	Work with Problems on Your Network	29
Part 2	JUNOS Internet Software in the Network	
Chapter 4	Stop and Start JUNOS Software	37
Chapter 5	Display JUNOS Software Information	45
Chapter 6	Check Router Configuration	49
Chapter 7	Upgrade JUNOS Software	57
Chapter 8	Reinstall JUNOS Software	73
Part 3	Verify Your Network Topology	
Chapter 9	Verify Juniper Networks Routers	91
Chapter 10	Verify Physical Interfaces on the Router	101
Chapter 11	Verify the IS-IS Protocol and Adjacencies	111

Chapter 12	Verify the OSPF Protocol and Neighbors	129
Chapter 13	Verify the BGP Protocol and Peers	157
Chapter 14	Verify the Routing Engine CPU Memory	179
Chapter 15	Verify Traffic and Packets through the Router	189
Chapter 16	Use the ping and traceroute Commands	199
Chapter 17	Use MIBs	205
 Part 4	 Gather System Management Information	
Chapter 18	Display Basic Chassis Information	227
Chapter 19	Display and Locate Files and Directories	231
Chapter 20	Check Time on a Router	239
Chapter 21	Check User Accounts and Permissions	247
 Part 5	 Search Log Messages	
Chapter 22	Track Normal Operations	261
Chapter 23	Track Error Conditions	273
Chapter 24	Collect Crash Data	299
 Part 6	 Appendix	
Appendix 1	Command-Line Interface Overview	321
 Part 7	 Index	
	Index	343

Table of Contents

About This Guide	xi
Objectives	xi
Audience	xii
Chapter Organization	xii
Using the Examples in This Manual	xiii
Documentation Conventions	xv
Related Juniper Networks Documentation	xvi
Documentation Feedback	xix
How to Request Support	xix

Part 1

Juniper Networks Hardware and Software

Chapter 1	Juniper Networks Router Overview	3
	Router Architecture	4
	Router Architecture for M-series Routers and T-series Platforms	4
	Data Flow through the Packet Forwarding Engine	5
	Data Flow through an M-series Router	6
	Data Flow through a T-series Routing Platform	7
	Hardware Components	9
	Chassis	9
	Flexible PIC Concentrators	10
	Physical Interface Cards	10
	Routing Engine	12
	Power Supplies	12
	Cooling System	13
	Monitor Hardware Components	14
	Contact JTAC	15
Chapter 2	Cheat Sheet for the CLI Commands	17
	CLI Operational Mode Top-Level Commands	18
	CLI Configuration Mode Top-Level Commands	20
	Load a Configuration Using Copy and Paste Commands	22
	Load a Configuration from a File To a Router	22
	Load a Configuration Using the display set Command	24
	CLI Keyboard Shortcuts	26
	Manage Output at the ---(more)--- Prompt	27

Chapter 3	Work with Problems on Your Network	29
	Isolate a Broken Network Connection	30
	Identify the Symptoms	31
	Isolate the Causes	32
	Take Appropriate Action	33
	Evaluate the Solution	34
 Part 2	 JUNOS Internet Software in the Network	
 Chapter 4	 Stop and Start JUNOS Software	 37
	Stop the JUNOS Software	38
	Reboot the JUNOS Software	39
	Restart a JUNOS Software Process	40
	Display Information about Software Processes	40
	Restart a JUNOS Software Process	41
	Check That the Process Has Restarted	42
 Chapter 5	 Display JUNOS Software Information	 45
	Display JUNOS Software Information	46
	Display Version Information for JUNOS Software Packages	47
 Chapter 6	 Check Router Configuration	 49
	Display the Current Active Router Configuration	50
	Display a Specific Configuration Hierarchy	54
	Display Additional Information about the Configuration	54
 Chapter 7	 Upgrade JUNOS Software	 57
	Before You Upgrade JUNOS Software	58
	Log the Software Version Information	59
	Log the Hardware Version Information	60
	Log the Active Configuration	61
	Log the Interfaces on the Router	61
	Log the BGP, IS-IS, and OSPF Adjacency Information	62
	Log the System Storage Information	63
	Back Up the Currently Running and Active File System	63
	Download JUNOS Software	64
	Upgrade JUNOS Software	69
	Copy JUNOS Software to the Router	69
	Add New Software	69
	Start the New Software	70
	After You Upgrade JUNOS Internet Software	71
	Compare Information Logged Before and After the Upgrade	71
	Back Up the New Software	71
 Chapter 8	 Reinstall JUNOS Software	 73
	Before You Reinstall JUNOS Software	75
	Log the Software Version Information	76
	Log the Hardware Version Information	77

Log the Chassis Environment Information	78
Log the System Boot-Message Information	78
Log the Active Configuration	80
Log the Interfaces on the Router	80
Log the BGP, IS-IS, and OSPF Adjacency Information	81
Log the System Storage Information	82
Back Up the Currently Running and Active File System	82
Have the Boot Floppy or PCMCIA Card Ready	83
Reinstall the JUNOS Software	83
Reconfigure the JUNOS Software	83
Configure Names and Addresses	84
Set the Root Password	85
Check Network Connectivity	85
Copy Backup Configurations to the Router	86
After You Reinstall JUNOS Software	87
Compare Information Logged Before and After the Reinstall	87
Back Up the New Software	87

Part 3

Verify Your Network Topology

Chapter 9	Verify Juniper Networks Routers	91
	Check Router Components	92
	Check the Router Component Status	93
	Check the Router Craft Interface	93
	Check the Component LEDs	94
	Display Detailed Component Environmental Information	95
	Display Detailed Operational Information About Components	96
	Gather Component Alarm Information	97
	Display the Current Router Alarms	97
	Display Error Messages in the Messages Log File	97
	Display Error Messages in the Chassis Process Log File	98
	Verify the Component Problem	98
	Fix the Problem	99
	Contact JTAC	99
	Return the Failed Component	100
Chapter 10	Verify Physical Interfaces on the Router	101
	Check Physical Interfaces on a Router	102
	Display Summary Interface Information	102
	Display Detailed Interface Information	103
	Display Real-Time Statistics about a Physical Interface	107
	Check System Logging	109
Chapter 11	Verify the IS-IS Protocol and Adjacencies	111
	Verify the IS-IS Configuration on a Router in a Network	112
	Check the Configuration of a Level 1/Level 2 Router	114
	Check the Configuration of a Level 1 Router	116
	Check the Configuration of a Level 2 Router	117
	Display the Status of IS-IS Adjacencies	119
	Verify Adjacent Routers	120

	Examine a Route	121
	Examine the Forwarding Table	123
	Examine the Link-State Database	124
	Examine a Link-State Protocol Data Unit Header	126
Chapter 12	Verify the OSPF Protocol and Neighbors	129
	Verify the OSPF Protocol	130
	Check OSPF on an ASBR	132
	Check OSPF on an ABR	135
	Check OSPF on a Stub Router	139
	Check OSPF Neighbors	141
	Verify OSPF Neighbors	142
	Examine the OSPF Link-State Database	144
	Examine OSPF Routes	148
	Examine the Forwarding Table	151
	Examine Link-State Advertisements in Detail	152
Chapter 13	Verify the BGP Protocol and Peers	157
	Verify the BGP Protocol	158
	Verify BGP on an Internal Router	159
	Verify BGP on a Border Router	162
	Verify BGP Peers	165
	Check That BGP Sessions Are Up	166
	Verify That a Neighbor is Advertising a Particular Route	169
	Verify That a Particular BGP Route Is Received on Your Router	170
	Examine BGP Routes and Route Selection	171
	Examine the Local Preference Selection	173
	Examine the Multiple Exit Discriminator Route Selection	174
	Examine the EBGP over IBGP Selection	175
	Examine the IGP Cost Selection	176
	Examine Routes in the Forwarding Table	177
Chapter 14	Verify the Routing Engine CPU Memory	179
	Check the Routing CPU Memory Usage	180
	Check Overall CPU and Memory Usage	180
	Check Routing Protocol Process (rpd) Memory Usage	183
	Display Tasks	185
Chapter 15	Verify Traffic and Packets through the Router	189
	Monitor Traffic through the Router	190
	Display Real-Time Statistics about All Interfaces on the Router	190
	Display Real-Time Statistics about an Interface	191
	Verify Packets	193
	Monitor Packets Sent from and Received by the Routing Engine	193
	Display Key IP Header Information	194
	Show Packet Count When a Firewall Filter Is Configured with the Count Option	195
	Display Traffic from the Point of View of the Packet Forwarding Engine	196

Chapter 16	Use the ping and traceroute Commands	199
	Check the Accessibility of Two Routers on the Edge	200
	Use Loopback Addresses	201
	Use Interface Addresses	202
	Examples of Unsuccessful ping and traceroute Commands	203
Chapter 17	Use MIBs	205
	Determine Which MIBs Are Supported by a Juniper Release	206
	Run Snmpwalk from an NMS System to a Juniper Router	207
	Use SNMP Trace Operations to Monitor a Router	209
	Configure Trace Operations for SNMP	209
	Query a MIB With SNMPGet	210
	Display the Output for SNMP Trace Operations	211
	Monitor Memory Usage on a Router	212
	Check Memory Utilization on Chassis Components	212
	Check Memory Utilization per Process	215
	Monitor CPU Utilization	218
	Check CPU Utilization	218
	Check CPU Utilization per Process	220
	Retrieve Version Information about Router Software Components	223
Part 4	Gather System Management Information	
Chapter 18	Display Basic Chassis Information	227
	Display Basic Chassis Information	228
Chapter 19	Display and Locate Files and Directories	231
	Copy a File on a Routing Engine	232
	Copy a File from One Routing Engine to Another	232
	Copy Files between the Local Router and a Remote System	232
	Maintain a Single Configuration File for Both Routing Engines	234
	Configure the New Group	234
	Apply the New Group	236
	List Files and Directories on a Router	237
	Rename a File on a Router	238
Chapter 20	Check Time on a Router	239
	Check the Time on a Router	240
	Check How Long Router Components Have Been Up	240
	Check the NTP Peers	243
	Check the NTP Status	244
Chapter 21	Check User Accounts and Permissions	247
	Understand User Accounts and Permissions	248
	Check for Users in Configuration Mode	249
	Check the Commands That Users Are Entering	250
	Configure the Log File for Tracking CLI Commands	250
	Display the Configured Log File	251

Log a User Out of the Router	252
Check When the Last Configuration Change Occurred	253
Configure Configuration Change Tracking	253
Display the Configured Log File	254
Force a Message to Logged-In User Terminals	255
Check RADIUS Server Connectivity	256

Part 5

Search Log Messages

Chapter 22	Track Normal Operations	261
	Configure System Logging	263
	Log Messages to a Local Log File	263
	Log Information to a Remote Host	265
	Log Information to a User Terminal	266
	Log Information to a Router Console	267
	Configure the Number and Size of Log Files	267
	Log BGP State Transition Events	268
	Display a Log File	270
	Monitor Messages in Near-Real Time	271
	Stop Monitoring Log Files	271
Chapter 23	Track Error Conditions	273
	Configure Routing Protocol Process Tracing	275
	Configure Routing Protocol Process Tracing	275
	Configure Routing Protocol Tracing for a Specific Routing Protocol	278
	Monitor Trace File Messages Written in Near-Real Time	279
	Stop Trace File Monitoring	280
	Configure BGP-Specific Options	281
	Display Detailed BGP Protocol Information	281
	Display Sent or Received BGP Packets	283
	Diagnose BGP Session Establishment Problems	284
	Configure IS-IS-Specific Options	286
	Display Detailed IS-IS Protocol Information	286
	Display Sent or Received IS-IS Protocol Packets	289
	Analyze IS-IS-Link State Packets in Detail	291
	Configure OSPF-Specific Options	293
	Diagnose OSPF Session Establishment Problems	293
	Analyze OSPF Link-State Advertisement Packets in Detail	297
Chapter 24	Collect Crash Data	299
	Understand Crash Data Collection	301
	Collect Crash Data for a Routing Engine Kernel	302
	Check the Routing Engine Core Files	302
	Collect and Send Crash Data to JTAC	303
	Collect Crash Data for Routing Engine Daemons	306
	Check for Daemon Core Files	306
	Collect and Send Crash Data to JTAC	307
	Collect Crash Data for the Packet Forwarding Engine Microkernel	310
	Display the Crash Stack Traceback and Registration Information	311
	Clear the NVRAM Contents	314

Check Packet Forwarding Engine Microkernel Core Files	315
Collect and Send Crash Data to JTAC	315

Part 6

Appendix

Appendix 1	Command-Line Interface Overview	321
	CLI Operational Mode	322
	Use the CLI Operational Mode.....	323
	Enter the CLI Operational Mode	323
	Get Help on Commands at a Hierarchy Level	323
	Get Help about Commands.....	324
	Have the CLI Complete Commands	325
	Use CLI Command Completion.....	326
	Display CLI Command History.....	326
	CLI Configuration Mode	327
	Configuration Statements and Identifiers.....	329
	Configuration Statement Hierarchy	331
	Use the CLI Configuration Mode	332

Part 7

Index

Index	343
--------------	------------

About This Guide

This preface provides the following guidelines for using the *JUNOS Internet Software for M-series and T-series Routing Platforms Baseline Network Operations Guide* and related Juniper Networks, Inc., technical documents:

- Objectives on page xi
- Audience on page xii
- Chapter Organization on page xii
- Using the Examples in This Manual on page xiii
- Documentation Conventions on page xv
- Related Juniper Networks Documentation on page xvi
- Documentation Feedback on page xix
- How to Request Support on page xix

Objectives

This guide provides operational information helpful for the most basic tasks associated with running a network using Juniper Networks products. This guide is not directly related to any particular release of the JUNOS Internet software.

To obtain the most current version of this manual, refer to the product documentation page on the Juniper Networks Web site, which is located at <http://www.juniper.net/>.

To order printed copies of this guide or to order a documentation CD-ROM, which contains this guide, please contact your sales representative.

Audience

This guide is designed for Network Operations Center (NOC) personnel who monitor a Juniper Networks M-series or T-series routing platform.

To use this guide, you need a broad understanding of networks in general, the Internet in particular, networking principles, and network configuration. You must also be familiar with one or more of the following Internet routing protocols:

- Border Gateway Protocol (BGP)
- Distance Vector Multicast Routing Protocol (DVMRP)
- Intermediate System-to-Intermediate System (IS-IS)
- Internet Control Message Protocol (ICMP) router discovery
- Internet Group Management Protocol (IGMP)
- Multiprotocol Label Switching (MPLS)
- Open Shortest Path First (OSPF)
- Protocol-Independent Multicast (PIM)
- Resource Reservation Protocol (RSVP)
- Routing Information Protocol (RIP)
- Simple Network Management Protocol (SNMP)

Personnel operating the equipment must be trained and competent; must not conduct themselves in a careless, willfully negligent, or hostile manner; and must abide by the instructions provided by the documentation.

Chapter Organization

Most chapters in this guide consist of a checklist at the beginning of the chapter listing the tasks and commands for monitoring the basic functions of a router and a network. Tasks include displaying various types of information about a router and the network, upgrading and reinstalling JUNOS software, tracing users and accounts, and collecting crash data. The tasks and commands are then explained in step-by-step procedures.

Each step-by-step procedure consists of some or all of the following parts:

- Purpose—Describes what is affected if this task is not performed or what is accomplished with this task.
- What Is... —Describes a component (usually hardware).
- Step(s) To Take—Lists the steps in the task.
- Action—Describes an action to perform in order to complete the step.

- Sample Output—Presents sample output relevant to the procedure.
- What It Means—Describes or summarizes what is presented in the sample output.
- Symptom/Indications—Describes a problem with the software or hardware.
- See Also—Lists other topics related to this task.
- Alternative Actions—Describes other commands or ways of doing the task.
- Syntax—Describes the full syntax of the command or configuration statement. For an explanation of how to read the syntax statements, see “Documentation Conventions” on page xv.

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

```
system {
  scripts {
    commit {
      file ex-script.xsl;
    }
  }
}
```

```

interfaces {
  fxp0 {
    disable;
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
    }
  }
}

```

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

```

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

```

Merging a Snippet

To merge a snippet, follow these steps:

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

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

```

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

```

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

```

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

```

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

```

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

```

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

Documentation Conventions

Table 1 defines notice icons used in this guide.

Table 1: Notice Icons



Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.

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

Table 2: Text and Syntax Conventions (1 of 2)

Convention	Element	Example
Bold sans serif typeface	Represents text that you type.	To enter configuration mode, type the <code>configure</code> command: user@host> configure
Fixed-width typeface	Represents output on the terminal screen.	user@host> show chassis alarms No alarms currently active
<i>Italic typeface</i>	<ul style="list-style-type: none"> Introduces important new terms. Identifies book names. Identifies RFC and Internet draft titles. 	<ul style="list-style-type: none"> A policy <i>term</i> is a named structure that defines match conditions and actions. <i>JUNOS System Basics Configuration Guide</i> RFC 1997, <i>BGP Communities Attribute</i>
<i>Italic sans serif typeface</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i>
Sans serif typeface	Represents names of configuration statements, commands, files, and directories; IP addresses; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none"> To configure a stub area, include the stub statement at the [edit protocols ospf area <i>area-id</i>] hierarchy level. The console port is labeled CONSOLE.
< > (angle brackets)	Enclose optional keywords or variables.	stub <default-metric <i>metric</i> >;
(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 (<i>string1</i> <i>string2</i> <i>string3</i>)
# (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 [<i>community-ids</i>]

Table 2: Text and Syntax Conventions (2 of 2)

Convention	Element	Example
Indentation and braces ({ })	Identify a level in the configuration hierarchy.	<pre>[edit] routing-options { static { route default { nexthop <i>address</i>; retain; } } }</pre>
; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
J-Web GUI Conventions		
Bold typeface	Represents J-Web graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none">■ In the Logical Interfaces box, select All Interfaces.■ To cancel the configuration, click Cancel.
> (bold right angle bracket)	Separates levels in a hierarchy of J-Web selections.	In the configuration editor hierarchy, select Protocols > Ospf .

Related Juniper Networks Documentation

Table 3 lists the books included in the *Network Operations Guide* series.

Table 3: JUNOS Internet Software Network Operations Guides

Book	Description
JUNOS Internet Software for M-series and T-series Routing Platforms Network Operations Guides	
<i>Baseline</i>	Describes the most basic tasks for running a network using Juniper Networks products. Tasks include upgrading and reinstalling JUNOS software, gathering basic system management information, verifying your network topology, and searching log messages.
<i>Interfaces</i>	Describes tasks for monitoring interfaces. Tasks include using loopback testing and locating alarms.
<i>MPLS</i>	Describes tasks for configuring, monitoring, and troubleshooting an example MPLS network. Tasks include verifying the correct configuration of the MPLS and RSVP protocols, displaying the status and statistics of MPLS running on all routers in the network, and using the layered MPLS troubleshooting model to investigate problems with an MPLS network.
<i>MPLS Log Reference</i>	Describes MPLS status and error messages that appear in the output of the show mpls lsp extensive command. The guide also describes how and when to configure Constrained Shortest Path First (CSPF) and RSVP trace options, and how to examine a CSPF or RSVP failure in a sample network.
<i>Hardware</i>	Describes tasks for monitoring M-series and T-series routing platforms.

Table 4 lists the software and hardware guides and release notes for Juniper Networks J-series, M-series, and T-series routing platforms and describes the contents of each document.

Table 4: Technical Documentation for J-series, M-series, and T-series Routing Platforms (1 of 3)

Document	Description
JUNOS Internet Software for J-series, M-series, and T-series Routing Platforms Configuration Guides	
<i>Class of Service</i>	Provides an overview of the class-of-service (CoS) functions of the JUNOS software and describes how to configure CoS features, including configuring multiple forwarding classes for transmitting packets, defining which packets are placed into each output queue, scheduling the transmission service level for each queue, and managing congestion through the random early detection (RED) algorithm.
<i>CLI User Guide</i>	Describes how to use the JUNOS command-line interface (CLI) to configure, monitor, and manage Juniper Networks routing platforms. This material was formerly covered in <i>JUNOS System Basics Configuration Guide</i> .
<i>Feature Guide</i>	Provides a detailed explanation and configuration examples for several of the most complex features in the JUNOS software.
<i>JUNOS-FIPS</i>	(M-series and T-series routing platforms only) Provides an overview of JUNOS-FIPS 140-2 concepts and describes how to install and configure the JUNOS-FIPS software. Describes FIPS-related commands and how to configure, authorize, and zeroize the Adaptive Services (AS) II FIPS Physical Interface Card (PIC).
<i>MPLS Applications</i>	Provides an overview of traffic engineering concepts and describes how to configure traffic engineering protocols.
<i>Multicast Protocols</i>	Provides an overview of multicast concepts and describes how to configure multicast routing protocols.
<i>Network Interfaces</i>	Provides an overview of the network interface functions of the JUNOS software and describes how to configure the network interfaces on the routing platform.
<i>Network Management</i>	Provides an overview of network management concepts and describes how to configure various network management features, such as SNMP and accounting options.
<i>Policy Framework</i>	Provides an overview of policy concepts and describes how to configure routing policy, firewall filters, forwarding options, and cflowd.
<i>Routing Protocols</i>	Provides an overview of routing concepts and describes how to configure routing, routing instances, and unicast routing protocols.
<i>Services Interfaces</i>	Provides an overview of the services interfaces functions of the JUNOS software and describes how to configure the services interfaces on the routing platform.
<i>Software Installation and Upgrade Guide</i>	Provides a description of JUNOS software components and packaging, and includes detailed information about how to initially configure, reinstall, and upgrade the JUNOS system software. This material was formerly covered in <i>JUNOS System Basics Configuration Guide</i> .
<i>System Basics</i>	Describes Juniper Networks routing platforms, and provides information about how to configure basic system parameters, supported protocols and software processes, authentication, and a variety of utilities for managing your router on the network.
<i>VPNs</i>	Provides an overview and describes how to configure Layer 2 and Layer 3 virtual private networks (VPNs), virtual private LAN service (VPLS), and Layer 2 circuits. Provides configuration examples.

Table 4: Technical Documentation for J-series, M-series, and T-series Routing Platforms (2 of 3)

Document	Description
JUNOS References	
<i>Hierarchy and RFC Reference</i>	Describes the JUNOS configuration mode commands. Provides a hierarchy reference that displays each level of a configuration hierarchy, and includes all possible configuration statements that can be used at that level. This material was formerly covered in <i>JUNOS System Basics Configuration Guide</i> .
<i>Interfaces Command Reference</i>	Describes the JUNOS software operational mode commands you use to monitor and troubleshoot interfaces.
<i>Routing Protocols and Policies Command Reference</i>	Describes the JUNOS software operational mode commands you use to monitor and troubleshoot routing protocols and policies, including firewall filters.
<i>System Basics and Services Command Reference</i>	Describes the JUNOS software operational mode commands you use to monitor and troubleshoot system basics, including commands for real-time monitoring and route (or path) tracing, system software management, and chassis management. Also describes commands for monitoring and troubleshooting services such as CoS, IP Security (IPSec), stateful firewalls, flow collection, and flow monitoring.
<i>System Log Messages Reference</i>	Describes how to access and interpret system log messages generated by JUNOS software modules and provides a reference page for each message.
J-Web User Guide	
<i>J-Web Interface User Guide</i>	Describes how to use the J-Web GUI to configure, monitor, and manage Juniper Networks routing platforms.
JUNOS API and Scripting Documentation	
<i>JUNOScript API Guide</i>	Describes how to use the JUNOScript application programming interface (API) to monitor and configure Juniper Networks routing platforms.
<i>JUNOS XML API Configuration Reference</i>	Provides reference pages for the configuration tag elements in the JUNOS XML API.
<i>JUNOS XML API Operational Reference</i>	Provides reference pages for the operational tag elements in the JUNOS XML API.
<i>JUNOS Configuration and Diagnostic Automation Guide</i>	Provides an overview, instructions for using, and examples of the commit script and self-diagnosis features of the JUNOS software. This guide explains how to enforce custom configuration rules defined in scripts that run at commit time, how to use commit script macros to provide simplified aliases for frequently used configuration statements, and how to configure diagnostic event policies and actions associated with each policy.
<i>NETCONF API Guide</i>	Describes how to use the NETCONF API to monitor and configure Juniper Networks routing platforms.
JUNOS Comprehensive Index and Glossary	
<i>Comprehensive Index and Glossary</i>	Provides a complete index of all JUNOS software books, the <i>JUNOScript API Guide</i> , and the <i>NETCONF API Guide</i> . Also provides a comprehensive glossary.
JUNOScope Documentation	
<i>JUNOScope Software User Guide</i>	Describes the JUNOScope software GUI, how to install and administer the software, and how to use the software to manage routing platform configuration files and monitor routing platform operations.
J-series Services Router Documentation	
<i>J-series Services Router Getting Started Guide</i>	Provides an overview, basic instructions, and specifications for J-series Services Routers. The guide explains how to prepare your site for installation, unpack and install the router and its components, install licenses, and establish basic connectivity.
<i>J-series Services Router Basic LAN and WAN Access Configuration Guide</i>	Explains how to configure the interfaces on J-series Services Routers for basic IP routing with standard routing protocols, ISDN backup, and digital subscriber line (DSL) connections.

Table 4: Technical Documentation for J-series, M-series, and T-series Routing Platforms (3 of 3)

Document	Description
<i>J-series Services Router Advanced WAN Access Configuration Guide</i>	Explains how to configure J-series Services Routers in virtual private networks (VPNs) and multicast networks, configure data link switching (DLSw) services, and apply routing techniques such as policies, stateless and stateful firewall filters, IP Security (IPSec) tunnels, and class-of-service (CoS) classification for safer, more efficient routing.
<i>J-series Services Router Administration Guide</i>	Shows how to manage users and operations, monitor network performance, upgrade software, and diagnose common problems on J-series Services Routers.
M-series and T-series Hardware Documentation	
<i>Hardware Guide</i>	Describes how to install, maintain, and troubleshoot routing platforms and components. Each platform has its own hardware guide.
<i>PIC Guide</i>	Describes the routing platform PICs. Each platform has its own PIC guide.
Release Notes	
<i>JUNOS Release Notes</i>	Summarize new features and known problems for a particular software release, provide corrections and updates to published JUNOS, JUNOScript, and NETCONF manuals, provide information that might have been omitted from the manuals, and describe upgrade and downgrade procedures.
<i>Hardware Release Notes</i>	Describe the available documentation for the routing platform and the supported PICs, and summarize known problems with the hardware and accompanying software. Each platform has its own release notes.
<i>JUNOScope Software Release Notes</i>	Contain corrections and updates to the published JUNOScope manual, provide information that might have been omitted from the manual, and describe upgrade and downgrade procedures.
<i>J-series Services Router Release Notes</i>	Briefly describe the J-series Services Router features, identify known hardware problems, and provide upgrade and downgrade instructions.

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. Send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at <http://www.juniper.net/techpubs/docbug/docbugreport.html>. 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

How to Request Support

For technical support, open a support case using the Case Manager link at <http://www.juniper.net/support/> or call 1-888-314-JTAC (from the United States, Canada, or Mexico) or 1-408-745-9500 (from elsewhere).

Part 1

Juniper Networks Hardware and Software

- Juniper Networks Router Overview on page 3
- Cheat Sheet for the CLI Commands on page 17
- Work with Problems on Your Network on page 29

Chapter 1

Juniper Networks Router Overview

Each Juniper Networks M-series and T-series routing platform is a complete routing system that supports a variety of high-speed interfaces (including SONET/SDH, Ethernet, and ATM) for large networks and network applications. Juniper Networks routers share common JUNOS software, features, and technology for compatibility across platforms.

Application-specific integrated circuits (ASICs) form a definitive part of the router design and enable the router to achieve data forwarding rates that match current fiber-optic capacity. All M-series routers use the Internet Processor II ASIC, which performs the route lookup function and several types of packet processing, such as filtering, policing, rate limiting, and sampling. The T-series platforms use the new T-series Internet Processor for route lookups and notification forwarding.

This chapter provides a general overview of Juniper Networks M-series and T-series routers and routing platforms:

- Router Architecture on page 4
- Hardware Components on page 9
- Monitor Hardware Components on page 14

Router Architecture

This section describes the following:

- Router Architecture for M-series Routers and T-series Platforms on page 4
- Data Flow through the Packet Forwarding Engine on page 5
- Data Flow through an M-series Router on page 6
- Data Flow through a T-series Routing Platform on page 7

Router Architecture for M-series Routers and T-series Platforms

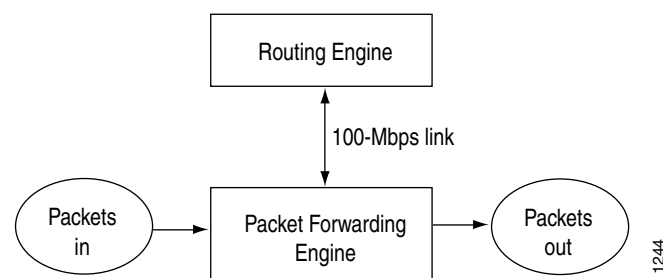
The router architecture of each Juniper Networks M-series router and T-series platform cleanly separates routing and control functions from packet forwarding operations, thereby eliminating bottlenecks and permitting the router to maintain a high level of performance. Each router consists of two major architectural components:

- The Routing Engine, which provides Layer 3 routing services and network management.
- The Packet Forwarding Engine, which provides all operations necessary for transit packet forwarding.

The Routing Engine and Packet Forwarding Engine perform their primary tasks independently, while constantly communicating through a high-speed internal link. This arrangement provides streamlined forwarding and routing control and the capability to run Internet-scale networks at high speeds.

Figure 1 illustrates the relationship between the Routing Engine and the Packet Forwarding Engine.

Figure 1: Router Architecture

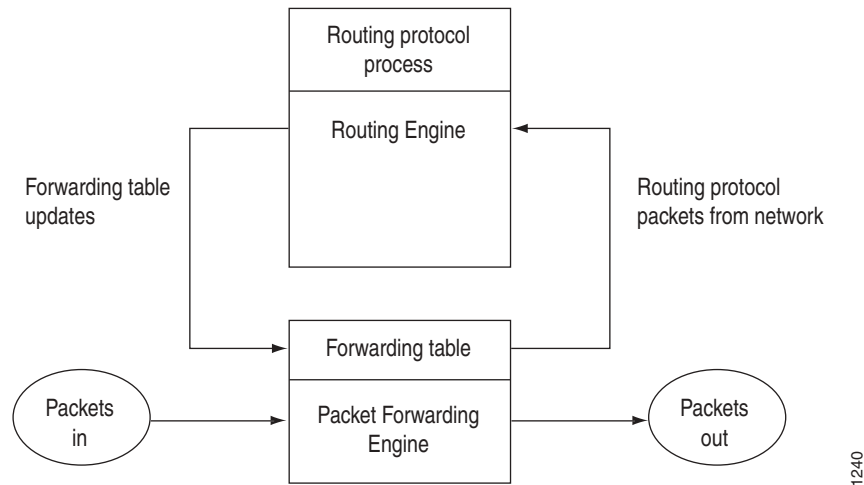


The Routing Engine consists of an Intel-based PCI platform running JUNOS software. For more information about JUNOS software, see “Cheat Sheet for the CLI Commands” on page 17 and “Command-Line Interface Overview” on page 321.

The Routing Engine constructs and maintains one or more routing tables. From the routing tables, the Routing Engine derives a table of active routes, called the forwarding table, which is then copied into the Packet Forwarding Engine.

The design of the Internet Processor II and T-series Internet Processor ASICs allows the forwarding table in the Packet Forwarding Engine to be updated without interrupting forwarding performance (see Figure 2).

Figure 2: Routing and Forwarding Table Updates



The Packet Forwarding Engine uses ASICs to perform Layer 2 and Layer 3 packet switching, route lookups, and packet forwarding. On M-series routers, the Packet Forwarding Engine includes the router midplane (on an M40 router, the backplane), Flexible PIC Concentrators (FPCs), Physical Interface Cards (PICs), and other components, unique to each router, that handle forwarding decisions.

The T-series platforms feature multiple Packet Forwarding Engines, up to a maximum of 16 for the T640 Internet routing node and 8 for the T320 Internet router. Each FPC has one or two Packet Forwarding Engines, each with its own memory buffer. Each Packet Forwarding Engine maintains a high-speed link to the Routing Engine. For information about T-series platforms, see the *T640 Internet Routing Node Hardware Guide* and the *T320 Internet Router Hardware Guide*.

Data Flow through the Packet Forwarding Engine

You can understand the function of the Packet Forwarding Engine by following the flow of a packet through the router: first into a PIC, then through the switching fabric, and finally out another PIC for transmission on a network link. Generally, the data flows through the Packet Forwarding Engine as follows:

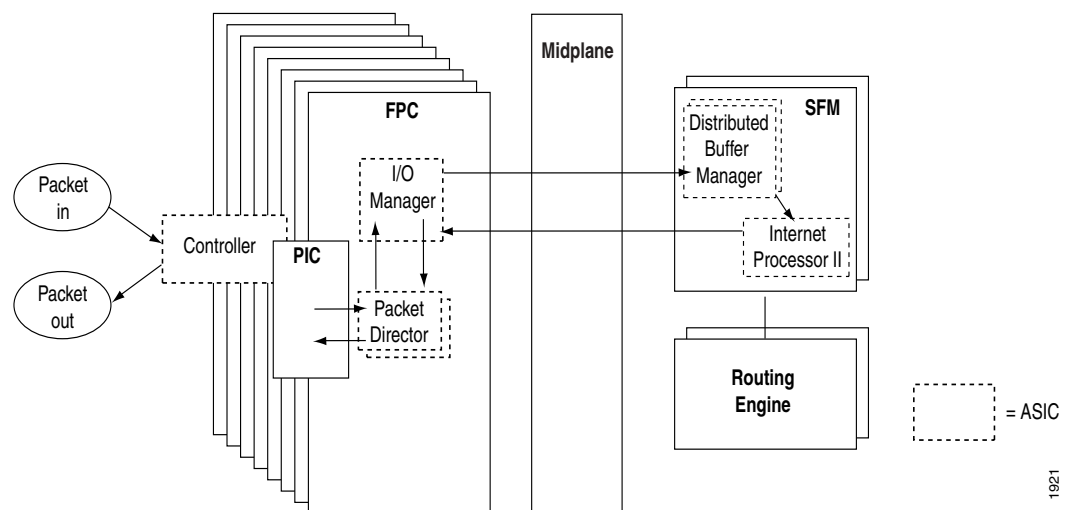
1. Packets enter the router through incoming PIC interfaces, which contain controllers that perform media-specific processing.
2. The PICs pass the packets to the FPCs, where they are divided into cells and are distributed to the router's buffer memory.
3. The Packet Forwarding Engine performs route lookups, forwards the notification to the destination port, reassembles the cells into packets, and sends them to the destination port on the outgoing PIC.
4. The PIC performs encapsulation and other media-specific processing, and sends the packets out into the network.

Data Flow through an M-series Router

Figure 3 illustrates the flow of data packets through an M-series router, using the M40e router architecture as an example. In this example, data flows in the following sequence:

1. A packet enters through the incoming PIC, which parses and de-encapsulates the packet, then passes it to the FPC.
2. On the FPC, the Packet Director ASIC distributes packets to the active I/O Manager ASICs, where each is divided into cells and sent across the midplane to the Switching and Forwarding Modules (SFMs). (On the M40e router, only one SFM is online at a time.) In addition, the behavior aggregate (BA) classifier determines the forwarding treatment for each packet.

Figure 3: Data Flow through an M40e Router



3. When cells arrive at an SFM, the Distributed Buffer Manager ASIC writes them into packet buffer memory, which is distributed evenly across the router's FPCs. The Distributed Buffer Manager ASIC also extracts information needed for route lookups and passes the information to the Internet Processor II ASIC.
4. The Internet Processor II ASIC performs the lookup in the full forwarding table, and finds the outgoing interface and specific next hop for each packet. In addition, the Internet Processor II ASIC performs filtering, policing, sampling and multfield classification, if configured.
5. The forwarding table forwards all unicast packets that do not have options and any multicast packets that have been previously cached. Packets with options are sent to the Routing Engine for resolution.
6. After the Internet Processor II has determined the next hop, it notifies a second Distributed Buffer Manager ASIC, which forwards the notification to the outgoing FPC. Queueing policy and rewrites occur at this time on the egress router. A pointer to the packet is queued at the outgoing port.

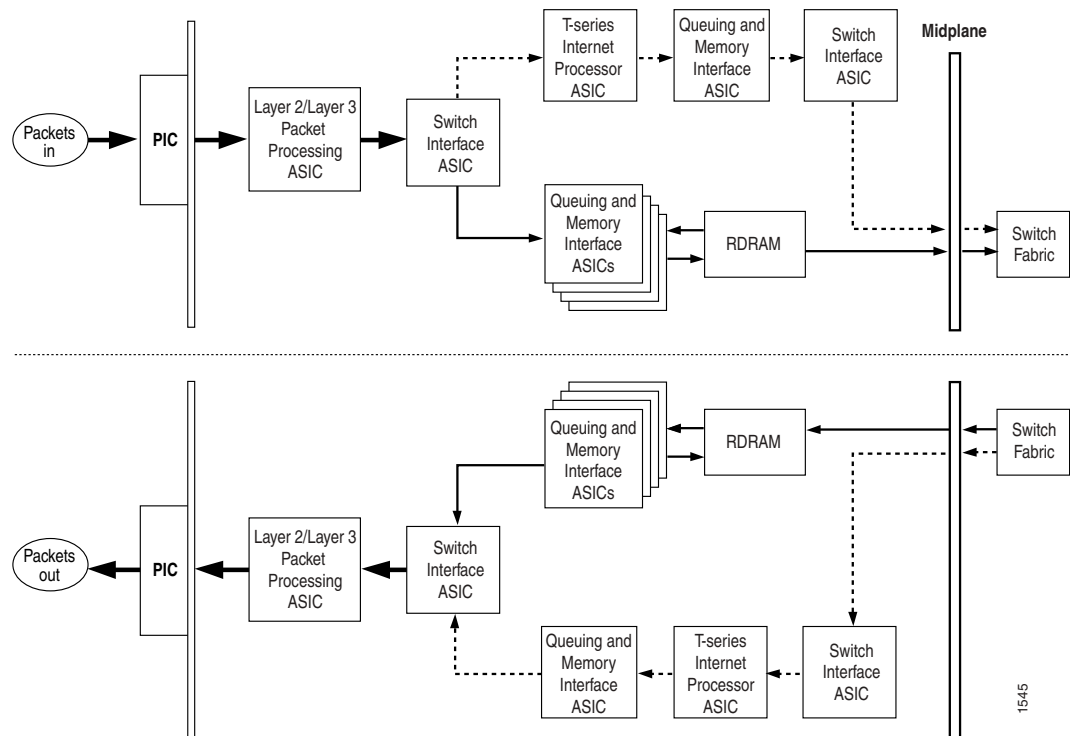
7. When the packet pointer reaches the front of the queue and is ready for transmission, the cells are read from packet buffer memory and are reassembled into the packet, which is passed to the outgoing PIC interface.
8. The PIC performs media-specific processing and sends the packet into the network.

Data Flow through a T-series Routing Platform

Figure 4 illustrates the data flow through a T640 routing node. In this example, data flows in the following sequence:

1. Packets enter through an incoming PIC and are passed to the Packet Forwarding Engine on the originating FPC.
2. The Layer2/Layer 3 Packet Processing ASIC parses the packets and divides them into cells. In addition, the behavior aggregate (BA) classifier determines the forwarding treatment for each packet.

Figure 4: Data Flow through a T640 Routing Node



3. The network-facing Switch Fabric ASIC places the lookup key in a notification and passes it to the T-series Internet Processor.
4. The Switch Fabric ASIC also passes the data cells to the Queuing and Memory Interface ASICs for buffering on the FPC.

5. The T-series Internet Processor performs the route lookup and forwards the notification to the Queuing and Memory Interface ASIC. In addition, if configured filtering, policing, sampling and multifeild classification, are performed at this time.
6. The Queuing and Memory Interface ASIC sends the notification to the switch-fabric-facing Switch Interface ASIC, which sends bandwidth requests through the switch fabric to the destination port, and issues read requests to the Queuing and Memory Interface ASIC to begin reading data cells out of memory.
7. The Switch Interface ASIC on the destination FPC sends bandwidth grants through the switch fabric to the originating Switch Interface ASIC.
8. Upon receipt of each grant, the originating Switch Interface ASIC sends a cell through the switch fabric to the the destination Packet Forwarding Engine.
9. On the destination Packet Forwarding Engine, the switch-fabric-facing Switch Interface ASIC receives the data cells, places the lookup key in a notification, and forwards the notification to the T-series Internet Processor.
10. The T-series Internet Processor performs the route lookup and forwards the notification to the Queuing and Memory Interface ASIC, which forwards it to the network-facing Switch Interface ASIC.
11. The Switch Interface ASIC sends requests to the Queuing and Memory Interface ASIC to read the data cells out of memory, and passes the cells to the Layer2/Layer 3 Packet Processing ASIC, which reassembles the cells into packets, performs the necessary Layer 2 encapsulation, and sends the packets to the outgoing PIC. Queueing policy and rewrites occur at this time on the egress router.
12. The PIC passes the packets into the network.

For more information about the M-series routers and T-series platforms, see the router platform-specific hardware guide, and the *JUNOS Hardware Network Operations Guide*.

Hardware Components

Each Juniper Networks router consists of a chassis and a set of components, including FPCs, PICs, Routing Engines, power supplies, cooling system, and cable management system. Many of the components are field-replaceable units. The following major components are discussed in this section:

- Chassis on page 9
- Flexible PIC Concentrators on page 10
- Physical Interface Cards on page 10
- Routing Engine on page 12
- Power Supplies on page 12
- Cooling System on page 13

Chassis

Chassis dimensions are listed in the physical specifications table for each router. For more information about chassis dimensions, see the router platform-specific hardware guide.

Each Juniper Networks router features a rigid sheet metal chassis that houses all of the router components. The chassis are designed to install into a variety of racks, including standard 19-inch equipment racks, telco center-mount racks, and four-post racks and cabinets. See Table 5 for the maximum number of each router type that can be installed into a rack. Each chassis includes mounting ears or support posts to facilitate rack mounting, and one or more points for connecting an electrostatic discharge (ESD) wrist strap for use when servicing the router.

Table 5: Maximum Number of Routers per Rack

Router or Routing Node	Maximum in Standard Rack
T640	2
T320	3
M160	2
M40e	2
M40	2
M20	5
M5 and M10	14

Each chassis includes a midplane (called the backplane on an M40 router). The midplane transfers data packets to and from the FPCs, distributes power to router components, and provides signal connectivity to the router components for system monitoring and control.

Flexible PIC Concentrators

The FPCs house the PICs used in the router and connect them to other router components. FPCs install into the front of the router in either a vertical or horizontal orientation, depending on the router. A compatible FPC can be installed into any available FPC slot, regardless of the PICs it contains. If a slot is not occupied by an FPC, a blank FPC panel must be installed to shield the empty slot and allow cooling air to circulate properly through the FPC card cage. Some routers support more than one type of FPC. Generally, the FPCs for each router are unique to that router; however, M20 and M40 FPCs are interchangeable, and the M40e and M160e type 1 FPCs are also interchangeable.

Physical Interface Cards

Juniper Networks M-series routers and T-series platforms use PICs to connect to a wide variety of network media. PICs receive incoming packets from the network and transmit outgoing packets to the network, performing framing and line-speed signaling for their specific media type. Before transmitting outgoing data packets, the PICs encapsulate the packets received from the FPCs. Each PIC is equipped with an ASIC that performs control functions specific to the PIC's media type.

See Table 6 for a list of current PIC interfaces.

Table 6: PIC Media Types

Media Type	Ports	Slots	Connectors	JUNOS Interface Name
ATM DS-3	4	Single	SC duplex	at
ATM E3	4	Single		
ATM OC-3	2	Single		
ATM OC-12	1	Single		
Channelized DS-3	4	Single	Posilock to BNC	–
Channelized E1	10	Single	RJ-48	
Channelized OC-12	1	Single	SC duplex	
Channelized STM-1	1	Single	SC duplex	
Multichannel DS-3	2	Single	Posilock to BNC	
DS-3	4	Single	Posilock to BNC	t3
E1	4	Single	RJ-48 or BNC	e1
E3	4	Single	Posilock to BNC	e3
Fast Ethernet	4 8 12 48	Single Single Single Single	RJ-45	fe
Gigabit Ethernet	1 2 4 4	Single Single Quad Single	SC duplex SC duplex SC duplex SC duplex	ge
10-Gigabit Ethernet	1	Quad	SC duplex	
ES	–	Single	–	es
Monitoring Services	–	Single	–	–
Multilink Services	–	Single	–	ml
Tunnel Services	–	Single	–	gr or ip

Media Type	Ports	Slots	Connectors	JUNOS Interface Name
SONET/SDH OC-3c	4	Single	SC duplex	so
SONET/SDH OC-12c	1	Single		
SONET/SDH OC-12c	4	Single		
SONET/SDH OC-48c	1	Quad		
SONEC-SDH OC-48c	1	Single		
SONET/SDH OC-48c	4	Single		
SONET/SDH OC-48c	4	Quad	LC duplex	
SONET/SDH OC-192c	1	Quad		
SONET/SDH OC-192c	1	Single		
T1	4	Single	RJ-48	t1

PICs install into the FPCs (on the M5 and M10 routers, into the FEB). Each FPC can accept up to four PICs. The PICs for each router are unique to that router.

See Table 7 for the number and type of PICs that are currently supported on each router.

Table 7: PICs Supported on Each M-series Router

Supported PICs	M160	M40e	M20 and M40	M5 and M10
ATM DS-3	4 per FPC		4 per FPC	M5-4, M10-8
ATM E3	4 per FPC		4 per FPC	M5-4, M10-8
ATM OC-3	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
ATM OC-12	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
Channelized DS-3	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
Channelized E1	4 per FPC		4 per FPC	M5-4, M10-8
Channelized OC-12	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
Channelized STM-1	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
Multichannel DS-3	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
DS-3	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
E1	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
E3	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
Fast Ethernet	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
Gigabit Ethernet	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
10-Gigabit Ethernet	1 per FPC		1 per FPC	M5-1, M10-2
ES	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
Monitoring Services	4 per FPC			
Multilink Services		4 per FPC	4 per FPC	M5-4, M10-8
Tunnel Services		4 per FPC	4 per FPC	M5-4, M10-8
SONET/SDH OC-3c	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
SONET/SDH OC-12c	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8
SONET/SDH OC-48c	4 per FPC	1 per FPC	1 per FPC	M5-1, M10-2
SONET/SDH OC-192c	1 per FPC			
T1	4 per FPC	4 per FPC	4 per FPC	M5-4, M10-8

Table 8: PICs Supported on Each T-series Router

Supported PICs	T640	T320
Gigabit Ethernet	4 per FPC	2 per FPC
10-Gigabit Ethernet	4 per FPC	2 per FPC
ES		
Monitoring Services		
Multilink Services		
Tunnel Services	4 per FPC	2 per FPC
SONET/SDH OC-3c		
SONET/SDH OC-12c	4 per FPC	2 per FPC
SONET/SDH OC-48c	4 per FPC	2 per FPC
SONET/SDH OC-192c	4 per FPC	2 per FPC

Routing Engine

The Routing Engine consists of an Intel-based PCI platform running the JUNOS software. The Routing Engine maintains the routing tables used by the router in which it is installed and controls the routing protocols on the router. The T640 routing node, and the T320, M160, M40e, and M20 routers support up to two Routing Engines, while the M40, M10, and M5 routers support a single Routing Engine.

Each Routing Engine consists of a CPU; SDRAM for storage of the routing and forwarding tables and other processes; a compact flash disk for primary storage of software images, configuration files, and microcode; a hard disk for secondary storage; a PC card slot (on some M40 routers, a floppy disk) for storage of software upgrades; and interfaces for out-of-band management access.

Power Supplies

Each Juniper Networks M-series router and T-series platform has one, two, or four load-sharing power supplies. A single power supply can provide full power while the router is operational. The power supplies are redundant: if a power supply is removed or fails, the other power supplies automatically assume the electrical load. For more information about the redundant power supplies in each router, see “Power Supplies” in the router platform-specific hardware guide.

The power supplies are connected to the router midplane (on an M40 router, to the router backplane), which distributes the different output voltages throughout the router and its components. Some routers can operate using either AC and DC power; other routers operate with DC power only. For information about the type of power used by each router, see the “Electrical Specifications” table in the router platform-specific hardware guide.

Cooling System

Each Juniper Networks M-series router and T-series platform features a cooling system designed to keep all router components within recommended operating temperature limits. If one component of the cooling system fails or is removed, the system automatically adjusts the speed of the remaining components to keep the temperature within the acceptable range. The cooling system for each router is unique and can consist of fans, impellers, and air filters. For information about the cooling system components of each router, see the “Major Hardware Components” table in the router platform-specific hardware guide.

Monitor Hardware Components



NOTE: If the System Control Board (SCB), System and Switch Board (SSB), or Forwarding Engine Board (FEB) is not running, information about chassis components is not available through the command-line interface (CLI).

Action To use the CLI to monitor Juniper Networks routers, follow these steps:

1. Log in to the router. The CLI operational mode prompt (`>`) appears.

If the operational mode prompt does not appear when you log in to the router, type **cli** to start the JUNOS software and enter operational mode. The prompt changes to `>`, indicating that you are in operational mode.

2. Use one of the operational mode CLI commands listed in Table 9 to monitor router hardware.

Table 9: Operational Mode CLI Commands for Router Monitoring

Command	Description
<code>show version</code>	Displays the router hostname, model number, and version of JUNOS software running on the router.
<code>show chassis firmware</code>	Displays the version of firmware running on the SCB, SFM, SSB, FEB, and FPCs.
<code>show chassis hardware</code>	Displays an inventory of the hardware components installed in the router, including the component name, version, part number, serial number, and a brief description.
<code>show chassis environment</code>	Displays environmental information about the router chassis, including the temperature and status.
<code>show chassis environment <i>component-name</i></code>	Displays more detailed environmental information for the following router components: FPCs, Front Panel Module (FPM), Miscellaneous Subsystem (MCS), PFE Clock Generator (PCG), Power Entry Module (PEM) or power supply, control board, SONET clock generator (SCG), Switch Interface Board (SIB), Routing Engine, and SFM. This command works only on the M40e, M160, and T320 routers, and the T640 routing node.
<code>show chassis craft-interface</code>	Displays operational status information about the router, including the alarm status and LED status of major components.
<code>show chassis alarms</code>	Displays the current router component alarms that have been generated, including the date, time, severity level, and description.
<code>show chassis <i>component-name</i></code>	Displays more detailed operational status information about the FPCs, Routing Engine, FEB, SCB, SFMs, and SSB router components, including the temperature of air passing by the Switch Plane Processor (SPP) card and the Switch Plane Router (SPR) card (the two SFM serial components), in degrees Centigrade. The command displays the total CPU DRAM and SRAM being used by the SFM processor. The command output displays the time that the SFM became active and how long the SFM has been up and running. A small uptime can indicate a problem.
<code>show log messages</code>	<p>Displays the contents of the <code>messages</code> system log file that records messages generated by component operational events, including error messages generated by component failures.</p> <p>To monitor the <code>messages</code> file in real time, use the <code>monitor start messages</code> CLI command. This command displays the new entries in the file until you stop monitoring by using the <code>monitor stop messages</code> CLI command.</p>

Command	Description
show log chassisd	<p>Displays the contents of the chassis daemon (chassisd) log file that keeps track of the state of each chassis component</p> <p>To monitor the chassisd file in real time, use the monitor start chassisd CLI command. This command displays the new entries in the file until you stop monitoring by using the monitor stop chassisd CLI command.</p>
request support information	<p>Use this command when you contact the Juniper Networks Technical Assistance Center (JTAC) about your component problem. This command is the equivalent of using the following CLI commands (see “Contact JTAC” on page 15):</p> <ul style="list-style-type: none"> ■ show version ■ show chassis firmware ■ show chassis hardware ■ show chassis environment ■ show interfaces extensive (for each configured interface) ■ show configuration (excluding any SECRET-DATA) ■ show system virtual-memory

Contact JTAC

If you cannot determine the cause of a problem or need additional assistance, contact JTAC at support@juniper.net or at 1-888-314-JTAC (within the United States) or 1-408-745-9500 (from outside the United States). For details on the information you need to provide for JTAC, See “Contact JTAC” on page 99. For steps to return a failed component, see “Return the Failed Component” on page 100.

Chapter 2

Cheat Sheet for the CLI Commands

This chapter provides quick reference information for the JUNOS software command-line interface (CLI). For more detailed information about using the CLI, see “Command-Line Interface Overview” on page 321.

- CLI Operational Mode Top-Level Commands on page 18
- CLI Configuration Mode Top-Level Commands on page 20
- Load a Configuration Using Copy and Paste Commands on page 22
- CLI Keyboard Shortcuts on page 26
- Manage Output at the ---(more)--- Prompt on page 27

CLI Operational Mode Top-Level Commands

In operational mode, you enter commands to monitor and diagnose the software, network connectivity, and the router. When you log in to the router and the CLI starts, you are at the top level of the CLI operational mode. At this level, there are several broad groups of CLI commands. Table 10 lists the top-level CLI operational mode commands and describes the options available for each command. The commands are listed in alphabetical order.

Table 10: CLI Operational Mode Top-Level Commands

Command	Description
clear	Clear statistics and protocol database information. Syntax: clear (arp bgp firewall helper igmp ike ilmi interfaces ipsec ipv6 isis ldp log mpls msdp multicast ospf pim rip ripng route rsvp snmp system vrrp)
configure	Enter CLI configuration mode. Alternative commands: configure <exclusive> <private>
file	Perform file manipulation operations, such as copy, delete, list, rename, and show. Syntax: file (compare copy delete list rename show)
help	Provide help information. Syntax: help (reference syslog topic)
monitor	Monitor a log file or interface traffic in real time. Syntax: monitor (interface list start stop traffic)
mtrace	Display trace information about a multicast path from a source to a receiver. Syntax: mtrace (from-source monitor to-gateway)
ping	Verify IP connectivity to another IP host or Asynchronous Transfer Mode (ATM) connectivity (ping ATM) using Operation Administration and Maintenance (OAM) cells to an ATM endstation. Syntax: ping host <interface source-interface> <bypass-routing> <count requests> <do-not-fragment> <interval seconds> <pattern string> <record-route> <routing-instance routing-instance-name> <size bytes> <strict> <tos type-of-service> <ttl value> <via route> <rapid detail> Syntax: ping atm interface interface <count count> <end-to-end segment> <interval interval> <sequence-number sequence-number> <vci vci> <brief> Syntax: ping vpn-interface vpn-interface host <local echo-address>
pipe	Filter the output of an operational mode or configuration mode command. Syntax: (compare count display <detail inheritance xml> except pattern find pattern last lines match pattern no-more resolve <file-names> save filename trim columns)
quit	Log out from the CLI process. Syntax: quit
request	Make system-level requests, such as halt or reboot the router, load software packages, and back up the router's file systems. Syntax: request system (halt reboot snapshot software)
restart	Restart the router hardware or software processes. Syntax: restart (fpc class-of-service gracefully immediately interface-control mib-process network-access-service remote-operations routing sampling sfm snmp soft)
set	Set CLI properties, the router's date and time, and the craft interface display text. Syntax: set (chassis cli date)

Command	Description
show	<p>Show information about all aspects of the software, including interfaces and routing protocols.</p> <p>Syntax: <code>show (accounting aps arp as-path bgp chassis cli configuration connections dvmrp firewall helper host igmp ike ilmi interfaces ipsec ipv6 isis l2circuit l2vpn ldp link-management log mpls msdp multicast ntp ospf pfe pim policer policy rip ripng route rsvp sap snmp system task ted version vrrp)</code></p>
ssh	<p>Open a secure shell to another host.</p> <p>Syntax: <code>ssh host <bypass-routing> <routing-instance routing-instance-name> <source address> <vpn-interface vpn-interface> <v1 v2></code></p>
start	<p>Start a software process.</p> <p>Syntax: <code>start shell</code></p>
telnet	<p>Start a telnet session to another host.</p> <p>Syntax: <code>telnet host <8bit> <bypass-routing> <inet inet6> <noresolve> <port port> <interface interface-name> <routing-instance routing-instance-name> <source address> <vpn-interface vpn-interface></code></p>
test	<p>Run various diagnostic debugging commands.</p> <p>Syntax: <code>test (configuration interface msdp policy)</code></p>
traceroute	<p>Trace the route to a remote host.</p> <p>Syntax: <code>traceroute host <as-number-lookup> <bypass-routing> <gateway address> <inet inet6> <noresolve> <routing-instance routing-instance-name> <source address> <tos value> <ttr value> <vpn-interface vpn-interface> <wait seconds></code></p>

CLI Configuration Mode Top-Level Commands

In configuration mode, you configure the JUNOS software by creating a hierarchy of configuration statements. You can do this using the CLI or by creating a text (ASCII) file that contains the statement hierarchy. (The statement hierarchy is identical in both the CLI and the text configuration file.) You can configure all properties of the JUNOS software, including interfaces, general routing information, routing protocols, and user access, as well as several system hardware properties. When you have finished entering the configuration statements, you commit them, which activates the configuration on the router.

Table 11 lists each CLI configuration mode command and describes the options available for each command. The commands are organized alphabetically.

Table 11: CLI Configuration Mode Commands

Command	Description
activate	Remove the <code>inactive:</code> tag from a statement, effectively reading the statement or identifier to the configuration. Statements or identifiers that have been activated take effect when you next issue the <code>commit</code> command. Syntax: <code>activate (statement-path identifier)</code>
annotate	Add comments to a configuration. Syntax: <code>annotate <statement-path> "comment-string"</code>
commit	Commit the set of changes to the database and cause the changes to take operational effect. Syntax: <code>commit <and-quit> <check> <confirmed <minutes>> <synchronize></code>
copy	Make a copy of an existing statement in the configuration. Syntax: <code>copy <statement-path> identifier 1 to identifier 2</code>
deactivate	Add the <code>inactive:</code> tag to a statement, effectively commenting out the statement or identifier from the configuration. Statements or identifiers marked as inactive do not take effect when you issue the <code>commit</code> command. Syntax: <code>deactivate (statement-path identifier)</code>
delete	Delete a statement or identifier. All subordinate statements and identifiers contained within the specified statement path are deleted with it. Syntax: <code>delete (statement-path identifier)</code>
edit	Move inside the specified statement hierarchy. If the statement does not exist, it is created. Syntax: <code>edit <statement-path></code>
exit	Exit the current level of the statement hierarchy, returning to the level prior to the last <code>edit</code> command, or exit from configuration mode. The <code>quit</code> and <code>exit</code> commands are synonyms. Syntax: <code>exit <configuration-mode></code>
help	Display help about available configuration statements. Syntax: <code>help (apropos reference syslog topic) <string></code>
insert	Insert an identifier into an existing hierarchy. Syntax: <code>insert <statement-path> identifier1 (before after) identifier2</code>
load	Load a configuration from an ASCII configuration file or from terminal input. Your current location in the configuration hierarchy is ignored when the load operation occurs. Syntax: <code>load (merge override replace) (filename terminal)</code>

Command	Description
quit	Exit the current level of the statement hierarchy, returning to the level prior to the last edit command, or exit from configuration mode. The quit and exit commands are synonyms. Syntax: quit <configuration-mode>
rename	Rename an existing configuration statement or identifier. Syntax: rename <statement-path> <i>identifier1</i> to <i>identifier2</i>
rollback	Return to a previously committed configuration. The software saves the last 10 committed configurations, including the rollback number, date, time, and name of the user who issued the commit configuration command. rollback 0 erases any configuration changes made to the current candidate configuration. The currently operational JUNOS software configuration is stored in the file juniper.conf , and the last three committed configurations are stored in the files juniper.conf.1.gz , juniper.conf.2.gz , and juniper.conf.3.gz . These four files are located in the directory /config/ , which is on the router's flash drive. The remaining six previous committed configurations, the files juniper.conf.4.gz through juniper.conf.9.gz , are stored in the directory /var/db/config/ , which is on the router's hard disk. Syntax: rollback <number>
run	Run an operational mode CLI command without exiting from configuration mode. Syntax: run <operation-command>
save	Save the configuration to an ASCII file in the user's home directory (by default) or to the user's terminal session. The statement hierarchy and the contents of the current level of the statement hierarchy (and below) are saved. This allows a section of the configuration to be saved, while fully specifying the statement hierarchy. Syntax: save <i>filename</i> <i>terminal</i>
set	Create a statement hierarchy and set identifier values. This is similar to the edit command except that your current level in the hierarchy does not change, and you can set identifier values, while the edit command only allows access to a statement path. Syntax: set (<i>statement-path</i> <i>identifier</i>)
show	Display the current configuration. Syntax: show (<i>statement-path</i> <i>identifier</i>)
status	Display the users currently editing the configuration. Syntax: status
top	Return to the top level of configuration command mode, indicated by the [edit] banner, or execute a command from the top level of the configuration. Syntax: top <configuration-command>
up	Move up one level in the statement hierarchy. Syntax: up <number>
update	Update a private database. For more information on the update command, see the <i>JUNOS System Basics and Services Command Reference</i> . Syntax: update

Load a Configuration Using Copy and Paste Commands

You can load configurations using the copy and paste commands in the following ways:

1. Load a Configuration from a File To a Router on page 22
2. Load a Configuration Using the display set Command on page 24

Load a Configuration from a File To a Router

Purpose You can create a file, copy the file to the local router, and then load the file into the CLI. After you have loaded the file, you can commit it to activate the configuration on the router, or you can edit the configuration interactively using the CLI and commit it at a later time.

Action To load a configuration from a file, follow these steps:

1. Create the configuration in a file using a text editor such as Notepad, making sure that the syntax of the configuration file is correct. See *JUNOS Internet Software Protocols, Class of Service, Chassis, and Management Command Reference*, for information about testing the syntax of a configuration file.
2. In the text file, use an option to perform the required action. The following table lists and describes some options. For an example of a text file, see “What It Means.”

Table 12: Options for the load Command

merge	Combines the current configuration and the configuration in <i>filename</i> or the one that you type at the terminal. A merge operation is useful when you are adding a new section to an existing configuration. If the existing configuration and the incoming configuration contain conflicting statements, the statements in the incoming configuration override those in the existing configuration.
override	Discards the current candidate configuration and loads the configuration in <i>filename</i> or the one that you type at the terminal. When you use the override option and commit the configuration, all system processes reparse the configuration. You can use the override option at any level of the hierarchy.
replace	Searches for the replace tags, deletes the existing statements of the same name, if any, and replaces them with the incoming configuration. If there is no existing statement of the same name, the replace operation adds the statements marked with the replace tag to the configuration. Note: For this operation to work, you must include replace tags in the text file or configuration you type at the terminal.

3. Enter **Ctrl+a** to select all the text, and **Ctrl+c** to copy the contents of the text file to the clipboard.

- On the router, enter configuration mode:

```
user@host> cli
[edit]
user@host#
```

- Load the configuration file:

```
user@host> load merge terminal
```

- At the prompt, paste the contents of the clipboard using the mouse and the paste icon.

```
[edit]
user@host# load merge terminal
[Type ^D at a new line to end input]
> Paste the contents of the clipboard here<
```

- Hit Enter.

- Enter Ctrl+d.

- Commit the configuration to activate it on the router, or you can edit the configuration interactively using the CLI and commit it at a later time.

Sample Output The following is an example of a text file with the **replace** option:

```
interfaces {
  replace:
    so-0/0/0 {
      unit 0 {
        family inet {
          address 10.1.34.1/30;
        }
      }
    }
}

protocols {
  replace:
    isis {
      interface so-0/0/1.0 {
        level 1 metric 10;
        level 2 disable;
      }
      interface fxp0.0 {
        disable;
      }
      interface lo0.0;
    }
}
```

The following output is for Step 4 through Step 8:

```
[edit]
user@R1# load merge terminal
[Type ^D at a new line to end input]
interfaces {
replace:
  so-0/0/0 {
    unit 0 {
      family inet {
        address 10.1.34.1/30;
      }
    }
  }
}
protocols {
replace:
  isis {
    interface so-0/0/1.0 {
      level 1 metric 10;
      level 2 disable;
    }
    interface fxp0.0 {
      disable;
    }
    interface lo0.0;
  }
}
load complete
```

What It Means The sample output shows a configuration loaded from a text file with the **replace** option. For more information about loading a configuration, see the *JUNOS System Basics Configuration Guide*.

Load a Configuration Using the *display set* Command

Purpose In configuration mode only, you can display the configuration as a series of configuration mode commands required to recreate the configuration. This is useful for users who are not familiar with how to use configuration mode commands or for users who wish to cut, paste, and edit the displayed configuration. In addition, you can duplicate the configuration of one router to another.

Action To load a configuration from the local router to a target router, follow these steps:

1. On the local router, enter configuration mode:

```
user@R1> cli
[edit]
user@host#
```

2. Go to the hierarchy level you want to copy. For example:

```
[edit]
user@R1# edit interfaces
```


3. Display the series of configuration commands required to recreate the configuration. For example:

```
[edit interfaces]
user@R1# show | display set
set interfaces so-0/0/0 unit 0 family inet accounting destination-class-usage
set interfaces so-0/0/0 unit 0 family inet address 10.1.12.1/30
set interfaces fxp0 unit 0 family inet address 10.168.70.143/21
set interfaces lo0 unit 0 family inet address 10.0.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.1000.0000.0003.00
```

4. Copy each line of the configuration individually from the local router to the target router. In the target router, you must be at the top level of the configuration and in configuration mode. For example:

```
mwazna@R2> edit
Entering configuration mode
```

```
[edit]
mwazna@R2# set interfaces so-0/0/0 unit 0 family inet accounting
destination-class-usage
```

5. Continue cutting and pasting each line of the configuration.
6. Commit the configuration to activate it on the router, or you can edit the configuration interactively using the CLI and commit it at a later time.

CLI Keyboard Shortcuts

In the CLI, you can use keyboard sequences to move around and edit a command line. You can also use keyboard sequences to scroll through a list of recently executed commands.

The following table lists some of the CLI keyboard sequences.

Table 13: CLI Keyboard Shortcuts

Keyboard sequence	Action
Ctrl+b	Move the cursor back one character.
Esc+b or Alt+b	Move the cursor back one word.
Ctrl + f	Move the cursor forward one character.
Esc+f or Alt+f	Move the cursor forward one word.
Ctrl+a	Move the cursor to the beginning of the command line.
Ctrl+e	Move the cursor to the end of the command line.
Ctrl+h, Delete, or Backspace	Delete the character before the cursor.
Ctrl+d	Delete the character at the cursor.
Ctrl+k	Delete the all characters from the cursor to the end of the command line.
Ctrl+u or Ctrl+x	Delete the all characters from the command line.
Ctrl+w, Esc + Backspace, or Alt + Backspace	Delete the word before the cursor.
Esc+d or Alt+d	Delete the word after the cursor.
Ctrl+y	Insert the most recently deleted text at the cursor.
Ctrl+l	Redraw the current line.
Ctrl+p	Scroll backward through the list of recently executed commands.
Ctrl+n	Scroll forward through the list of recently executed commands.
Ctrl+r	Search the CLI history incrementally in reverse order for lines matching the search string.
Esc+/ or Alt+/ Esc-1 through Esc-9 or Alt-1 through Alt-9	Search the CLI history for words for which the current word is a prefix. Specify the number of times to execute a keyboard sequence.

Manage Output at the **—(more)—** Prompt

If the output is longer than the screen length, it appears one screen at a time with the UNIX **—(more)—** prompt at the end of the screen. The **—(more)—** prompt indicates that more output is available. The following table lists the keyboard sequences you can use at the **—(more)—** prompt.

Table 14: Keyboard Shortcuts at the **—(more)— Prompt**

Keyboard Shortcut	Action
Enter, Return, k, Ctrl+m, Ctrl+n, or down arrow	Scroll down one line.
Tab, d, Ctrl+d, or Ctrl+x	Scroll down one-half screen.
Space or Ctrl+f	Scroll down one whole screen.
Ctrl+e or g	Scroll down to the bottom of the output.
n (or no-more)	Display the output all at once instead of one screen at a time.
j, Ctrl-h, Ctrl-p, or up arrow	Scroll up one line.
u or Ctrl-u	Scroll up one-half screen.
b or Ctrl-b	Scroll up one whole screen.
Ctrl-a or g	Scroll up to the bottom of the output.
/ <i>string</i>	Search forward for a string.
? <i>string</i>	Search backward for a string.
n	Repeat previous search for a string.
m or M (or match <i>string</i>)	Find a text string. You are prompted for the string to match
e or E (or except <i>string</i>)	Find, ignoring a text string. You are prompted for the string to ignore.
Ctrl-C, q, Q, or Ctrl-k	Interrupt the display of output.
H (Same as specifying hold)	Hold the CLI at the More prompt after displaying all output.
c or C	Clear any match conditions and display the complete output.
Ctrl-l	Redraw the output on the screen.
s or S (or save <i>filename</i>)	Save the command output to a file. You are prompted for a filename.

Chapter 3

Work with Problems on Your Network

This chapter describes how to work with problems on your network. It discusses troubleshooting basics, using an example network, and includes the commands you might use to diagnose problems with the router and network.

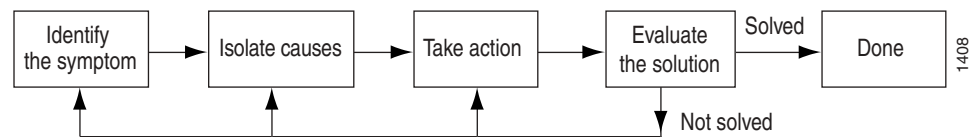
Table 15: Checklist for Working with Problems on Your Network

Work with Problems on Your Network Tasks	Command or Action
Isolate a Broken Network Connection on page 30	
1. Identify the Symptoms on page 31	ping (<i>ip-address</i> <i>hostname</i>) show route (<i>ip-address</i> <i>hostname</i>) tracert (<i>ip-address</i> <i>hostname</i>)
2. Isolate the Causes on page 32	show < configuration interfaces protocols route >
3. Take Appropriate Action on page 33	[edit] delete routing options static route <i>destination-prefix</i> commit and-quit show route <i>destination-prefix</i>
4. Evaluate the Solution on page 34	show route (<i>ip-address</i> <i>hostname</i>) ping (<i>ip-address</i> <i>hostname</i>) count 3 tracert (<i>ip-address</i> <i>hostname</i>)

Isolate a Broken Network Connection

Purpose By applying the standard four-step process illustrated in Figure 5, you can isolate a failed node in the network.

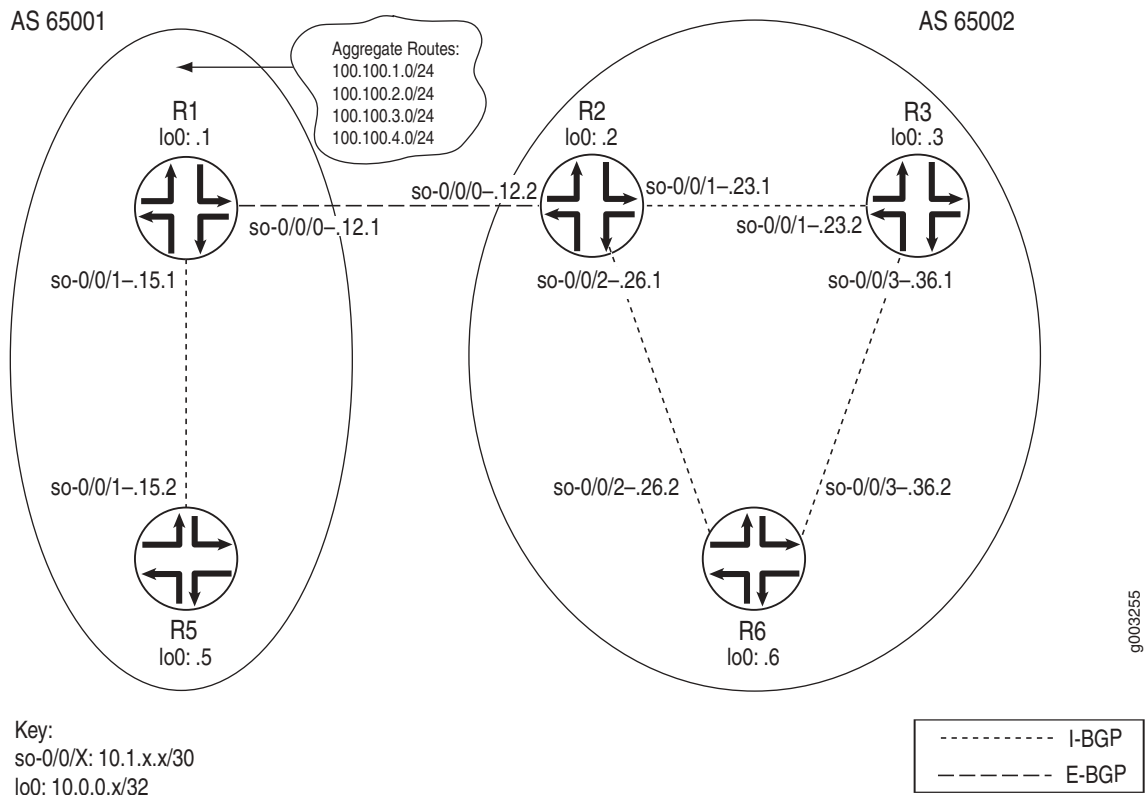
Figure 5: Process for Diagnosing Problems in Your Network



Before you embark on the four-step process, however, it is important that you are prepared for the inevitable problems that occur on all networks. While you might find a solution to a problem by simply trying a variety of actions, you can reach an appropriate solution more quickly if you are systematic in your approach to the maintenance and monitoring of your network. To prepare for problems on your network, understand how the network functions under normal conditions, have records of baseline network activity, and carefully observe the behavior of your network during a problem situation.

Figure 6 shows the network topology used in this section to illustrate the process of diagnosing problems in a network.

Figure 6: Network with a Problem



The network in Figure 6 consists of two autonomous systems (ASs). AS 65001 includes two routers, and AS 65002 includes three routers. The border router (R1) in AS 65001 announces aggregated prefixes **100.100/24** to the AS 65002 network. The problem in this network is that **R6** does not have access to **R5** because of a loop between **R2** and **R6**.

Steps To Take To isolate a failed connection in your network, follow these steps:

1. Identify the Symptoms on page 31
2. Isolate the Causes on page 32
3. Take Appropriate Action on page 33
4. Evaluate the Solution on page 34

Step 1: Identify the Symptoms

Purpose The symptoms of a problem in your network are usually quite obvious, such as the failure to reach a remote host.

Action To identify the symptoms of a problem on your network, start at one end of your network and follow the routes to the other end, entering all or one of the following JUNOS command-line interfaces (CLI) operational mode commands:

```
user@host> ping (ip-address | host-name)
user@host> show route (ip-address | host-name)
user@host> traceroute (ip-address | host-name)
```

Sample Output

```
user@R6> ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS Len ID Flg off TTL Pro cks Src Dst
 4 5 00 0054 e2db 0 0000 01 01 a8c6 10.1.26.2 10.0.0.5

36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS Len ID Flg off TTL Pro cks Src Dst
 4 5 00 0054 e2de 0 0000 01 01 a8c3 10.1.26.2 10.0.0.5

36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS Len ID Flg off TTL Pro cks Src Dst
 4 5 00 0054 e2e2 0 0000 01 01 a8bf 10.1.26.2 10.0.0.5

^C
--- 10.0.0.5 ping statistics ---
3 packets transmitted, 0 packets received, 100% packet loss

user@R6> show route 10.0.0.5

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32          *[IS-IS/165] 00:02:39, metric 10
                    > to 10.1.26.1 via so-0/0/2.0

user@R6> traceroute 10.0.0.5
traceroute to 10.0.0.5 (10.0.0.5), 30 hops max, 40 byte packets
 1 10.1.26.1 (10.1.26.1) 0.649 ms 0.521 ms 0.490 ms
```

```

2 10.1.26.2 (10.1.26.2) 0.521 ms 0.537 ms 0.507 ms
3 10.1.26.1 (10.1.26.1) 0.523 ms 0.536 ms 0.514 ms
4 10.1.26.2 (10.1.26.2) 0.528 ms 0.551 ms 0.523 ms
5 10.1.26.1 (10.1.26.1) 0.531 ms 0.550 ms 0.524 ms

```

What It Means The sample output shows an unsuccessful `ping` command in which the packets are being rejected because the time to live is exceeded. The output for the `show route` command shows the interface (10.1.26.1) that you can examine further for possible problems. The `traceroute` command shows the loop between 10.1.26.1 (R2) and 10.1.26.2 (R6), as indicated by the continuous repetition of the two interface addresses.

Step 2: Isolate the Causes

Purpose A particular symptom can be the result of one or more causes. Narrow down the focus of your search to find each individual cause of the unwanted behavior.

Action To isolate the cause of a particular problem, enter one or all of the following JUNOS CLI operational mode command:

```
user@host> show < configuration | bgp | interfaces | isis | ospf | route >
```

Your particular problem may require the use of more than just the commands listed above. See the appropriate command reference for a more exhaustive list of commonly used operational mode commands.

Sample Output

```

user@R6> show interfaces terse
Interface      Admin Link Proto Local Remote
so-0/0/0       up    up
so-0/0/0.0     up    up    inet 10.1.56.2/30
               up    up    iso
so-0/0/2       up    up
so-0/0/2.0     up    up    inet 10.1.26.2/30
               up    up    iso
so-0/0/3       up    up
so-0/0/3.0     up    up    inet 10.1.36.2/30
               up    up    iso
[...Output truncated...]

```

The following sample output is from R2:

```

user@R2> show route 10.0.0.5

inet.0: 22 destinations, 25 routes (22 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32      *[Static/5] 00:16:21
                  > to 10.1.26.2 via so-0/0/2.0
                  [BGP/170] 3d 20:23:35, MED 5, localpref 100
                  AS path: 65001 I
                  > to 10.1.12.1 via so-0/0/0.0

```

What It Means The sample output shows that all interfaces on R6 are up. The output from R2 shows that a static route `[Static/5]` configured on R2 points to R6 (10.1.26.2) and is the preferred route to R5 because of its low preference value. However, the route is looping from R2 to R6, as indicated by the missing reference to R5 (10.1.15.2).

Step 3: Take Appropriate Action

Purpose The appropriate action depends on the type of problem you have isolated. In this example, a static route configured on R2 is deleted from the [routing-options] hierarchy level. Other appropriate actions might include the following:

- Check the local router's configuration and edit it if appropriate.
- Troubleshoot the intermediate router.
- Check the remote host configuration and edit it if appropriate.
- Troubleshoot routing protocols.
- Identify additional possible causes.

Action To resolve the problem in this example, enter the following JUNOS CLI commands:

```
[edit]
user@R2# delete routing-options static route destination-prefix
user@R2# commit and-quit
user@R2# show route destination-prefix
```

Sample Output

```
[edit]
user@R2# delete routing-options static route 10.0.0.5/32

[edit]
user@R2# commit and-quit
commit complete
Exiting configuration mode

user@R2> show route 10.0.0.5

inet.0: 22 destinations, 24 routes (22 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32          *[BGP/170] 3d 20:26:17, MED 5, localpref 100
                    AS path: 65001 I
                    > to 10.1.12.1 via so-0/0/0.0
```

What It Means The sample output shows the static route deleted from the [routing-options] hierarchy and the new configuration committed. The output for the **show route** command now shows the BGP route as the preferred route, as indicated by the asterisk (*).

Step 4: Evaluate the Solution

Purpose If the problem is solved, you are finished. If the problem remains or a new problem is identified, start the process over again.

You can address possible causes in any order. In relation to the network in Figure 6, we chose to work from the local router toward the remote router, but you might start at a different point, particularly if you have reason to believe that the problem is related to a known issue, such as a recent change in configuration.

Action To evaluate the solution, enter the following JUNOS CLI commands:

```
user@host> show route (ip-address | host-name)
user@host> ping (ip-address | host-name)
user@host> traceroute (ip-address | host-name)
```

Sample Output user@R6> show route 10.0.0.5

```
inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32          *[BGP/170] 00:01:35, MED 5, localpref 100, from 10.0.0.2
                     AS path: 65001 I
                     > to 10.1.26.1 via so-0/0/2.0
```

```
user@R6> ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
64 bytes from 10.0.0.5: icmp_seq=0 ttl=253 time=0.866 ms
64 bytes from 10.0.0.5: icmp_seq=1 ttl=253 time=0.837 ms
64 bytes from 10.0.0.5: icmp_seq=2 ttl=253 time=0.796 ms
^C
--- 10.0.0.5 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.796/0.833/0.866/0.029 ms
```

```
user@R6> traceroute 10.0.0.5
traceroute to 10.0.0.5 (10.0.0.5), 30 hops max, 40 byte packets
 1 10.1.26.1 (10.1.26.1) 0.629 ms 0.538 ms 0.497 ms
 2 10.1.12.1 (10.1.12.1) 0.534 ms 0.538 ms 0.510 ms
 3 10.0.0.5 (10.0.0.5) 0.776 ms 0.705 ms 0.672 ms
```

What It Means The sample output shows that there is now a connection between R6 and R5. The `show route` command shows that the BGP route to R5 is preferred, as indicated by the asterisk (*). The `ping` command is successful and the `traceroute` command shows that the path from R6 to R5 is through R2 (10.1.26.1), and then through R1 (10.1.12.1).

Part 2

JUNOS Internet Software in the Network

- Stop and Start JUNOS Software on page 37
- Display JUNOS Software Information on page 45
- Check Router Configuration on page 49
- Upgrade JUNOS Software on page 57
- Reinstall JUNOS Software on page 73

Chapter 4

Stop and Start JUNOS Software

This chapter describes how to stop and start the JUNOS software after it has been installed. (See Table 16.)

Table 16: Checklist for Stopping and Starting the JUNOS Software

Stop and Start JUNOS Software Tasks	Command or Action
1. Stop the JUNOS Software on page 38	request system halt
2. Reboot the JUNOS Software on page 39	request system reboot
Restart a JUNOS Software Process on page 40	
1. Display Information about Software Processes on page 40	show system processes extensive
2. Restart a JUNOS Software Process on page 41	restart (class-of-service interface-control mib-process network-access-service remote-operations routing sampling snmp) <gracefully> <immediately> <soft>
3. Check That the Process Has Restarted on page 42	show system processes extensive

Stop the JUNOS Software

Purpose To avoid damage to the file system, gracefully shut down the JUNOS software before powering down the router. If you have configured a backup Routing Engine, it must be shut down before the master Routing Engine.

Action To stop the JUNOS software, use the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> request system halt
```

Sample Output

```
user@host> request system halt
Halt the system ? [yes,no] (no) yes
shutdown: [pid 3110]
Shutdown NOW!

*** FINAL System shutdown message from root@host ***
System going down IMMEDIATELY

user@host> Dec 17 17:28:40 init: syslogd (PID 2514) exited with status=0 Normal
Exit
Waiting (max 60 seconds) for system process `bufdaemon' to stop...stopped
Waiting (max 60 seconds) for system process `syncer' to stop...stopped
syncing disks... 4
done
Uptime: 3h31m41s
ata0: resetting devices .. done
The operating system has halted.
Please press any key to reboot.
```

What It Means The sample output shows that all system process have stopped and the operating system was halted immediately. For more detailed information on the **request system halt** command, see the *JUNOS System Basics and Services Command Reference*.

Reboot the JUNOS Software

Purpose Reboot JUNOS software after a software upgrade and occasionally to recover from an error condition.

Action To reboot the JUNOS software, use the following JUNOS CLI operational mode command:

```
user@host> request system reboot
```

Sample Output

```
user@host> request system reboot
Reboot the system ? [yes,no] (no) yes
shutdown: [pid 845]
Shutdown NOW!

*** FINAL System shutdown message from root@host ***
System going down IMMEDIATELY

user@host> Dec 17 17:34:20 init: syslogd (PID 409) exited with status=0 Normal
Exit
Waiting (max 60 seconds) for system process `bufdaemon' to stop...stopped
Waiting (max 60 seconds) for system process `syncer' to stop...stopped
syncing disks... 10 6
done
Uptime: 2m45s
ata0: resetting devices .. done
Rebooting...
```

What It Means The sample output shows the final stages of the system shutdown and the execution of the reboot. Reboot requests are recorded to the system log files, which you can view with the `show log messages` command. You can view the process names with the `show system processes` command. For more information about the `show system processes` command, see “Check That the Process Has Restarted” on page 42. For more detailed information about rebooting your system, see the *JUNOS System Basics and Services Command Reference*.

Restart a JUNOS Software Process

Purpose Restart a JUNOS software process when you need to recover from an error condition



NOTE: Never restart any of the software processes unless instructed to do so by a customer support engineer.

Steps To Take To restart a JUNOS software process, follow these steps:

1. Display Information about Software Processes on page 40
2. Restart a JUNOS Software Process on page 41
3. Check That the Process Has Restarted on page 42

Step 1: Display Information about Software Processes

Purpose Display information about software processes to begin diagnosing an error condition.

Action To display information about the software processes that are running on the router, use the following JUNOS CLI operational mode command:

```
user@host> show system processes extensive
```

Sample Output

```
user@host> show system processes extensive
last pid: 750; load averages: 0.00, 0.00, 0.00 up 0+00:58:50 18:34:17
52 processes: 1 running, 51 sleeping
Mem: 50M Active, 19M Inact, 38M Wired, 264K Cache, 86M Buf, 642M Free
Swap: 768M Total, 768M Free
```

PID	USERNAME	PRI	NICE	SIZE	RES	STATE	TIME	WCPU	CPU	COMMAND
546	root	10	0	9096K	1720K	nanslp	0:21	0.00%	0.00%	chassisd
685	root	2	0	12716K	3840K	kqread	0:01	0.00%	0.00%	rpd
553	root	2	0	8792K	1544K	select	0:01	0.00%	0.00%	mib2d
552	root	2	0	8632K	1556K	select	0:01	0.00%	0.00%	snmpd
563	root	2	0	9316K	1564K	select	0:00	0.00%	0.00%	kmd
564	root	2	0	7736K	948K	select	0:00	0.00%	0.00%	fud
131	root	10	0	770M	25568K	mfsidl	0:00	0.00%	0.00%	newfs
547	root	2	0	7732K	888K	select	0:00	0.00%	0.00%	alarmd
545	root	2	0	10292K	2268K	select	0:00	0.00%	0.00%	dcd
550	root	2	-12	1308K	692K	select	0:00	0.00%	0.00%	ntpd
1	root	10	0	816K	520K	wait	0:00	0.00%	0.00%	init
750	root	32	0	21716K	828K	RUN	0:00	0.00%	0.00%	top
560	root	2	0	8208K	1088K	select	0:00	0.00%	0.00%	rmopd
561	root	2	0	8188K	1156K	select	0:00	0.00%	0.00%	cosd
559	root	2	0	1632K	840K	select	0:00	0.00%	0.00%	ilmid

What It Means The sample output shows the central processing unit (CPU) utilization and lists the processes in order of CPU utilization.

Table 17 lists and describes the output fields included in the sample output for the `show processes extensive` command. The fields are listed in alphabetical order.

Table 17: Show System Processes Extensive Output Fields

Field	Description
COMMAND	Command that is running.
CPU	Raw (unweighted) CPU usage. The value of this field is used to sort the processes in the output.
last pid	Last process identifier assigned to the process.
load averages	Three load averages, followed by the current time.
Mem	Information about physical and virtual memory allocation.
NICE	UNIX “nice” value. The nice value allows a process to change its final scheduling priority.
PID	Process identifier.
PRI	Current kernel scheduling priority of the process. A lower number indicates a higher priority.
processes	Number of existing processes and the number of processes in each state (sleeping, running, starting, zombies, and stopped).
RES	Current amount of resident memory, in KB.
SIZE	Total size of the process (text, data, and stack), in KB.
STATE	Current state of the process (sleep, wait, run, idle, zombi, or stop).
Swap	Information about physical and virtual memory allocation.
USERNAME	Owner of the process.
WCPU	Weighted CPU usage.

For more details, see “Verify the Routing Engine CPU Memory” on page 179, and the *JUNOS Internet Software Protocols, Class of Service, Chassis, and Management Command Reference*.

Step 2: Restart a JUNOS Software Process

Action To restart a JUNOS software process, use the following JUNOS CLI operational mode command and include the process you wish to restart. For example:

```
user@host> restart routing
```

Sample Output user@host> restart routing
Routing protocol daemon started, pid 751

What It Means The sample output shows that the routing protocol daemon was restarted and the process identification (PID) was changed from 685 in the previous sample output to 751.

Table 18 lists and describes the options available for the **restart** command.

Table 18: Options to Restart a JUNOS Software Process

Option	Description
class-of-service	Restart the class-of-service process, which controls the router's class-of-service configuration.
gracefully	Restart the software process by sending the equivalent of a UNIX SIGTERM signal.
immediately	Immediately restart the process by sending the equivalent of a UNIX SIGKILL signal.
interface-control	Restart the interface process, which controls the router's physical interface devices and logical interfaces.
mib-process	Restart the Management Information Base (MIB) II process, which provides the router's MIB II agent.
network-access-service	Restart the network access process, which provides the router's Challenge Handshake Authentication Process (CHAP) authentication service.
remote-operations	Restart the remote operations process, which provides the ping and traceroute MIBs.
routing	Restart the routing protocol process, which controls the routing protocols that run on the router and maintains the routing tables.
sampling	Restart the sampling process, which performs packet sampling and cflowd export.
snmp	Restart the Simple Network Management Process (SNMP) process, which provides the router's SNMP master agent.
soft	Reread and reactivate the configuration without completely restarting the software processes. For example, Border Gateway Protocol (BGP) peers stay up and the routing table stays constant. This option is the equivalent of a UNIX SIGHUP signal; omitting this option is the equivalent of a UNIX SIGTERM (kill) operation.

Step 3: Check That the Process Has Restarted

Purpose After you have entered the `restart` command to restart a process, make sure that the process is up and running.

Action To check that a process has restarted, use the following JUNOS CLI operational mode command:

```
user@host> show system processes extensive
```

Sample Output 1

```
user@host> show system processes extensive
last pid: 750; load averages: 0.00, 0.00, 0.00 up 0+00:58:50 18:34:17
52 processes: 1 running, 51 sleeping
Mem: 50M Active, 19M Inact, 38M Wired, 264K Cache, 86M Buf, 642M Free
Swap: 768M Total, 768M Free

PID USERNAME PRI NICE  SIZE  RES STATE  TIME  WCPU   CPU COMMAND
546 root      10  0   9096K 1720K nanslp 0:21 0.00% 0.00% chassisd
685 root       2  0  12716K 3840K kqread 0:01 0.00% 0.00% rpd
553 root       2  0   8792K 1544K select 0:01 0.00% 0.00% mib2d
552 root       2  0   8632K 1556K select 0:01 0.00% 0.00% snmpd
563 root       2  0   9316K 1564K select 0:00 0.00% 0.00% kmd
```

```

564 root      2   0   7736K   948K select 0:00 0.00% 0.00% fud
131 root     10   0    770M 25568K mfsidl 0:00 0.00% 0.00% newfs
547 root      2   0   7732K   888K select 0:00 0.00% 0.00% alarmd
545 root      2   0  10292K  2268K select 0:00 0.00% 0.00% dcd
550 root      2 -12  1308K   692K select 0:00 0.00% 0.00% ntpd
   1 root     10   0    816K   520K wait   0:00 0.00% 0.00% init
750 root     32   0  21716K   828K RUN    0:00 0.00% 0.00% top
560 root      2   0   8208K  1088K select 0:00 0.00% 0.00% rmopd
561 root      2   0   8188K  1156K select 0:00 0.00% 0.00% cosd
559 root      2   0   1632K   840K select 0:00 0.00% 0.00% ilmid

```

Sample Output 2 user@host> **show system processes extensive**
last pid: 758; load averages: 0.00, 0.00, 0.00 up 0+01:01:48 18:37:15
52 processes: 1 running, 51 sleeping
Mem: 51M Active, 19M Inact, 38M Wired, 156K Cache, 86M Buf, 642M Free
Swap: 768M Total, 768M Free

PID	USERNAME	PRI	NICE	SIZE	RES	STATE	TIME	WCPU	CPU	COMMAND
546	root	10	0	9096K	1720K	nanslp	0:22	0.05%	0.05%	chassisd
553	root	2	0	8792K	1544K	select	0:01	0.00%	0.00%	mib2d
552	root	2	0	8632K	1556K	select	0:01	0.00%	0.00%	snmpd
563	root	2	0	9316K	1564K	select	0:00	0.00%	0.00%	kmd
564	root	2	0	7736K	948K	select	0:00	0.00%	0.00%	fud
131	root	10	0	770M	25568K	mfsidl	0:00	0.00%	0.00%	newfs
547	root	2	0	7732K	888K	select	0:00	0.00%	0.00%	alarmd
545	root	2	0	10292K	2268K	select	0:00	0.00%	0.00%	dcd
1	root	10	0	816K	520K	wait	0:00	0.00%	0.00%	init
550	root	2	-12	1308K	692K	select	0:00	0.00%	0.00%	ntpd
758	root	32	0	21716K	832K	RUN	0:00	0.00%	0.00%	top
560	root	2	0	8208K	1088K	select	0:00	0.00%	0.00%	rmopd
561	root	2	0	8188K	1156K	select	0:00	0.00%	0.00%	cosd
559	root	2	0	1632K	840K	select	0:00	0.00%	0.00%	ilmid
573	lab	2	0	7480K	2580K	select	0:00	0.00%	0.00%	cli
751	root	2	0	12716K	3944K	kqread	0:00	0.00%	0.00%	rpdp
558	root	2	20	8708K	1880K	select	0:00	0.00%	0.00%	sampled
555	root	2	0	1856K	932K	select	0:00	0.00%	0.00%	vrripd
686	root	2	0	7808K	940K	select	0:00	0.00%	0.00%	apspd

What It Means The sample output shows that the routing protocol process (rpdp) was restarted because the process identifier (PID) of the process was renamed from 685, as shown in the Sample Output 1, to 751 as shown in Sample Output 2.

Chapter 5

Display JUNOS Software Information

This chapter describes how to display the hostname and version information for the JUNOS software running on a router. (See Table 19.)

Table 19: Checklist for Displaying JUNOS Software Information

Tasks for Displaying JUNOS Software Information and Status	Command or Action
Display JUNOS Software Information on page 46	show version
Display Version Information for JUNOS Software Packages on page 47	show version brief

Display JUNOS Software Information

Purpose Display JUNOS software information and status to determine if the version of JUNOS software that you are running supports particular features or hardware. You can also determine whether particular software bugs will affect your version of JUNOS software.

Action To display JUNOS software information, use the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> show version
```

Sample Output

```
user@host> show version
Hostname: my-router.net
Model: m160
JUNOS Base OS boot [5.5R2.3]
JUNOS Base OS Software Suite [5.5R2.3]
JUNOS Kernel Software Suite [5.5R2.3]
JUNOS Packet Forwarding Engine Support [5.5R2.3]
JUNOS Routing Software Suite [5.5R2.3]
JUNOS Online Documentation [5.5R2.3]
JUNOS Crypto Software Suite [5.5R2.3]
KERNEL 5.5R2.3 #0 built by builder on 2002-11-21 22:56:20 UTC
MGD release 5.5R2.3 built by builder on 2002-11-21 22:36:05 UTC
CLI release 5.5R2.3 built by builder on 2002-11-21 22:33:44 UTC
CHASSISD release 5.5R2.3 built by builder on 2002-11-21 22:32:10 UTC
DCD release 5.5R2.3 built by builder on 2002-11-21 22:30:06 UTC
RPD release 5.5R2.3 built by builder on 2002-11-21 22:37:08 UTC
SNMPD release 5.5R2.3 built by builder on 2002-11-21 22:43:14 UTC
MIB2D release 5.5R2.3 built by builder on 2002-11-21 22:36:10 UTC
APSD release 5.5R2.3 built by builder on 2002-11-21 22:32:07 UTC
VRRPD release 5.5R2.3 built by builder on 2002-11-21 22:43:26 UTC
ALARM release 5.5R2.3 built by builder on 2002-11-21 22:32:01 UTC
PFED release 5.5R2.3 built by builder on 2002-11-21 22:36:53 UTC
CRAFTD release 5.5R2.3 built by builder on 2002-11-21 22:33:59 UTC
SAMPLED release 5.5R2.3 built by builder on 2002-11-21 22:43:01 UTC
ILMID release 5.5R2.3 built by builder on 2002-11-21 22:35:17 UTC
RMOPD release 5.5R2.3 built by builder on 2002-11-21 22:37:01 UTC
COSD release 5.5R2.3 built by builder on 2002-11-21 22:33:50 UTC
KMD release 5.5R2.3 built by builder on 2002-11-21 22:35:29 UTC
FSAD release 5.5R2.3 built by builder on 2002-11-21 22:34:14 UTC
SERVICED release 5.5R2.3 built by builder on 2002-11-21 22:43:07 UTC
IRSD release 5.5R2.3 built by builder on 2002-11-21 22:35:21 UTC
NASD release 5.5R2.3 built by builder on 2002-11-21 22:36:47 UTC
FUD release 5.5R2.3 built by builder on 2002-11-21 22:34:17 UTC
PPMD release 5.5R2.3 built by builder on 2002-11-21 22:36:58 UTC
LMPD release 5.5R2.3 built by builder on 2002-11-21 22:36:01 UTC
RTSPD release 5.5R2.3 built by builder on 2002-11-21 22:42:58 UTC
SMARTD release 5.5R2.3 built by builder on 2002-11-21 22:47:50 UTC
jkernel-dd release 5.5R2.3 built by builder on 2002-11-21 22:27:20 UTC
jroute-dd release 5.5R2.3 built by builder on 2002-11-21 22:27:34 UTC
jcrypto-dd release 5.5R2.3 built by builder on 2002-11-21 22:27:46 UTC
```

What It Means The sample output shows the hostname, the version information for the JUNOS software packages installed on the machine, and the version information for each software process.

Display Version Information for JUNOS Software Packages

Purpose Display version information for JUNOS software packages to determine if they support particular features or hardware. You can also determine whether particular software bugs will affect your version of JUNOS software.

Action To display brief information and status for the kernel and Packet Forwarding Engine, use the following CLI operational mode command:

```
user@host> show version brief
```

Sample Output The following sample output is for worldwide nonencrypted JUNOS software:

```
user@host> show version brief
Hostname: my-router.net
Model: m10
JUNOS Base OS boot [5.5R2.3]
JUNOS Base OS Software Suite [5.5R2.3]
JUNOS Kernel Software Suite [5.5R2.3]
JUNOS Packet Forwarding Engine Support [5.5R2.3]
JUNOS Routing Software Suite [5.5R2.3]
JUNOS Online Documentation [5.5R2.3]
```

The following sample output is for Canada and USA encrypted JUNOS software:

```
user@host> show version brief
Hostname: my-router.net
Model: m10
JUNOS Base OS boot [5.5R2.3]
JUNOS Base OS Software Suite [5.5R2.3]
JUNOS Kernel Software Suite [5.5R2.3]
JUNOS Packet Forwarding Engine Support [5.5R2.3]
JUNOS Routing Software Suite [5.5R2.3]
JUNOS Online Documentation [5.5R2.3]
JUNOS Crypto Software Suite [5.5R2.3]
```

What It Means The sample output shows version information for the JUNOS software packages installed on the router. If the **JUNOS Crypto Software Suite** is listed the router has Canada and USA encrypted JUNOS software. If the **JUNOS Crypto Software Suite** is not listed, the router is running worldwide nonencrypted JUNOS software.

Chapter 6

Check Router Configuration

This chapter describes how to check the configuration on your router. (See Table 20.)

Table 20: Checklist for Checking the Router Configuration

Check Router Configuration Tasks	Command or Action
Display the Current Active Router Configuration on page 50	show configuration
Display a Specific Configuration Hierarchy on page 54	show configuration <i>statement-path</i>
Display Additional Information about the Configuration on page 54	[edit] show < <i>hierarchy-level</i> > display detail

Display the Current Active Router Configuration

Action To display the current, active router configuration, use the following command-line interface (CLI) operational mode command:

```
user@host> show configuration
```

Sample Output

```
user@host> show configuration
version "5.6I0";
groups {
  global {
    system {
      host-name potter;
      domain-name harry.potter.net;
      domain-search [ harry.potter.net potter.net hryptr.net ];
      backup-router 10.110.12.254;
      time-zone America/Los_Angeles;
      debugger-on-panic;
      debugger-on-break;
      dump-on-panic;
      authentication-order [ tacplus password radius ];
      root-authentication {
        encrypted-password "$1$0Ff5.$I7.kUgMmx/4WKwUAG"; # SECRET-DATA
      }
      name-server {
        172.17.28.101;
        172.17.28.100;
      }
      radius-server {
        10.168.5.73 {
          secret "$9$Nd-YoDjq.PT4oZjik5T369pBIhS1L7dC"; # SECRET-DATA
          timeout 5;
          retry 3;
        }
      }
      tacplus-server {
        10.168.5.73 {
          secret "$9$.539IRSM8701lMX-2gqmfTz6"; # SECRET-DATA
          timeout 15;
          single-connection;
        }
      }
      login {
        class superuser-local {
          permissions all;
        }
        class wheel {
          permissions [ admin clear field floppy interface maintenance
network reset routing shell snmp system trace view ];
        }
        class readonly {
          permissions [ interface network routing system trace view ];
        }
        user rpe {
          uid 1230;
          class superuser;
          shell csh;
          authentication {
            encrypted-password FN5oyk/qZ07F2; # SECRET-DATA
          }
        }
      }
    }
  }
}
```

[...output truncated...]

```

    }
  }
}
static-host-mapping {
  crater sysid 0102.5524.5045;
  badlands sysid 0102.5524.5046;
  [...output truncated...]
}
services {
  finger;
  ftp;
  rlogin;
  rsh;
  ssh;
telnet;
}
syslog {
  user * {
    any emergency;
  }
  host log {
    any notice;
    pfe info;
    interactive-commands any;
  }
  file messages {
    any notice;
    authorization info;
    pfe info;
    archive world-readable;
  }
  file security {
    interactive-commands any;
    archive world-readable;
  }
  file white_bx {
    daemon notice;
    archive size 40m world-readable;
  }
}
processes {
  routing enable;
  snmp enable;
  tnp-process enable;
  ntp enable;
  inet-process enable;
  mib-process enable;
  management enable;
  watchdog enable;
}
ntp {
  boot-server ntp.juniper.net;
  server 172.17.27.46;
}
}
chassis {
  dump-on-panic;
}
snmp {
  location "Systest lab";
  contact "Brian Matheson";
  interface fxp0.0;
  community public {

```

```

        authorization read-only;
    }
    community private {
        authorization read-write;
    }
}
routing-options {
    static {
        /* corporate and alpha net */
        route 172.16.0.0/12 {
            next-hop 10.168.14.254;
            retain;
            no-readvertise; [...output truncated...]
        }
    }
}
}
re1;
}
apply-groups [ global re0 re1 ];
chassis {
    fpc 0 {
        pic 0 {
            mlfr-uni-nni-bundles 4;
        }
    }
}
interfaces {
    ls-0/0/0:0 {
        encapsulation multilink-frame-relay-uni-nni;
        unit 0 {
            dlci 100;
            family inet {
                address 10.53.99.2/32 {
                    destination 10.53.99.1;
                }
            }
        }
    }
    ct3-0/1/0 {
        partition 1 interface-type t1;
        partition 2 interface-type t1;
        partition 3 interface-type t1;
        partition 4 interface-type t1;
    }
    t1-0/1/0:1 {
        encapsulation multilink-frame-relay-uni-nni;
        unit 0 {
            family mlfr-uni-nni {
                bundle ls-0/0/0:0;
            }
        }
    }
}
routing-options {
    static {
        route 10.1.1.0/24 next-hop 10.53.99.1;
    }
    autonomous-system 69;
    forwarding-table {
        export pplb;
    }
}
}

```

```

protocols {
  bgp {
    disable;
    group int {
      type internal;
      neighbor 10.255.14.30;
      [...output truncated...] }
    }
  isis {
    disable;
    interface all {
      level 1 disable;
    }
    interface fxp0.0 {
      disable;
    }
  }
  inactive: ospf {
    traffic-engineering;
    reference-bandwidth 4g;
    area 0.0.0.0 {
      interface all;
      interface fxp0.0 {
        disable;
      }
    }
  }
}
policy-options {
  policy-statement pplb {
    then {
      load-balance per-packet;
    }
  }
}
[...Output truncated...]

```

What It Means The sample output shows the current, active configuration for the router. When displaying the configuration, the CLI indents each subordinate hierarchy level, inserts braces to indicate the beginning and end of each hierarchy level, and places semicolons at the end of statements that are at the lowest level of the hierarchy.

The configuration statements appear in a fixed order. Interfaces appear alphabetically by type, and then in numerical order by slot number, Physical Interface Card (PIC) number, and port number.

Display a Specific Configuration Hierarchy

Action To view a specific configuration hierarchy, use the following CLI operational mode command;

```
user@host> show configuration statement-path
```

Sample Output

```
user@host> show configuration protocols bgp
group ebgp {
  type external;
  peer-as 65001;
  neighbor 10.168.20.1;
}
```

What It Means The sample output shows the active configuration under the [protocol bgp] hierarchy level.

Display Additional Information about the Configuration

Purpose You can display additional information when you are not sure of the meaning of configuration statements and what permission bits are required to add and modify them.

Action To display additional information about the entire configuration, use the following CLI configuration mode command:

```
user@host# show | display detail
```

To display additional information about a specific hierarchy, use the following CLI configuration mode command:

```
user@host# show <hierarchy-level> | display detail
```

Sample Output The following sample output is for the first command. The output for a particular hierarchy appears similar to its section in this sample output.

```
user@host> edit
user@host# show | display detail
##
## version: Software version information
## require: system
##
version "3.4R1 [tlim]";
system {
##
## host-name: Host name for this router
## match: ^[:alnum:]._-]+$
## require: system
##
host-name router-name;
##
## domain-name: Domain name for this router
## match: ^[:alnum:]._-]+$
## require: system
##
domain-name isp.net;
```

```

##
## backup-router: Address of router to use while booting
##
backup-router 10.168.100.1;
root-authentication {
    ##
    ## encrypted-password: Crypted password string
    ##
    encrypted-password "$1$BYJQE$/ocQof8pmcm7MSGK0"; # SECRET-DATA
}
##
## name-server: DNS name servers
## require: system
##
name-server {
    ##
    ## name-server: DNS name server address
    ##
    208.197.1.0;
}
login {
    ##
    ## class: User name (login)
    ## match: ^[:alnum:][_~]+$
    ##
    class superuser {
        ##
        ## permissions: Set of permitted operation categories
        ##
        permissions all;
    }
    ...
    ##
    ## services: System services
    ## require: system
    ##
    services {
        ## services: Service name
        ##
        ftp;
        ##
        ## services: Service name
        ##
        telnet;
        ##
    }
}
syslog {
    ##
    ## file-name: File to record logging data
    ##
    file messages {
        ##
        ## Facility type
        ## Level name
        ##
        any notice;
        ##
        ## Facility type
        ## Level name
        ##
    }
    authorization info;
}
}

```

```

}
chassis {
  alarm {
    sonet {
      ##
      ## lol: Loss of light
      ## alias: loss-of-light
      ##
      lol red;
    }
  }
}
interfaces {
  ##
  ## Interface name
  ##
  at-2/1/1 {
    atm-options {
      ##
      ## vpi: Virtual path index
      ## range: 0 .. 255
      ## maximum-vcs: Maximum number of virtual circuits on this VP
      ##
      vpi 0 maximum-vcs 512;
    }
    ##
    ## unit: Logical unit number
    ## range: 0 .. 16384
    ##
    unit 0 {
      ##
      ## vci: ATM point-to-point virtual circuit identifier ([vpi.]vci)
      ## match: ^([[:digit:]]+.){0,1}[[:digit:]]+$
      ##
      vci 0.128;
    }
  }
  ...
}

```

What It Means The sample output shows additional information that includes the help string explaining each configuration statement, and the permission bits required to add and modify the configuration statement.

Chapter 7

Upgrade JUNOS Software

As new features become available or as software problems are fixed, you might periodically upgrade the router software. (See Table 21.)

Table 21: Checklist for Upgrading JUNOS Software

Upgrade JUNOS Software Tasks	Command or Action
Before You Upgrade JUNOS Software on page 58	
1. Log the Software Version Information on page 59	show version save <i>filename</i>
2. Log the Hardware Version Information on page 60	show chassis hardware save <i>filename</i>
3. Log the Active Configuration on page 61	show configuration save <i>filename</i>
4. Log the Interfaces on the Router on page 61	show interface terse save <i>filename</i>
5. Log the BGP, IS-IS, and OSPF Adjacency Information on page 62	show bgp summary save <i>filename</i> show isis adjacency brief save <i>filename</i> show ospf neighbor brief save <i>filename</i>
6. Log the System Storage Information on page 63	show system storage save <i>filename</i>
7. Back Up the Currently Running and Active File System on page 63	request system snapshot
8. Download JUNOS Software on page 64	http://www.juniper.net/support
Upgrade JUNOS Software on page 69	
1. Copy JUNOS Software to the Router on page 69	file copy <code>ftp://username:prompt@ftp.hostname.net/jbundle-package-name/var/tmp/jbundle-package-name</code>
2. Add New Software on page 69	request system software add/var/tmp/jbundle-package-name
3. Start the New Software on page 70	request system reboot
After You Upgrade JUNOS Internet Software on page 71	
1. Compare Information Logged Before and After the Upgrade on page 71	show version save <i>filename</i> show chassis hardware save <i>filename</i> show configuration save <i>filename</i> show interface terse save <i>filename</i> show bgp summary save <i>filename</i> show isis adjacency brief save <i>filename</i> show ospf neighbor brief save <i>filename</i> show system storage save <i>filename</i>
2. Back Up the New Software on page 71	request system snapshot

Before You Upgrade JUNOS Software

Purpose Before you upgrade the JUNOS software, it is important to log information about the existing system so that after the upgrade you can compare the same information to verify that all components are installed and working as expected. Also, during the process of logging information, you might find an existing problem that you did not know about and might have thought was caused by the upgrade.

Steps to Take To log important information about your system, follow these steps:

1. Log the Software Version Information on page 59
2. Log the Hardware Version Information on page 60
3. Log the Active Configuration on page 61
4. Log the Interfaces on the Router on page 61
5. Log the BGP, IS-IS, and OSPF Adjacency Information on page 62
6. Log the System Storage Information on page 63
7. Back Up the Currently Running and Active File System on page 63
8. Download JUNOS Software on page 64

In all the logging steps, you can use your terminal program to save the output from the commands, or use the **save** command to redirect the output to an external file.

To save the output to a file on another machine, use the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> command | save filename
```

By default, the file is placed in your home directory on the router. To redirect the output to a file on another machine, change the filename to include the path to that machine and file. For information about how you can specify the filename, see the *JUNOS System Basics and Services Command Reference*.

The following example stores the output of the **show version** command in a file:

```
user@host> show version | save filename
Wrote 1143 lines of output to 'filename'
```

Step 1: Log the Software Version Information

Action To log the JUNOS software version information, use the following JUNOS CLI operational mode command:

```
user@host> show version | save filename
```

Sample Output user@host> show version | save test
Wrote 39 lines of output to 'test'

```
user@host> show version
Hostname: my-router.net
Model: m10
JUNOS Base OS boot [5.0R5]
JUNOS Base OS Software Suite [5.0R5]
JUNOS Kernel Software Suite [5.0R5]
JUNOS Routing Software Suite [5.0R5]
JUNOS Packet Forwarding Engine Support [5.0R5]
JUNOS Crypto Software Suite [5.0R5]
JUNOS Online Documentation [5.0R5]
KERNEL 5.0R5 #0 built by builder on 2002-03-02 05:10:28 UTC
MGD release 5.0R5 built by builder on 2002-03-02 04:45:32 UTC
CLI release 5.0R5 built by builder on 2002-03-02 04:44:22 UTC
CHASSISD release 5.0R5 built by builder on 2002-03-02 04:43:37 UTC
DCD release 5.0R5 built by builder on 2002-03-02 04:42:47 UTC
RPD release 5.0R5 built by builder on 2002-03-02 04:46:17 UTC
SNMPD release 5.0R5 built by builder on 2002-03-02 04:52:26 UTC
MIB2D release 5.0R5 built by builder on 2002-03-02 04:45:37 UTC
APSD release 5.0R5 built by builder on 2002-03-02 04:43:31 UTC
VRRPD release 5.0R5 built by builder on 2002-03-02 04:52:34 UTC
ALARMD release 5.0R5 built by builder on 2002-03-02 04:43:24 UTC
PFED release 5.0R5 built by builder on 2002-03-02 04:46:06 UTC
CRAFTD release 5.0R5 built by builder on 2002-03-02 04:44:30 UTC
SAMPLED release 5.0R5 built by builder on 2002-03-02 04:52:20 UTC
ILMID release 5.0R5 built by builder on 2002-03-02 04:45:21 UTC
BPRELAYD release 5.0R5 built by builder on 2002-03-02 04:42:41 UTC
RMOPD release 5.0R5 built by builder on 2002-03-02 04:46:11 UTC
jkernel-dd release 5.0R5 built by builder on 2002-03-02 04:41:07 UTC
jroute-dd release 5.0R5 built by builder on 2002-03-02 04:41:21 UTC
jdocs-dd release 5.0R5 built by builder on 2002-03-02 04:39:11 UTC
```

What It Means The sample output shows the hostname, router model, and the different JUNOS software packages, processes, and documents.

Step 2: Log the Hardware Version Information

Action To log the router chassis hardware version information, use the following JUNOS CLI operational mode command:

```
user@host> show chassis hardware | save filename
```

Sample Output The output for the M-series routers varies depending on the chassis components of each router. All routers have a chassis, midplanes or backplanes, power supplies, and Flexible PIC Concentrators (FPCs). Refer to the hardware guides for information about the different chassis components.

```
user@host> show chassis hardware | save test
Wrote 43 lines of output to 'test'
```

```
user@host> show chassis hardware
```

Item	Version	Part number	Serial number	Description
Chassis			101	M160
Midplane	REV 02	710-001245	S/N AB4107	
FPM CMB	REV 01	710-001642	S/N AA2911	
FPM Display	REV 01	710-001647	S/N AA2999	
CIP	REV 02	710-001593	S/N AA9563	
PEM 0	Rev 01	740-001243	S/N KJ35769	DC
PEM 1	Rev 01	740-001243	S/N KJ35765	DC
PCG 0	REV 01	710-001568	S/N AA9794	
PCG 1	REV 01	710-001568	S/N AA9804	
Host 1			da000004f8d57001	teknor
MCS 1	REV 03	710-001226	S/N AA9777	
SFM 0 SPP	REV 04	710-001228	S/N AA2975	
SFM 0 SPR	REV 02	710-001224	S/N AA9838	Internet Processor I
SFM 1 SPP	REV 04	710-001228	S/N AA2860	
SFM 1 SPR	REV 01	710-001224	S/N AB0139	Internet Processor I
FPC 0	REV 03	710-001255	S/N AA9806	FPC Type 1
CPU	REV 02	710-001217	S/N AA9590	
PIC 1	REV 05	750-000616	S/N AA1527	1x OC-12 ATM, MM
PIC 2	REV 05	750-000616	S/N AA1535	1x OC-12 ATM, MM
PIC 3	REV 01	750-000616	S/N AA1519	1x OC-12 ATM, MM
FPC 1	REV 02	710-001611	S/N AA9523	FPC Type 2
CPU	REV 02	710-001217	S/N AA9571	
PIC 0	REV 03	750-001900	S/N AA9626	1x STM-16 SDH, SMIR
PIC 1	REV 01	710-002381	S/N AD3633	2x G/E, 1000 BASE-SX
FPC 2				FPC Type OC192
CPU	REV 03	710-001217	S/N AB3329	
PIC 0	REV 01			1x OC-192 SM SR-2

What It Means The sample output shows the hardware inventory for an M160 router with a chassis serial number of 101. For each component, the output shows the version number, part number, serial number, and description.

Step 3: Log the Active Configuration

Action To log the active configuration on the router, use the following JUNOS CLI operational mode command:

```
user@host> show configuration | save filename
```

Sample Output user@host> show configuration | save test
Wrote 4076 lines of output to 'test'

```
user@host> show configuration
system {
  host-name lab8;
  domain-name juniper.net;
  backup-router 10.1.1.254;
  time-zone America/Los_Angeles;
  default-address-selection;
  dump-on-panic;
  name-server {
[...Output truncated...]
```

What It Means The sample output shows the configuration currently running on the router, which is the last committed configuration.

Step 4: Log the Interfaces on the Router

Action To log the interfaces on the router, use the following JUNOS CLI operational mode command:

```
user@host> show interface terse | save filename
```

Sample Output user@host> show interface terse | save test
Wrote 81 lines of output to 'test'

```
user@host> show interface terse
Interface      Admin Link Proto Local          Remote
at-1/3/0       up    up
at-1/3/0.0     up    up    inet  1.0.0.1        --> 1.0.0.2
               iso
fxp0           up    up
fxp0.0         up    up    inet  10.168.5.59/24
gre            down  up
ipip           down  up
lo0            up    up
lo0.0          up    up    inet  127.0.0.1        --> 0/0
               iso
47.0005.80ff.f800.0000.0108.0001.1921.6800.5059.00
so-1/2/0       up    down
so-1/2/1       down  down
so-1/2/2       down  down
so-1/2/3       down  down
so-2/0/0       up    up
so-2/0/0.0     up    up    inet  1.2.3.4          --> 1.2.3.5
               iso
[...Output truncated...]
```

What It Means The sample output shows summary information about the physical and logical interfaces on the router.

Step 5: Log the BGP, IS-IS, and OSPF Adjacency Information

Purpose The following commands log useful information about the Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (IS-IS), and Open Shortest Path First (OSPF) protocols. If you have other protocols installed, such as Multiprotocol Label Switching (MPLS), Resource Reservation Protocol (RSVP), or Protocol Independent Multicast (PIM), you also might log summary information for them.

Action To log protocol peer information, use the following JUNOS CLI operational mode commands:

```
user@host> show bgp summary | save filename
user@host> show isis adjacency brief | save filename
user@host> show ospf neighbor brief | save filename
```

Sample Output 1 user@host> show bgp summary | save test
Wrote 45 lines of output to 'test'

```
user@host> show bgp summary
Groups: 1 Peers: 1 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State   Pending
inet.0         4          4          0          0        0      0        0
Peer           AS      InPkt   OutPkt   OutQ   Flaps Last Up/Dwn
State|#Active/Received/Damped...
9.9.3.1         2      2627    2628     0     0   21:50:12 4/4/0
0/0/0
```

Sample Output 2 user@host> show isis adjacency brief | save test
Wrote 10 lines of output to 'test'

```
user@host> show isis adjacency brief
IS-IS adjacency database:
Interface System L State Hold (secs) SNPA
so-1/0/0.0 1921.6800.5067 2 Up 13
so-1/1/0.0 1921.6800.5067 2 Up 25
so-1/2/0.0 1921.6800.5067 2 Up 20
so-1/3/0.0 1921.6800.5067 2 Up 19
so-2/0/0.0 1921.6800.5066 2 Up 19
so-2/1/0.0 1921.6800.5066 2 Up 17
so-2/2/0.0 1921.6800.5066 2 Up 20
so-2/3/0.0 1921.6800.5066 2 Up 20
so-5/0/0.0 ranier 2 Up 17
```

Sample Output 3 user@host> show ospf neighbor brief | save test
Wrote 10 lines of output to 'test'

```
user@host> show ospf neighbor brief
Address      Intf      State      ID          Pri  Dead
10.168.254.225 fxp3.0    2Way       10.250.240.32 128  36
10.168.254.230 fxp3.0    Full       10.250.240.8  128  38
10.168.254.229 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
```

What It Means Sample output 1 displays summary information about BGP and its neighbors. Sample output 2 displays information about IS-IS neighbors. Sample output 3 displays information about all OSPF neighbors.

Step 6: Log the System Storage Information

Action To log system storage statistics for the amount of free disk space in the router's file system, use the following JUNOS CLI operational mode command:

```
user@host> show system storage | save filename
```

Sample Output user@host> show system storage | save test
Wrote 14 lines of output to 'test'

```
user@host> show system storage
Filesystem 1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a    65687    26700    33733    44%      /
devfs          16         16         0    100%    /dev/
/dev/vn1       9310     9310         0    100%    /packages/mnt/jbase
/dev/vn2       8442     8442         0    100%    /packages/mnt/jkernel-5.0R5.1
/dev/vn3      11486    11486         0    100%    /packages/mnt/jpfe-5.0R5.1
/dev/vn4       5742     5742         0    100%    /packages/mnt/jroute-5.0R5.1
/dev/vn5       1488     1488         0    100%    /packages/mnt/jcrypto-5.0R5.1
/dev/vn6        792         792         0    100%    /packages/mnt/jdocs-5.0R5.1
mfs:2373    1015815         3    934547     0%      /tmp
/dev/ad0s1e    25263         11    23231     0%      /config
procfs         4          4         0    100%    /proc
/dev/ad1s1f   9825963   1811085   7228801    20%     /var
```

What It Means The sample output shows statistics about the amount of free disk space in the router's file system. Values are displayed in 1024-byte (1-KB) blocks.

Step 7: Back Up the Currently Running and Active File System

Action To back up the currently running and active file system so that you can recover to a known, stable environment in case there is a problem during the upgrade, use the following JUNOS CLI operational mode command:

```
user@host> request system snapshot
```

Sample Output user@host> request system snapshot
umount: /altroot: not currently mounted
Copying / to /altroot.. (this may take a few minutes)
umount: /altconfig: not currently mounted
Copying /config to /altconfig.. (this may take a few minutes)

The following filesystems were archived: / /config

What It Means The root file system is backed up to /altroot, and /config is backed up to /altconfig. The root and /config file systems are on the router's internal flash drive, and the /altroot and /altconfig file systems are on the router's hard drive.



NOTE: After you issue the `request system snapshot` command, you cannot return to the previous version of the software because the running and backup copies of the software are identical.

Step 8: Download JUNOS Software



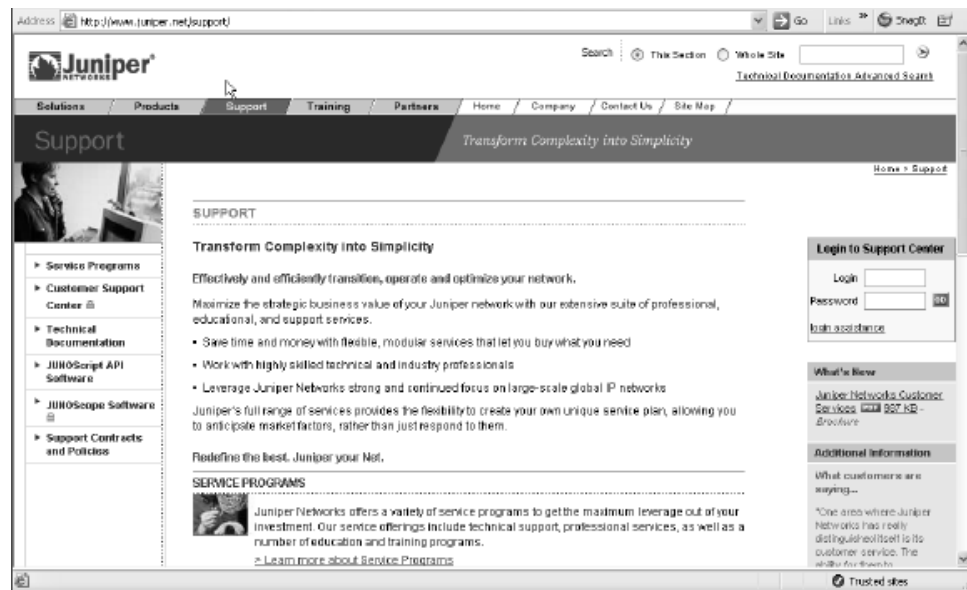
NOTE: To download the JUNOS software packages, you must have a service contract and an access account. Try to download the software packages a few days before you intend to install them, as you may need to verify your service contract and access account. If you need help obtaining an account, contact your Juniper Networks sales representative or send an e-mail to logistics@juniper.net.

Action To download the software packages from the Juniper Networks Support Web site, follow these steps:

1. Enter the following site address:

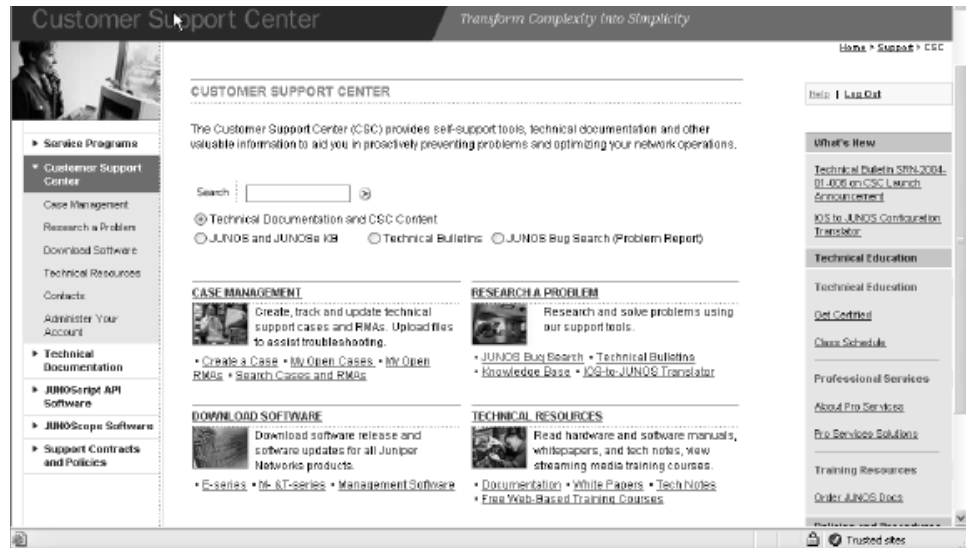
<http://www.juniper.net/support>

The following screen appears.

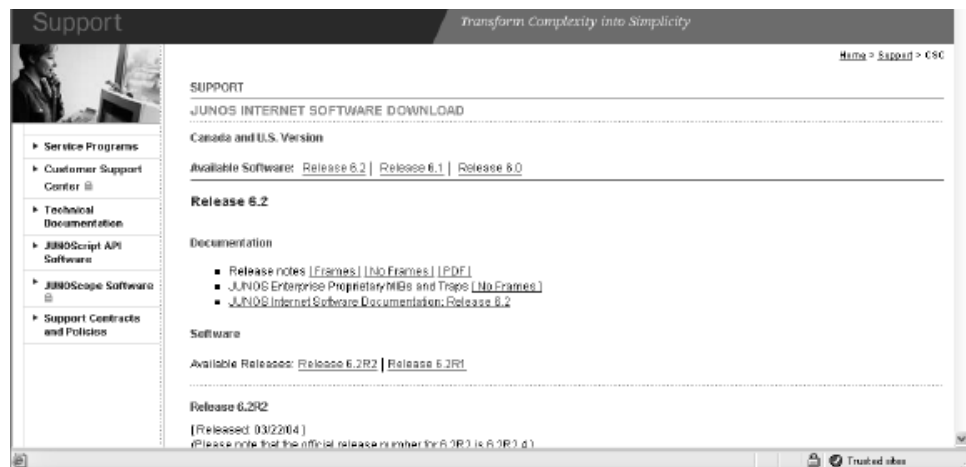


2. In Login to Support Center, enter your login and password.

The customer support center screen appears.

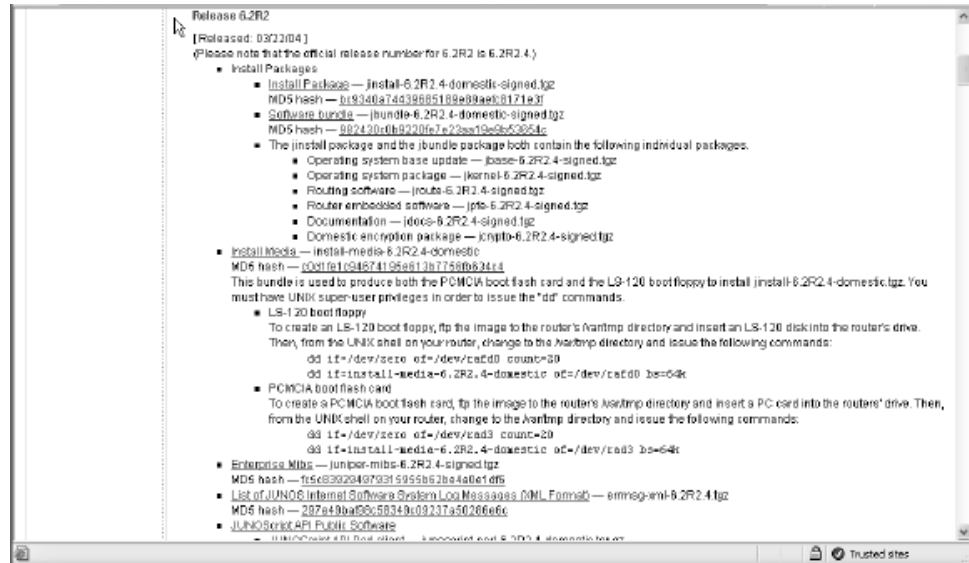


- From Download Software, select the M- & T-series link. The Software Download screen appears.

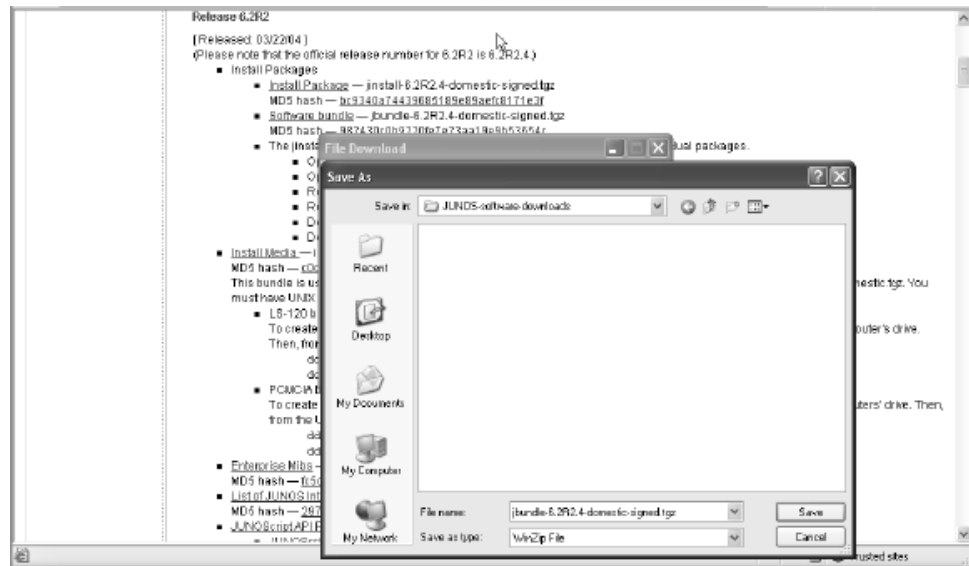


- From Available Releases, click the software release you want.

The Software to Download screen appears.



5. Click the software bundle you want to download. The Save As screen appears.



6. Click Save to download the software packages. Download the software packages to a server, not to the router.

What It Means Each JUNOS software release consists of the following software packages:

- **jbase**—Additions to the operating system
- **jkernel**—Operating system package
- **jroute**—Software that runs on the Routing Engine
- **jpfe**—Software that runs on the Packet Forwarding Engine
- **jdocs**—Documentation for the software
- **jcrypto**—Security software (in domestic software only)

The packages are also grouped together in a bundle, called **jbundle**. Normally, you use the bundle to upgrade all of the software packages at the same time.



NOTE: If you are upgrading to Release 5.0 from Release 4.x or downgrading from Release 5.0 to Release 4.x, use the **jinstall** package. Otherwise, use the **jbundle** package to upgrade to a new release.

Downgrading from Release 5.0 to Release 4.x can be a two-step process. For more information, see *JUNOS Internet Software System Basics Configuration Guide*.

You also can upgrade the software packages individually but this is not recommended. When upgrading to a new release, you must install the entire bundle; do not upgrade packages individually unless instructed to do so by the Juniper Networks Technical Assistance Center (JTAC).

Two sets of JUNOS software packages are provided: one for customers in the United States and Canada, and another for worldwide customers. The worldwide version does not include any capabilities that provide encryption of data leaving the router. Otherwise, the two packages are identical.

Upgrade JUNOS Software

Purpose As new features become available or as software problems are fixed, you might periodically upgrade the router software.

Steps To Take To upgrade JUNOS software, follow these steps:

1. Copy JUNOS Software to the Router on page 69
2. Add New Software on page 69
3. Start the New Software on page 70

Step 1: Copy JUNOS Software to the Router

Action Copy the software packages from the server to the router. We recommend that you copy them to the `/var/tmp` directory, which is on the rotating medium (hard disk) and is a large file system. Use the following CLI command:

```
user@host> file copy
ftp://username:prompt@ftp.hostname.net/jbundle-package-name
/var/tmp/jbundle-package-name
```

Step 2: Add New Software

Action To add new software packages, use the following JUNOS CLI operational mode command:

```
user@host> request system software add /var/tmp/jbundle-package-name
```

package-name is the full URL to the file and *release-number* is the major software release number; for example, 4.2R1. Before the new software is added, the existing software is automatically deleted.



NOTE: Even though you are adding the new software, the changes do not take effect until the router has completed rebooting.

Sample Output

```
user@host> request system software add /var/tmp/jinstall-5.2R2.3-domestic.tgz
Installing package '/var/tmp/jinstall-5.2R2.3-domestic.tgz'
Auto-deleting old jroute...
Auto-deleting old jdocs...
Auto-deleting old jpfe...
Auto-deleting old jkernel...
Adding JUNOS base software 5.2R2.3
Adding jkernel...
Adding jpfe...
Adding jdocs...
Adding jroute...
NOTICE: uncommitted changes have been saved in
/var/db/config/juniper.conf.pre-install
Saving package file in /var/sw/pkg/jinstall-5.2R2.3-domestic.tgz
```

Step 3: Start the New Software

Purpose After you have added new software packages, you must reboot the router for the new software to take effect.

Action To reboot the router to complete the upgrade, use the following JUNOS CLI operational mode command:

```
user@host> request system reboot
```

After You Upgrade JUNOS Internet Software

Steps To Take To verify that the new version of JUNOS software is running as expected after the upgrade, follow these steps:

1. Compare Information Logged Before and After the Upgrade on page 71
2. Back Up the New Software on page 71

Step 1: Compare Information Logged Before and After the Upgrade

Purpose Compare the operation of the system before and after the upgrade to ensure that everything is working as expected.

Action To obtain system information, use the following JUNOS CLI operational mode commands:

```
user@host> show version
user@host> show chassis hardware
user@host> show configuration
user@host> show interface terse
user@host> show bgp summary
user@host> show isis adjacency brief
user@host> show ospf neighbor brief
user@host> show system storage
```

Compare the information from these commands with the information you logged before the upgrade.

Step 2: Back Up the New Software

Purpose After a week or so, when you are satisfied that the new software is running successfully, we recommend that you back up the upgraded software.

Action To back up the upgraded software, use the following JUNOS CLI operational mode command:

```
user@host> request system snapshot
```

Sample Output The root file system is backed up to /altroot, and /config is backed up to /altconfig. The root and /config file systems are on the router's internal flash drive, and the /altroot and /altconfig file systems are on the router's hard drive.



NOTE: After you issue the `request system snapshot` command, you cannot return to the previous version of the software because the running and backup copies of the software are identical.

Chapter 8

Reinstall JUNOS Software

If the JUNOS software becomes damaged, you must reinstall it. (See Table 22.)

Table 22: Checklist for Reinstalling JUNOS Software

JUNOS Software Reinstallation Tasks	Command or Action
Before You Reinstall JUNOS Software on page 75	
1. Log the Software Version Information on page 76	show version save <i>filename</i>
2. Log the Hardware Version Information on page 77	show chassis hardware save <i>filename</i>
3. Log the Chassis Environment Information on page 78	show chassis environment save <i>filename</i>
4. Log the System Boot-Message Information on page 78	show system boot-messages save <i>filename</i>
5. Log the Active Configuration on page 80	show configuration save <i>filename</i>
6. Log the Interfaces on the Router on page 80	show interface terse save <i>filename</i>
7. Log the BGP, IS-IS, and OSPF Adjacency Information on page 81	show bgp summary save <i>filename</i> show isis adjacency brief save <i>filename</i> show ospf neighbor brief save <i>filename</i>
8. Log the System Storage Information on page 82	show system storage save <i>filename</i>
9. Back Up the Currently Running and Active File System on page 82	request system snapshot
10. Have the Boot Floppy or PCMCIA Card Ready on page 83	http://www.juniper.net/support
Reinstall the JUNOS Software on page 83	Insert the floppy and reboot the system.
Reconfigure the JUNOS Software on page 83	
1. Configure Names and Addresses on page 84	Log in as root. Start the CLI. Enter configuration mode: configure set system host-name <i>host-name</i> set system domain-name <i>domain-name</i> set interfaces fxp0 unit 0 family inet address <i>address/prefix-length</i> set system backup-router <i>address</i> set system name-server <i>address</i>
2. Set the Root Password on page 85	set system root-authentication plain-text-password set system root-authentication encrypted-password password set system root-authentication ssh-rsa key commit exit
3. Check Network Connectivity on page 85	ping <i>address</i>

JUNOS Software Reinstallation Tasks	Command or Action
4. Copy Backup Configurations to the Router on page 86	file copy var/tmp configure [edit] load merge /config/ <i>filename</i> or load replace /config/ <i>filename</i> [edit] commit
After You Reinstall JUNOS Software on page 87	
1. Compare Information Logged Before and After the Reinstall on page 87	show version save <i>filename</i> show chassis hardware save <i>filename</i> show chassis environment save <i>filename</i> show system boot-messages save <i>filename</i> show configuration save <i>filename</i> show interfaces terse save <i>filename</i> show bgp summary show isis adjacency brief show ospf neighbor brief save <i>filename</i> show system storage save <i>filename</i>
2. Back Up the New Software on page 87	request system snapshot

Before You Reinstall JUNOS Software

Purpose Before you reinstall the JUNOS software, it is important to log information about the existing system so that after the reinstall you can verify that all software components are installed and working as expected. Also, while logging information, you might find an existing problem that you did not know about and might have thought was caused by the reinstall.

Steps To Take To log important information about your system, follow these steps:

1. Log the Software Version Information on page 76
2. Log the Hardware Version Information on page 77
3. Log the Chassis Environment Information on page 78
4. Log the System Boot-Message Information on page 78
5. Log the Active Configuration on page 80
6. Log the Interfaces on the Router on page 80
7. Log the BGP, IS-IS, and OSPF Adjacency Information on page 81
8. Log the System Storage Information on page 82
9. Back Up the Currently Running and Active File System on page 82
10. Have the Boot Floppy or PCMCIA Card Ready on page 83

In all of the logging steps, you can use your terminal program to save the output from the commands, or use the **save** command to redirect the output to an external file.

To save the output to a file on another machine, use the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> command | save filename
```

By default, the file is placed in your home directory on the router. To redirect the output to a file on another machine, change the filename to include the path to that machine and file. For information about how you can specify the filename, see the *JUNOS System Basics and Services Command Reference*.

The following example stores the output of the **show version** command in a file:

```
user@host> show version | save filename
Wrote 1143 lines of output to 'filename'
```

Step 1: Log the Software Version Information

Action To log the JUNOS software version information, use the following JUNOS CLI operational mode command:

```
user@host> show version | save filename
```

Sample Output user@host> show version | save test
Wrote 39 lines of output to 'test'

```
user@host> show version
Hostname: my-router.net
Model: m10
JUNOS Base OS boot [5.0R5]
JUNOS Base OS Software Suite [5.0R5]
JUNOS Kernel Software Suite [5.0R5]
JUNOS Routing Software Suite [5.0R5]
JUNOS Packet Forwarding Engine Support [5.0R5]
JUNOS Crypto Software Suite [5.0R5]
JUNOS Online Documentation [5.0R5]
KERNEL 5.0R5 #0 built by builder on 2002-03-02 05:10:28 UTC
MGD release 5.0R5 built by builder on 2002-03-02 04:45:32 UTC
CLI release 5.0R5 built by builder on 2002-03-02 04:44:22 UTC
CHASSISD release 5.0R5 built by builder on 2002-03-02 04:43:37 UTC
DCD release 5.0R5 built by builder on 2002-03-02 04:42:47 UTC
RPD release 5.0R5 built by builder on 2002-03-02 04:46:17 UTC
SNMPD release 5.0R5 built by builder on 2002-03-02 04:52:26 UTC
MIB2D release 5.0R5 built by builder on 2002-03-02 04:45:37 UTC
APSD release 5.0R5 built by builder on 2002-03-02 04:43:31 UTC
VRRPD release 5.0R5 built by builder on 2002-03-02 04:52:34 UTC
ALARMD release 5.0R5 built by builder on 2002-03-02 04:43:24 UTC
PFED release 5.0R5 built by builder on 2002-03-02 04:46:06 UTC
CRAFTD release 5.0R5 built by builder on 2002-03-02 04:44:30 UTC
SAMPLED release 5.0R5 built by builder on 2002-03-02 04:52:20 UTC
ILMID release 5.0R5 built by builder on 2002-03-02 04:45:21 UTC
BPRELAYD release 5.0R5 built by builder on 2002-03-02 04:42:41 UTC
RMOPD release 5.0R5 built by builder on 2002-03-02 04:46:11 UTC
jkernel-dd release 5.0R5 built by builder on 2002-03-02 04:41:07 UTC
jroute-dd release 5.0R5 built by builder on 2002-03-02 04:41:21 UTC
jdocs-dd release 5.0R5 built by builder on 2002-03-02 04:39:11 UTC
```

What It Means The sample output shows the hostname, router model, and the different JUNOS software packages, processes, and documents.

Step 2: Log the Hardware Version Information

Purpose You should log hardware version information in the rare event that a router cannot successfully reboot and you cannot obtain the Routing Engine serial number. The Routing Engine serial number is necessary for Juniper Networks Technical Assistance Center (JTAC) to issue a return to manufacturing authorization (RMA). Without the Routing Engine serial number, an onsite technician must be dispatched to issue the RMA.

Action To log the router chassis hardware version information, use the following JUNOS CLI operational mode command:

```
user@host> show chassis hardware | save filename
```

Sample Output The output for the M-series routers varies depending on the chassis components of each router. All routers have a chassis, midplanes or backplanes, power supplies, and Flexible PIC Concentrators (FPCs). Refer to the hardware guides for information about the different chassis components.

```
user@host> show chassis hardware | save test
Wrote 43 lines of output to 'test'
```

```
user@host> show chassis hardware
```

Item	Version	Part number	Serial number	Description
Chassis			101	M160
Midplane	REV 02	710-001245	S/N AB4107	
FPM CMB	REV 01	710-001642	S/N AA2911	
FPM Display	REV 01	710-001647	S/N AA2999	
CIP	REV 02	710-001593	S/N AA9563	
PEM 0	Rev 01	740-001243	S/N KJ35769	DC
PEM 1	Rev 01	740-001243	S/N KJ35765	DC
PCG 0	REV 01	710-001568	S/N AA9794	
PCG 1	REV 01	710-001568	S/N AA9804	
Host 1			da000004f8d57001	teknor
MCS 1	REV 03	710-001226	S/N AA9777	
SFM 0 SPP	REV 04	710-001228	S/N AA2975	
SFM 0 SPR	REV 02	710-001224	S/N AA9838	Internet Processor I
SFM 1 SPP	REV 04	710-001228	S/N AA2860	
SFM 1 SPR	REV 01	710-001224	S/N AB0139	Internet Processor I
FPC 0	REV 03	710-001255	S/N AA9806	FPC Type 1
CPU	REV 02	710-001217	S/N AA9590	
PIC 1	REV 05	750-000616	S/N AA1527	1x OC-12 ATM, MM
PIC 2	REV 05	750-000616	S/N AA1535	1x OC-12 ATM, MM
PIC 3	REV 01	750-000616	S/N AA1519	1x OC-12 ATM, MM
FPC 1	REV 02	710-001611	S/N AA9523	FPC Type 2
CPU	REV 02	710-001217	S/N AA9571	
PIC 0	REV 03	750-001900	S/N AA9626	1x STM-16 SDH, SMIR
PIC 1	REV 01	710-002381	S/N AD3633	2x G/E, 1000 BASE-SX
FPC 2				FPC Type OC192
CPU	REV 03	710-001217	S/N AB3329	
PIC 0	REV 01			1x OC-192 SM SR-2

What It Means The sample output shows the hardware inventory for an M160 router with a chassis serial number of 101. For each component, the output shows the version number, part number, serial number, and description.

Step 3: Log the Chassis Environment Information

Action To log the router chassis environment information, use the following JUNOS CLI operational mode command:

```
user@host> show chassis environment | save filename
```

Sample Output The following example shows output from the show chassis environment command for an M5 router:

```
user@m5-host> show chassis environment | save test
Wrote 14 lines of output to 'test'

user@m5-host> show chassis environment
Class Item                Status    Measurement
-----
Power Power Supply A      OK
       Power Supply B      OK
Temp  FPC Slot 0             OK        32 degrees C / 89 degrees F
       FEB                 OK        31 degrees C / 87 degrees F
       PS Intake           OK        26 degrees C / 78 degrees F
       PS Exhaust          OK        31 degrees C / 87 degrees F
Fans  Left Fan 1           OK        Spinning at normal speed
       Left Fan 2           OK        Spinning at normal speed
       Left Fan 3           OK        Spinning at normal speed
       Left Fan 4           OK        Spinning at normal speed
```

What It Means The sample output shows the environmental information about the router chassis, including the temperature and information about the fans, power supplies, and Routing Engine.

Step 4: Log the System Boot-Message Information

Action To log the system boot-message information, use the following JUNOS CLI operational mode command:

```
user@host> show system boot-messages | save filename
```

Sample Output

```
user@host> show system boot-messages | save test
Wrote 80 lines of output to 'test'

user@host> show system boot-messages
Copyright (c) 1992-1998 FreeBSD Inc.
Copyright (c) 1996-2000 Juniper Networks, Inc.
All rights reserved.
Copyright (c) 1982, 1986, 1989, 1991, 1993
    The Regents of the University of California. All rights reserved.

JUNOS 4.1-20000216-Zf8469 #0: 2000-02-16 12:57:28 UTC
    tlim@single.juniper.net:/p/build/20000216-0905/4.1/release_kernel/sys/compile/GENERIC
CPU: Pentium Pro (332.55-MHz 686-class CPU)
    Origin = "GenuineIntel" Id = 0x66a Stepping=10
    Features=0x183f9ff<FPU,VME,DE,PSE,TSC,MSR,PAE,MCE,CX8,SEP,MTRR,PGE,MCA,CMOV,<b16>,<b17>,MMX,<b24>>
Teknor CPU Card Recognized
```

```

real memory = 805306368 (786432K bytes)
avail memory = 786280448 (767852K bytes)
Probing for devices on PCI bus 0:
chip0 <generic PCI bridge (vendor=8086 device=7192 subclass=0)> rev 3 class 60000 on pci0:0:0
chip1 <Intel 82371AB PCI-ISA bridge> rev 1 class 60100 on pci0:7:0
chip2 <Intel 82371AB IDE interface> rev 1 class 10180 on pci0:7:1
chip3 <Intel 82371AB USB interface> rev 1 class c0300 int d irq 11 on pci0:7:2
smb0 <Intel 82371AB SMB controller> rev 1 class 68000 on pci0:7:3
pcic0 <TI PCI-1131 PCI-CardBus Bridge> rev 1 class 60700 int a irq 15 on pci0:13:0
TI1131 PCI Config Reg: [pci only][FUNC0 pci int]
pcic1 <TI PCI-1131 PCI-CardBus Bridge> rev 1 class 60700 int b irq 12 on pci0:13:1
TI1131 PCI Config Reg: [pci only][FUNC1 pci int]
fxp0 <Intel EtherExpress Pro 10/100B Ethernet> rev 8 class 20000 int a irq 12 on pci0:16:0
chip4 <generic PCI bridge (vendor=1011 device=0022 subclass=4)> rev 4 class 60400 on pci0:17:0
fxp1 <Intel EtherExpress Pro 10/100B Ethernet> rev 8 class 20000 int a irq 10 on pci0:19:0
Probing for devices on PCI bus 1:mcs0 <Miscellaneous Control Subsystem> rev 12 class ff0000 int a irq
12 on pci1:13:0
fxp2 <Intel EtherExpress Pro 10/100B Ethernet> rev 8 class 20000 int a irq 10 on pci1:14:0
Probing for devices on the ISA bus:
sc0 at 0x60-0x6f irq 1 on motherboard
sc0: EGA color <16 virtual consoles, flags=0x0>
ed0 not found at 0x300
ed1 not found at 0x280
ed2 not found at 0x340
psm0 not found at 0x60
sio0 at 0x3f8-0x3ff irq 4 flags 0x20010 on isa
sio0: type 16550A, console
sio1 at 0x3e8-0x3ef irq 5 flags 0x20000 on isa
sio1: type 16550A
sio2 at 0x2f8-0x2ff irq 3 flags 0x20000 on isa
sio2: type 16550A
pcic0 at 0x3e0-0x3e1 on isa
PC-Card ctlr(0) TI PCI-1131 [CardBus bridge mode] (5 mem & 2 I/O windows)
pcic0: slot 0 controller I/O address 0x3e0
npx0 flags 0x1 on motherboard
npx0: INT 16 interface
fdc0: direction bit not set
fdc0: cmd 3 failed at out byte 1 of 3
fdc0 not found at 0x3f0
wdc0 at 0x1f0-0x1f7 irq 14 on isa
wdc0: unit 0 (wd0): <SunDisk SDCFB-80>, single-sector-i/o
wd0: 76MB (156672 sectors), 612 cyls, 8 heads, 32 S/T, 512 B/S
wdc0: unit 1 (wd1): <IBM-DCXA-210000>
wd1: 8063MB (16514064 sectors), 16383 cyls, 16 heads, 63 S/T, 512 B/S
wdc1 not found at 0x170
wdc2 not found at 0x180
ep0 not found at 0x300
fxp0: Ethernet address 00:a0:a5:12:05:5a
fxp1: Ethernet address 00:a0:a5:12:05:59
fxp2: Ethernet address 02:00:00:00:00:01
swapon: adding /dev/wd1s1b as swap device
Automatic reboot in progress...
/dev/rwd0s1a: clean, 16599 free (95 frags, 2063 blocks, 0.1% fragmentation)
/dev/rwd0s1e: clean, 9233 free (9 frags, 1153 blocks, 0.1% fragmentation)
/dev/rwd0s1a: clean, 16599 free (95 frags, 2063 blocks, 0.1% fragmentation)
/dev/rwd1s1f: clean, 4301055 free (335 frags, 537590 blocks, 0.0% fragmentation)

```

What It Means The sample output shows the initial messages generated by the system kernel upon boot. This is the content of the `/var/run/dmesg.boot` file.

Step 5: Log the Active Configuration

Action To log the active configuration on the router, use the following JUNOS CLI operational mode command:

```
user@host> show configuration | save filename
```

Sample Output user@host> show configuration | save test
Wrote 4076 lines of output to 'test'

```
user@host> show configuration
system {
  host-name lab8;
  domain-name juniper.net;
  backup-router 10.1.1.254;
  time-zone America/Los_Angeles;
  default-address-selection;
  dump-on-panic;
  name-server {
[...Output truncated...]
```

What It Means The sample output shows the configuration currently running on the router, which is the last committed configuration.

Step 6: Log the Interfaces on the Router

Action To log the interfaces on the router, use the following JUNOS CLI operational mode command:

```
user@host> show interface terse | save filename
```

Sample Output user@host> show interfaces terse | save test
Wrote 81 lines of output to 'test'

```
user@host> show interfaces terse
Interface      Admin Link Proto Local Remote
at-1/3/0       up    up
at-1/3/0.0     up    up    inet  1.0.0.1    --> 1.0.0.2
               iso
fxp0           up    up
fxp0.0         up    up    inet  10.168.5.59/24
gre            down  up
ipip            down  up
lo0            up    up
lo0.0          up    up    inet  127.0.0.1    --> 0/0
               iso
47.0005.80ff.f800.0000.0108.0001.1921.6800.5059.00
so-1/2/0       up    down
so-1/2/1       down  down
so-1/2/2       down  down
so-1/2/3       down  down
so-2/0/0       up    up
so-2/0/0.0     up    up    inet  1.2.3.4      --> 1.2.3.5
               iso
[...Output truncated...]
```

What It Means The sample output displays summary information about the physical and logical interfaces on the router.

Step 7: Log the BGP, IS-IS, and OSPF Adjacency Information

Purpose The following commands log useful information about Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (IS-IS), and Open Shortest Path First (OSPF) protocols. If you have other protocols installed, such as Multiprotocol Label Switching (MPLS), Resource Reservation Protocol (RSVP), or Protocol Independent Multicast (PIM), you also might log summary information for them.

Action To log the protocol peer information, use the following JUNOS CLI operational mode commands:

```
user@host> show bgp summary | save filename
user@host> show isis adjacency brief | save filename
user@host> show ospf neighbor brief | save filename
```

Sample Output 1

```
user@host> show bgp summary | save test
Wrote 45 lines of output to 'test'
```

```
user@host> show bgp summary
Groups: 1 Peers: 1 Down peers: 0
Table Tot Paths Act Paths Suppressed History Damp State Pending
inet.0 4 4 0 0 0 0 0
Peer AS InPkt OutPkt OutQ Flaps Last Up/Dwn State#Active/Received/Damped..
9.9.3.1 2 2627 2628 0 0 21:50:12 4/4/0 0/0/0
```

Sample Output 2 user@host> show isis adjacency brief | save test
Wrote 7 lines of output to 'test'

```
user@host> show isis adjacency brief
IS-IS adjacency database:
Interface System L State Hold (secs) SNPA
so-1/0/0.0 1921.6800.5067 2 Up 13
so-1/1/0.0 1921.6800.5067 2 Up 25
so-1/2/0.0 1921.6800.5067 2 Up 20
so-1/3/0.0 1921.6800.5067 2 Up 19
so-2/0/0.0 1921.6800.5066 2 Up 19
so-2/1/0.0 1921.6800.5066 2 Up 17
so-2/2/0.0 1921.6800.5066 2 Up 20
so-2/3/0.0 1921.6800.5066 2 Up 20
so-5/0/0.0 ranier 2 Up 17
```

Sample Output 3 user@host> show ospf neighbor brief | save test
Wrote 10 lines of output to 'test'

```
user@host> show ospf neighbor brief
Address Intf State ID Pri Dead
10.168.254.225 fxp3.0 2Way 10.250.240.32 128 36
10.168.254.230 fxp3.0 Full 10.250.240.8 128 38
10.168.254.229 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
```

What It Means Sample output 1 displays summary information about BGP and its neighbors. Sample output 2 displays information about IS-IS neighbors. Sample output 3 displays information about all OSPF neighbors.

Step 8: Log the System Storage Information

Action To log the system storage statistics for the amount of free disk space in the router's file system, use the following JUNOS CLI operational mode command:

```
user@host> show system storage | save filename
```

Sample Output user@host> show system storage | save test
Wrote 14 lines of output to 'test'

```
user@host> show system storage
Filesystem 1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a    65687    26700    33733    44%      /
devfs          16         16         0    100%    /dev/
/dev/vn1       9310     9310         0    100%    /packages/mnt/jbase
/dev/vn2       8442     8442         0    100%    /packages/mnt/jkernel-5.0R5.1
/dev/vn3      11486    11486         0    100%    /packages/mnt/jpfe-5.0R5.1
/dev/vn4       5742     5742         0    100%    /packages/mnt/jroute-5.0R5.1
/dev/vn5       1488     1488         0    100%    /packages/mnt/jcrypto-5.0R5.1
/dev/vn6        792         792         0    100%    /packages/mnt/jdocs-5.0R5.1
mfs:2373    1015815         3    934547     0%      /tmp
/dev/ad0s1e    25263         11    23231     0%      /config
procfs         4          4         0    100%    /proc
/dev/ad1s1f   9825963   1811085   7228801    20%     /var
```

What It Means The sample output displays statistics about the amount of free disk space in the router's file system. Values are displayed in 1024-byte (1-KB) blocks.

Step 9: Back Up the Currently Running and Active File System

Action To back up the currently running and active file system so that you can recover to a known, stable environment in case there is a problem during the reinstall, use the following JUNOS CLI operational mode command:

```
user@host> request system snapshot
```

Sample Output user@host> request system snapshot
umount: /altroot: not currently mounted
Copying / to /altroot.. (this may take a few minutes)
umount: /altconfig: not currently mounted
Copying /config to /altconfig.. (this may take a few minutes)

The following filesystems were archived: / /config

What It Means The root file system is backed up to /altroot, and /config is backed up to /altconfig. The root and /config file systems are on the router's internal flash drive, and the /altroot and /altconfig file systems are on the router's hard drive.



NOTE: After you issue the `request system snapshot` command, you cannot return to the previous version of the software because the running and backup copies of the software are identical.

Step 10: Have the Boot Floppy or PCMCIA Card Ready

Action Have available the removable medium that shipped with the router (also called a boot floppy) or the Personal Computer Memory Card International Association (PCMCIA) card. If you do not have a boot floppy, contact customer support at <http://www.juniper.net/support>.

Reinstall the JUNOS Software

Action To reinstall the JUNOS software, follow these steps:

1. Insert the removable medium (boot floppy) into the router.
2. Reboot the router, either by power-cycling it or by issuing the `request system reboot` command from the CLI.
3. At the following prompt, type **y**:

WARNING: The installation will erase the contents of your disk. Do you wish to continue (y/n)?

The router copies the software from the removable medium onto your system, occasionally displaying status messages. This can take up to 10 minutes.

4. Remove the removable medium when prompted.

The router reboots from the primary boot device on which the software is installed. When the reboot is complete, the router displays the login prompt.

Reconfigure the JUNOS Software

Purpose After you have reinstalled the software, you must copy the router's configuration files back to the router. (You also can configure the router from scratch, as described in *JUNOS Internet Software System Basics Configuration Guide*.) However, before you can copy the configuration files, you must establish network connectivity.

Steps To Take To reconfigure the software, follow these steps:

1. Configure Names and Addresses on page 84
2. Set the Root Password on page 85
3. Check Network Connectivity on page 85
4. Copy Backup Configurations to the Router on page 86

Step 1: Configure Names and Addresses

Action To configure the machine name, domain name, and various addresses, follow these steps:

1. Log in as **root**. There is no password.
2. Start the CLI:

```
root# cli
root@>
```

3. Enter configuration mode:

```
cli> configure
[edit]
root@#
```

4. Configure the name of the machine. If the name includes spaces, enclose the entire name in quotation marks (" "):

```
[edit]
root@# set system host-name host-name
```

5. Configure the machine's domain name:

```
[edit]
root@# set system domain-name domain-name
```

6. Configure the IP address and prefix length for the router's management Ethernet interface:

```
[edit]
root@# set interfaces fxp0 unit 0 family inet address address/prefix-length
```

7. Configure the IP address of a default router. This system is called the backup router because it is used only while the routing protocol process is not running.

```
[edit]
root@# set system backup-router address
```

8. Configure the IP address of a Domain Name Server (DNS) server:

```
[edit]
root@# set system name-server address
```

Step 2: Set the Root Password

Action To set the root password, follow these steps:

1. To set the root password, enter a clear-text password that the system will encrypt, a password that is already encrypted, or a secure shell (ssh) public key string.

- To enter a clear-text password, use the following command to set the root password:

```
[edit]
root@# set system root-authentication plain-text-password
New password: password
Retype new password: password
```

- To enter a password that is already encrypted, use the following command to set the root password:

```
[edit]
root@# set system root-authentication encrypted-password password
```

- To enter an ssh public string, use the following command to set the root password:

```
[edit]
root@# set system root-authentication ssh-rsa key
```

2. Commit the changes:

```
[edit]
root@# commit
```

3. Exit from configuration mode:

```
[edit]
root@# exit
root@>
```

Step 3: Check Network Connectivity

Action To check that the router has network connectivity, issue a **ping** command to a system on the network:

```
root@> ping address
```

If there is no response, verify that there is a route to the **address** using the **show route** command. If the address is outside your **fxp0** subnet, add a static route. Once the backup configuration is loaded and committed, the static route is no longer needed and should be deleted.

Step 4: Copy Backup Configurations to the Router

Action To copy backup configurations to the router, follow these steps:

1. To copy the existing configuration and any backup configurations back onto the router, use the **file copy** command. Place the files in the **/var/tmp** directory.

```
user@host> file copy var/tmp/filename
```

2. Load and activate the desired configuration:

```
root@> configure
[edit]
root@# load merge /config/filename or load replace /config/filename
[edit]
root@# commit
```

After You Reinstall JUNOS Software

Steps To Take To verify that the new version of the JUNOS software is running as expected after the reinstall, follow these steps:

1. Compare Information Logged Before and After the Reinstall on page 87
2. Back Up the New Software on page 87

Step 1: Compare Information Logged Before and After the Reinstall

Purpose Compare the operation of the system before and after the reinstall to ensure that everything is working as expected.

Action To obtain system information, use the following commands:

```
user@host> show version
user@host> show chassis hardware
user@host> show chassis environment
user@host> show system boot-messages
user@host> show configuration
user@host> show interface terse
user@host> show bgp summary
user@host> show isis adjacency brief
user@host> show ospf neighbor brief
user@host> show system storage
```

Compare the information from these commands with the information you obtained before the reinstall.

Step 2: Back Up the New Software

Purpose After a week or so, when you are satisfied that the new software is running successfully, we recommend that you back up the reinstalled software.

Action To back up the reinstalled software, use the following JUNOS CLI operational mode command:

```
user@host> request system snapshot
```

Sample Output The root file system is backed up to /altroot, and /config is backed up to /altconfig. The root and /config file systems are on the router's internal flash drive, and the /altroot and /altconfig file systems are on the router's hard drive.



NOTE: After you issue the `request system snapshot` command, you cannot return to the previous version of the software because the running and backup copies of the software are identical.

Part 3

Verify Your Network Topology

- Verify Juniper Networks Routers on page 91
- Verify Physical Interfaces on the Router on page 101
- Verify the IS-IS Protocol and Adjacencies on page 111
- Verify the OSPF Protocol and Neighbors on page 129
- Verify the BGP Protocol and Peers on page 157
- Verify the Routing Engine CPU Memory on page 179
- Verify Traffic and Packets through the Router on page 189
- Use the ping and traceroute Commands on page 199
- Use MIBs on page 205

Chapter 9

Verify Juniper Networks Routers

This chapter describes how to check the hardware components of Juniper Networks routers on your network. (See Table 23.)

Table 23: Checklist for Verifying Juniper Networks Routers

Verify Juniper Networks Routers Tasks	Command or Action
Check Router Components on page 92	
1. Check the Router Component Status on page 93	
a. Check the Router Craft Interface on page 93	<code>show chassis craft-interface</code>
b. Check the Component LEDs on page 94	<code>show chassis craft-interface</code>
c. Display Detailed Component Environmental Information on page 95	<code>show chassis environment <i>component-name</i></code>
d. Display Detailed Operational Information About Components on page 96	<code>show chassis <i>component-name</i></code>
2. Gather Component Alarm Information on page 97	
a. Display the Current Router Alarms on page 97	<code>show chassis alarms</code>
b. Display Error Messages in the Messages Log File on page 97	<code>show log messages</code>
c. Display Error Messages in the Chassis Process Log File on page 98	<code>show log chassisd</code>
3. Verify the Component Problem on page 98	Make sure that the component is well seated in its slot and connected to the router midplane. Perform a swap test on the component with a problem.
4. Fix the Problem on page 99	Take action and correct the problem. For example, replace a dirty air filter, clean a fiber-optic cable, connect the component securely to the midplane, or reset the component. Otherwise, escalate the alarm condition and contact JTAC. Do not straighten bent pins.
5. Contact JTAC on page 99	<code>request support information</code> <code>request support information save <i>filename</i></code>
6. Return the Failed Component on page 100	<code>show chassis hardware</code> Obtain a Return Material Authorization (RMA) number from JTAC. You can send e-mail to support@juniper.net or call 1-888-314-JTAC (within the United States) or 1-408-745-9500 (from outside the United States).

Check Router Components

Purpose When you monitor router components, you are making sure that there are no hardware problems with the router. In the event of a minor problem, you can try to fix it. For more difficult situations, you can call for assistance from the Juniper Networks Technical Assistance Center (JTAC).

Steps To Take To monitor M-series and T-series router components, follow these steps:

1. Check the Router Component Status on page 93
2. Gather Component Alarm Information on page 97
3. Verify the Component Problem on page 98
4. Fix the Problem on page 99
5. Contact JTAC on page 99
6. Return the Failed Component on page 100

Check the Router Component Status

Purpose When you check the router craft interface, the component LEDs, and the environmental and operational information, you are either physically inspecting the components or obtaining output about their status from commands you issue from the command-line interface (CLI).

Steps To Take To check the router component status, follow these steps:

1. Check the Router Craft Interface on page 93
2. Check the Component LEDs on page 94
3. Display Detailed Component Environmental Information on page 95
4. Display Detailed Operational Information About Components on page 96

Step 1: Check the Router Craft Interface

Action To check the craft interface information for router status, do one of the following:

- Use the following CLI command:

```
user@host> show chassis craft-interface
```

The command output displays the router alarm indicator status, the LCD display information (router name, router uptime, and status message that rotate at 2-second intervals), and the major component LED status.

- Physically look at the router craft interface. Table 24 shows the component characteristics of the craft interface for each M-series router and T-series platform.

Table 24: Craft Interface Components for the M-series Routers and T-series Platforms

Component	M5 and M10	M20	M40	M40e	M160	T320	T640
Alarm LEDs	X	X	X	X	X	X	X
Lamp test button	X					X	X
Alarm cutoff button		X		X	X	X	
Alarm relay contacts		X	X				
Link and activity status lights	X	X					
LCD display and navigation buttons			X	X	X	X	X
Routing Engine ports	X	X	X				
Routing Engine LEDs		X	X			X	
Host module LEDs				X			
Host subsystem LEDs						X	X
Physical Interface Card (PIC) online and offline buttons	X						

Component	M5 and M10	M20	M40	M40e	M160	T320	T640
Flexible PIC Concentrator (FPC) LEDs		X	X	X	X	X	X
FPC offline buttons		X		X		X	X
FPC online buttons						X	X
Switch Interface Board (SIB) LEDs						X	X

Step 2: Check the Component LEDs

Action To check the component LED status, do one of the following:

- Use the following CLI command:

```
user@host> show chassis craft-interface
```

The output shows the messages that are currently displayed on the craft interface (for routers that have a display on the craft interface).

For examples of sample output, see the *JUNOS System Basics and Services Command Reference*.

- Physically look at the craft interface. You see the following component LEDs: Routing Engine, FPCs, PICs, host module (for M40e and M160 routers), and host subsystem and SIB (for T-series platforms).
- Look at the LEDs on the component faceplate. Figure 25 describes where the LEDs are located on the router or platform.

Table 25: Component LED Location on the Router or Platform

Component	LED Location on the Router	Router or Platform
CB	On the Control Board (CB) faceplate.	T320 router and T640 routing node
FPC	On the FPC faceplate at the front of the router.	M20, M40, M40e, and M160 routers
Host module	On the craft interface. Remove the component cover.	M40e and M160 routers
MCS	On the Miscellaneous Control System (MCS) faceplate at the rear of the router. Remove the component cover.	M40e and M160 routers
PIC	On the craft interface. On the PIC faceplate at the front of the router.	M5 and M10 routers All other routers
PCG	On the PFE clock generator (PCG) faceplate at the rear of the router. Remove the component cover.	M40e and M160 routers
Power supply	On the power supply faceplate at the bottom rear of the router.	All routers
Routing Engine	On the rear Routing Engine panel. On the craft interface.	M20 router M20, M40, M40e, and M160 routers
SCB	On the System Control Board (SCB) faceplate at the front of the router, vertical in the middle of the FPC card cage.	M40 router
SCG	On the SONET Clock Generator (SCG) faceplate.	T320 router and T640 routing node

Component	LED Location on the Router	Router or Platform
SFM	On the Switching and Forwarding Module (SFM) faceplate at the rear of the router. Remove the component cover.	M40e and M160 routers
SIB	On the SIB faceplate.	T320 router and T640 routing node
SSB	On the System and Switch Board (SSB) faceplate at the top front of the router.	M20 router

Step 3: Display Detailed Component Environmental Information

Action To display detailed environmental status information about a component, use the following CLI command:

```
user@host> show chassis environment component-name
```

Sample Output The command output displays the temperature of the air passing by the component, in degrees Centigrade. It also displays the total percentage of CPU, interrupt, heap space, and buffer space being used by the component processor, including the total DRAM available to the component processor. The command output displays the time when the component started running and how long the component has been running. A short uptime can indicate a problem with the component.

For examples of sample output, see the *JUNOS System Basics and Services Command Reference*.

Figure 26 lists the operational mode CLI commands that display more detailed information for each router and platform component.

Table 26: CLI Commands for Detailed Component Environment Status

Component	Operational Mode CLI Command	Router or Platform
CB	show chassis environment cb	T320 and T640 platforms
Forwarding Engine Board (FEB)	show chassis feb	M5 and M10 routers
FPC	show chassis environment fpc	M40e and M160 routers, and T-series platforms
Front panel module (FPM) or craft interface	show chassis environment fpm	M40e and M160 routers, and T-series platforms
MCS	show chassis environment mcs	M40e and M160 routers
PCG	show chassis environment pcg	M40e and M160 routers
Power Entry Module (PEM) or power supply	show chassis environment pem	M40e and M160 routers, and T-series platforms
Routing Engine	show chassis environment routing-engine	M40e and M160 routers, and T-series platforms
SONET Clock Generator (SCG)	show chassis environment scg	T320 and T640 platforms
SFM	show chassis environment sfm	M40e and M160 routers
SIG	show chassis environment sib	T320 and T640 platforms

Step 4: Display Detailed Operational Information About Components

Action To display detailed operational information about a component, use the following CLI command:

```
user@host> show chassis component-name
```

The command output displays the temperature of the air passing by the component, in degrees Centigrade and Fahrenheit. It displays the total percentage of CPU, interrupt, heap space, and buffer space being used by the component processor, including the total DRAM available to the component processor. The command output displays the time when the component started running and how long the component has been running. A short uptime can indicate a problem with the component.

For examples of sample output, see the *JUNOS System Basics and Services Command Reference*.

Figure 27 lists the components for which you can display more detailed operational status information.

Table 27: CLI Commands for Detailed Operational Status of Components

Component	Operational Mode CLI Command	
FEB	show chassis feb	M5 and M10 routers
FPC	show chassis fpc	M40e and M160 routers, and T-series platforms
Routing Engine	show chassis routing-engine	M40e and M160 routers, and T-series platforms
SCB	show chassis scb	M40 routers
SFM	show chassis sfm	M40e and M160 routers
SSB	show chassis ssb	M20 routers
Switch Processor Mezzanine Board (SPMB)	show chassis spmb	T320 and T640 platforms

Gather Component Alarm Information

Purpose When you obtain information about component alarms and error messages, you determine when a problem with a component first appeared and the details of the situation.

Steps To Take To gather component alarm information, follow these steps:

1. Display the Current Router Alarms on page 97
2. Display Error Messages in the Messages Log File on page 97
3. Display Error Messages in the Chassis Process Log File on page 98

Step 1: Display the Current Router Alarms

Action To display the current router component alarms, use the following CLI command:

```
user@host> show chassis alarms
```

The command output displays the number of alarms currently active, the time when the alarm began, the severity level, and an alarm description. Note the date and time of an alarm so that you can correlate it with error messages in the `messages` system log file.

For examples of sample output, see the *JUNOS System Basics and Services Command Reference*.

Step 2: Display Error Messages in the Messages Log File

Action To display router component error messages in the `messages` system log file, use the following CLI command:

```
user@host> show log messages
```

The `messages` system log file records the time the failure or event occurred, the severity level, a code, and a message description. Error messages in the `messages` system log file are logged at least 5 minutes before and after the alarm event.

To search for specific information in the log file, use the `| match component-name` command; for example, use `show log messages | match fpc`. If there is a space in the component name, enclose the component name in quotation marks; for example, `| match "power supply"`.

To search for multiple items in the log file, use the `| match` command followed by the multiple items, separated by the `|` (pipe), and enclosed in quotation marks; for example, `| match "fpc | sfm | kernel | tnp"`.

To monitor the `messages` file in real time, use the `monitor start messages` CLI command. This command displays the new entries in the file until you stop monitoring by using the `monitor stop messages` CLI command.

For more information about system log messages, see the *JUNOS System Log Messages Reference*.

Step 3: Display Error Messages in the Chassis Process Log File

Action To display router component errors in the chassis process (chassisd) system log file, use the following CLI command:

```
user@host> show log chassisd
```

The chassis process (chassisd) log file tracks the state of each chassis component. For examples of sample output, see the *JUNOS System Basics and Services Command Reference*.

To search for specific information in the log file, use the `| match component-name` command; for example, `show log messages | match fpc`. If there is a space in the component name, enclose the component name in quotation marks; for example, `| match "power supply"`.

To search for multiple items in the log file, use the `| match` command followed by the multiple items, separated by the `|` (pipe), and enclosed in quotation marks; for example, `| match "fpc | sfm | kernel | tnp"`.

To monitor the chassisd file in real time, use the `monitor start chassisd` CLI command. This command displays the new entries in the file until you stop monitoring by using the `monitor stop chassisd` CLI command.

Verify the Component Problem

Purpose Test a component only if it is not associated with a previously reported router component failure case and if testing will not compromise the integrity of the router and other components.

Action To verify component failure, follow these steps:

1. Make sure that the component is well seated in its slot and connected to the router midplane.
2. Perform a swap test on the component that has failed or has a problem. Take the component offline if necessary, remove it, and replace it with one that you know works. If the replaced component works, it confirms that there was a problem with the component you removed.



NOTE: Before performing a swap test, always check for bent pins in the midplane and check the component for stuck pins in the connector. Pins stuck in the component connector can damage other good slots during a swap test.

Fix the Problem

Action If the router alarm condition is your responsibility, take action and correct it. For example, replace a dirty air filter, clean a fiber-optic cable, connect the component securely to the midplane, or reset the component. Otherwise, escalate the alarm condition and contact JTAC.



NOTE: Do not straighten component pins. If the pins on a component are bent, return the component with a Return Material Authorization (RMA). Straightening the pins may cause intermittent problems in the future.

Contact JTAC

Action If you cannot determine the cause of a problem or need additional assistance, contact JTAC at support@juniper.net, or at 1-888-314-JTAC (within the United States) or 1-408-745-9500 (from outside the United States).

To provide JTAC with information about the system, use the following CLI command:

```
user@host> request support information
```

Include the command output in your support request.

Because the output of this command is generally quite long, you can redirect the output to a file by using the following CLI command:

```
user@host> request support information | save filename
```

The **request support information** command is a combination of the following CLI operational mode commands:

- **show version**—Displays version information for the JUNOS software packages and the software for each software process.
- **show chassis firmware**—Displays the version levels of the firmware running on the SCB, SFM, SSB, FEB, and FPCs.
- **show chassis hardware**—Displays a list of all components installed in the router chassis. The output includes the component name, version, part number, serial number, and a brief description.
- **show chassis environment**—Displays environmental information about the router chassis, including the temperature and information about the fans, power supplies, and Routing Engine.
- **show interfaces extensive**—Displays static status information about router interfaces.

- **show configuration** (excluding any **SECRET-DATA**)—Displays the configuration that currently is running on the router, which is the last committed configuration. If you have modified the configuration since you last committed it, the configuration information displayed by the **show configuration** command will be different from that displayed with the **show** command from the **[edit]** hierarchy level in JUNOS software CLI configuration mode.
- **show system virtual-memory**—Displays the usage of JUNOS kernel memory, listed first by size of allocation and then by type of usage.

Return the Failed Component

Action To return a failed component, follow these steps:

1. Determine the part number and serial number of the component. To list the numbers for all components installed in the chassis, use the following CLI command:

```
user@host> show chassis hardware
```

If the component does not appear in the hardware inventory listing, check the failed component for the attached serial number ID label.



NOTE: The cooling system components (fans and impellers) do not have serial numbers. Therefore, you will not see a serial number listed in the hardware inventory or a serial number ID label on the component.

2. Obtain a Return Material Authorization (RMA) number from JTAC. You can send e-mail to support@juniper.net, or call 1-888-314-JTAC (within the United States) or 1-408-745-9500 (from outside the United States).

Provide the following information in your e-mail message or during the telephone call:

- Part number, description, and serial number of the component
- Your name, organization name, telephone number, fax number, and e-mail address
- Shipping address for the replacement component, including a contact name, phone number, and e-mail address
- Description of the failure, including log messages

The support representative will validate your request and issue an RMA number for the return of the component.

3. Pack the router or component for shipment, as described in the appropriate router hardware guide. Label the package with the corresponding RMA number.

Chapter 10

Verify Physical Interfaces on the Router

This chapter describes how to check the physical interfaces on a Juniper Networks router. (See Table 28.)

Table 28: Checklist for Verifying Physical Interfaces on a Router

Verify Physical Interfaces Tasks	Command or Action
Check Physical Interfaces on a Router on page 102	
1. Display Summary Interface Information on page 102	show interfaces terse show interfaces terse <i>interface-name</i>
2. Display Detailed Interface Information on page 103	show interfaces <i>interface-name</i> extensive
Display Real-Time Statistics about a Physical Interface on page 107	monitor interface <i>interface-name</i>
Check System Logging on page 109	show log messages match <i>interface-name</i>

Check Physical Interfaces on a Router

Purpose When you check the physical interfaces on a router, you gather information to quickly diagnose problems.

Steps To Take To check the physical interfaces on a router, follow these steps:

1. Display Summary Interface Information on page 102
2. Display Detailed Interface Information on page 103

Step 1: Display Summary Interface Information

Purpose By displaying a summary of the interfaces on a router, you begin the process of isolating problems when they occur.

Action To display a summary of all interfaces on a router or a specific group of interfaces, use one of the following JUNOS command-line interface (CLI) operational mode commands:

```
user@host> show interfaces terse
user@host> show interfaces terse interface-name
```

Sample Output The following sample output shows all interfaces on a router:

```
user@host> show interfaces terse
Interface           Admin Link Proto Local              Remote
so-5/0/0            up    down
t3-6/0/0            up    down
t3-6/0/1            up    down
t3-6/0/2            up    down
t3-6/0/3            up    down
at-6/1/0            up    down
fe-7/0/0            up    up
fe-7/0/0.0          up    up    vpls
fe-7/0/1            up    up
fe-7/0/2            up    up
fe-7/0/3            up    up
t3-7/1/0            up    down
t3-7/1/1            up    down
t3-7/1/2            up    down
t3-7/1/3            up    down
dsc                 up    up
fxp0                up    up
fxp0.0              up    up    inet  10.168.4.32/24
fxp1                up    up
fxp1.0              up    up    tnp   4
gre                 up    up
ipip                up    up
lo0                 up    up
lo0.0               up    up    inet  127.0.0.1      --> 0/0
lsi                 up    up
mtun                up    up
pimd                up    up
pime                up    up
tap                 up    up
```

The following sample output is for a specific group of SONET interfaces on a router:

```
user@host> show interfaces terse so*
so-0/0/0      up    up
so-0/0/0.1    up    down inet 10.1.13.2/30
              iso
so-0/0/0.2    up    down inet 10.1.23.2/30
              iso
so-0/0/0.4    up    down inet 10.1.34.1/30
              iso
so-0/0/0.5    up    up    inet 10.1.35.1/30
              iso
so-0/0/1      up    up
so-0/0/2      up    up
so-0/0/3      up    up
              iso
              iso
              iso
              iso
              iso
              iso
47.0005.80ff.f800.0000.0108.0001.0102.5524.5219.00
```

What It Means The sample output shows summary information about the interfaces on the router listed in order of type of interface. The information includes the name of the interface, whether it is turned on (up) or off (down), whether the state of the link is up or down, the protocol configured on the interface, the local address of the interface, and the address of the remote side of the connection if the interface is a point-to-point interface.

Step 2: Display Detailed Interface Information

Purpose Detailed interface information is useful when you need to further investigate the status of an interface after you have determined that there might be a problem.

Action To display detailed information about the status of an interface, use the following JUNOS CLI operational mode command:

```
user@host> show interfaces interface-name extensive
```

Sample Output The sample output is for an ATM interface. The fields vary depending on the type of interface.

```
user@host> show interfaces at-7/0/0 extensive
Physical interface: at-7/0/0, Enabled, Physical link is Up
  Interface index: 101, SNMP ifIndex: 106, Generation: 100
  Description: bangkok51 at-1/1/0
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SONET mode,
  Speed: OC12, Loopback: None,
  Payload scrambler: Enabled
  Device flags   : Present Running
  Link flags     : None
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:90:69:10:c7:72
  Last flapped   : 2002-05-23 12:12:18 PDT (1d 03:20 ago)
  Statistics last cleared: Never
  Traffic statistics:
    Input bytes   :          9526697          744 bps
    Output bytes  :         10458384          496 bps
```

```

Input packets:          129969          0 pps
Output packets:         126940          0 pps
Input errors:
  Errors: 0, Drops: 0, Invalid VCs: 0, Framing errors: 0, Policed
discards: 0, L3 incompletes: 0,
  L2 channel errors: 0, L2 mismatch timeouts: 0
Output errors:
  Carrier transitions: 0, Errors: 0, Drops: 0, Aged packets: 0
SONET alarms   : None
SONET defects  : None
SONET PHY:
  Seconds      Count  State
  PLL Lock     0       0 OK
  PHY Light    0       0 OK
SONET section:
  BIP-B1       1       9
  SEF          0       0 OK
  LOS          0       0 OK
  LOF          0       0 OK
  ES-S         1
  SES-S        0
  SEFS-S       0
SONET line:
  BIP-B2       1      183
  REI-L        1      323
  RDI-L        0       0 OK
  AIS-L        0       0 OK
  BERR-SF      0       0 OK
  BERR-SD      0       0 OK
  ES-L         1
  SES-L        0
  UAS-L        0
  ES-LFE       1
  SES-LFE      0
  UAS-LFE      0
SONET path:
  BIP-B3       1      31
  REI-P        1     216
  LOP-P        0       0 OK
  AIS-P        0       0 OK
  RDI-P        0       0 OK
  UNEQ-P       0       0 OK
  PLM-P        0       0 OK
  ES-P         1
  SES-P        0
  UAS-P        0
  ES-PFE       1
  SES-PFE      0
  UAS-PFE      0
Received SONET overhead:
  F1      : 0x00, J0      : 0x00, K1      : 0x00, K2      : 0x00
  S1      : 0x00, C2      : 0x13, C2(cmp) : 0x13, F2      : 0x00
  Z3      : 0x00, Z4      : 0x00, S1(cmp) : 0x00, V5      : 0x00
  V5(cmp) : 0x00
Transmitted SONET overhead:
  F1      : 0x00, J0      : 0x01, K1      : 0x00, K2      : 0x00
  S1      : 0x00, C2      : 0x13, F2      : 0x00, Z3      : 0x00
  Z4      : 0x00, V5      : 0x00
ATM status:
  HCS state: Sync
  LOC      : OK
ATM Statistics:
  Uncorrectable HCS errors: 77, Correctable HCS errors: 5, Tx cell
FIFO overruns: 0,

```



```

Rx cell FIFO overruns: 1, Rx cell FIFO underruns: 0, Input cell
count: 421980,
Output cell count: 139110927341, Output idle cell count: 1671702365,
Output VC queue drops: 0,
Input no buffers: 0, Input length errors: 0, Input timeouts: 0,
Input invalid VCs: 143301,
Input bad CRCs: 0, Input OAM cell no buffers: 0
Packet Forwarding Engine configuration:
Destination slot: 7
CoS transmit queue      Bandwidth      Buffer      Priority
Limit
                                %      bps      %      bytes
0 best-effort            0          0      0          0
low  none
1 expedited-forwarding  0          0      0          0
low  none
2 assured-forwarding    0          0      0          0
low  none
3 network-control       0          0      0          0
low  none
Logical interface at-7/0/0.100 (Index 49) (SNMP ifIndex 143) (Generation 76)
Flags: Point-To-Point Inverse-ARP SNMP-Traps Encapsulation: ATM-SNAP
Traffic statistics:
Input bytes :          9993
Output bytes :         16246
Input packets:         151
Output packets:        136
Local statistics:
Input bytes :          9993
Output bytes :         16246
Input packets:         151
Output packets:        136
Transit statistics:
Input bytes :           0          0 bps
Output bytes :           0          0 bps
Input packets:           0          0 pps
Output packets:          0          0 pps
Protocol inet, MTU: 4470, Flags: None, Generation: 200 Route table: 0
Addresses, Flags: Is-Preferred Is-Primary
Destination: 10.9.140.1, Local: 10.9.140.2, Broadcast:
Unspecified, Generation: 106
Protocol iso, MTU: 4470, Flags: None, Generation: 201 Route table: 0
Protocol mpls, MTU: 4458, Flags: None, Generation: 202 Route table: 0
VCI 0.200
Flags: Active, Inverse-ARP, OAM, Shaping
VBR, Peak: 12mbps, Sustained: 10mbps, Burst size: 24, Queue length: 0
OAM, Period 10 sec, Up count: 5, Down count: 4
Total down time: 0 sec, Last down: Never
ATM per-VC transmit statistics:
Tail queue packet drops: 0
OAM F5 cell statistics:
Total received: 49, Total sent: 49
Loopback received: 49, Loopback sent: 49
Last received: 00:00:08, Last sent: 00:00:08
RDI received: 0, RDI sent: 0
AIS received: 0
Traffic statistics:
Input bytes :          9993
Output bytes :         16246
Input packets:         151
Output packets:        136

```

What It Means The sample output shows static status information about this particular ATM interface. For examples of sample output for supported interfaces, see the *JUNOS Network Interfaces Configuration Guide*.

Table 29 lists the interface types supported by the JUNOS software and shows the interface name as it appears in the output.

Table 29: Interface Types Supported by the JUNOS Software

Interface Group	Interface Type	Format of <i>interface-name</i>
ATM	ATM	<i>at-fpc/pic/port</i>
Channelized	Channelized DS-3 to DS-0	<i>ds-fpc/pic/port:T1channel:DS-0 channel t1</i>
	Channelized DS-3 to DS-1	<i>t1-fpc/pic/port:channel t1</i>
	Channelized E1	<i>ds-fpc/pic/port:ds-0 channel e1</i>
	Channelized OC-3 to T1	
	Channelized OC-12 to DS-3	<i>t3-fpc/pic/port:channel</i>
	Channelized STM-1 to E1	<i>e1-fpc/pic/port:channel</i>
T1, T3, E1, E3	E1	<i>e1-fpc/pic/port</i>
	E3	<i>e3-fpc/pic/port</i>
	T1	<i>t1-fpc/pic/port</i>
	T3	<i>t3-fpc/pic/port</i>
Ethernet	Aggregated Ethernet	<i>ae-fpc/pic/port</i>
	Fast Ethernet	<i>fe-fpc/pic/port</i>
	Gigabit Ethernet	<i>ge-fpc/pic/port</i>
	10-Gigabit Ethernet	<i>ge-fpc/pic/port</i>
	Internal Ethernet	<i>fxp</i>
	Management Ethernet	<i>fxp</i>
Multilink	Frame Relay	<i>ml-fpc/pic/port</i>
	PPP	<i>ml-fpc/pic/port</i>
SONET/SDH	Aggregated SONET/SDH	<i>as-fpc/pic/port</i>
	SONET/SDH	<i>so-fpc/pic/port</i>
Other	Encryption	<i>es-fpc/pic/port:es</i>
	GRE tunnel	<i>gr-fpc/pic/port</i>
	IP-IP tunnel	<i>ip-fpc/pic/port</i>
	Loopback	<i>lo</i>

Display Real-Time Statistics about a Physical Interface

Purpose Displaying real-time statistics about a physical interface is useful when you need to narrow down possible causes of an interface problem. The **monitor** command checks for and displays common interface failures, indicates whether loopback is detected, and shows any increases in framing errors.



NOTE: If you are accessing the router from the console connection, make sure you set the CLI terminal type using the **set cli terminal** command.

Action To display real-time statistics about a physical interface, use the following JUNOS CLI operational mode command:

```
user@host> monitor interface interface-name
```

Sample Output

```
user@host> monitor interface so-0/0/0
router1                               Seconds: 19                               Time: 15:46:29

Interface: so-0/0/0, Enabled, Link is Up
Encapsulation: PPP, Keepalives, Speed: OC48
Traffic statistics:
    Input packets:                6045 (0 pps)                [11]
    Input bytes:                  6290065 (0 bps)              [13882]
    Output packets:               10376 (0 pps)               [10]
    Output bytes:                 10365540 (0 bps)            [9418]
Encapsulation statistics:
    Input keepalives:             1901                        [2]
    Output keepalives:            1901                        [2]
    NCP state: Opened
    LCP state: Opened
Error statistics:
    Input errors:                 0                          [0]
    Input drops:                 0                          [0]
    Input framing errors:         0                          [0]
    Policed discards:            0                          [0]
    L3 incompletes:              0                          [0]
    L2 channel errors:           0                          [0]
    L2 mismatch timeouts:        0                          [0]
    Carrier transitions:         1                          [0]
    Output errors:               0                          [0]
    Output drops:               0                          [0]
    Aged packets:               0                          [0]
Active alarms : None
Active defects: None
SONET error counts/seconds:
    LOS count                    1                          [0]
    LOF count                    1                          [0]
    SEF count                    1                          [0]
    ES-S                        0                          [0]
    SES-S                        0                          [0]
SONET statistics:
    BIP-B1                      458871                    [0]
    BIP-B2                      460072                    [0]
    REI-L                       465610                    [0]
    BIP-B3                      458978                    [0]
    REI-P                       458773                    [0]
Received SONET overhead:
    F1      : 0x00  J0      : 0x00  K1      : 0x00
    K2      : 0x00  S1      : 0x00  C2      : 0x00
```

```

C2(cmp) : 0x00 F2      : 0x00 Z3      : 0x00
Z4      : 0x00 S1(cmp) : 0x00
Transmitted SONET overhead:
F1      : 0x00 J0      : 0x01 K1      : 0x00
K2      : 0x00 S1      : 0x00 C2      : 0xcf
F2      : 0x00 Z3      : 0x00 Z4      : 0x00

```

Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'

What It Means The sample output displays real-time statistics about the physical interface (updating them every second), the amount that each field has changed since you started the command or since you cleared the counters by using the **C** key. It also checks for and displays common interface failures, such as SONET/SDH and T3 alarms, detected loopbacks, and increases in framing errors.

To control the output of the command while it is running, use the keys shown in Table 30.

Table 30: Monitor Interface Output Control Keys

Key	Action
N	Display information about the next interface. The monitor interface command scrolls through the physical or logical interfaces in the same order that they are displayed by the show interfaces terse command.
I	Display information about a different interface. The command prompts you for the name of a specific interface.
F	Freeze the display, halting the display of updated statistics.
T	Thaw the display, resuming the display of updated statistics.
C	Clear (zero) the current delta counters since monitor interface was started. It does not clear the cumulative counter.
Q	Stop the monitor interface command.

Check System Logging

Purpose By looking through the messages file for any entries pertaining to the interface that you are interested in, you can further investigate a problem with an interface.

Action To check system logging, use the following JUNOS CLI operational mode command:

```
user@host> show log messages | match interface-name
```

Sample Output

```
user@host> show log messages | match so-0/3/1  
May 2 12:10:58 router rpd[729]: RPD_ISIS_ADJDOWN: IS-IS lost L2 adjacency to ABC-CORE-RTR1 on  
so-0/3/1.0, reason: Interface Down  
May 2 12:11:27 router mib2d[575]: SNMP_TRAP_LINK_DOWN: ifIndex 25, ifAdminStatus up(1), ifOperStatus  
down(2), ifName so-0/3/1  
May 2 12:11:27 router rpd[729]: RPD_ISIS_ADJDOWN: IS-IS lost L2 adjacency to ABC-CORE-RTR1 on  
so-0/3/1.0, reason: Interface Down  
May 2 12:11:31 router rpd[729]: RPD_LDP_NBRDOWN: LDP neighbor 130.81.4.109 (so-0/3/1.0) is down
```

What It Means The sample output shows entries in the messages file pertaining to the SONET interface, **so-0/3/1**, and its Intermediate System-to-Intermediate System (IS-IS) adjacencies and Label Distribution Protocol (LDP) neighbors. The entries indicate that the interface went down on May 2 at 12:11:27, and that both the IS-IS adjacency and the LDP neighbor are down.

Chapter 11

Verify the IS-IS Protocol and Adjacencies

This chapter describes how to check whether the Intermediate System-to-Intermediate System (IS-IS) protocol is configured correctly on a Juniper Networks router and that the proper adjacencies are formed in a network. (See Table 31.)

Table 31: Checklist for Verifying the IS-IS Protocol and Adjacencies

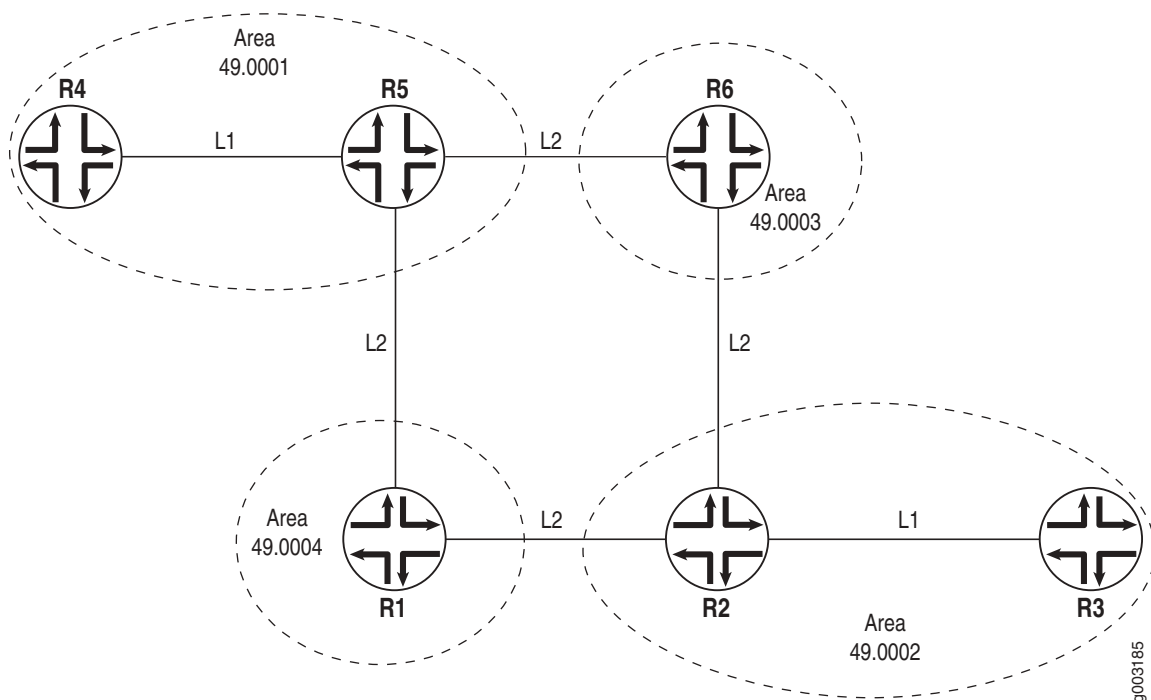
Verify the IS-IS Protocol and Adjacencies Tasks	Command or Action
Verify the IS-IS Configuration on a Router in a Network on page 112	
1. Check the Configuration of a Level 1/Level 2 Router on page 114	[edit protocols isis] show [edit protocols isis] run show isis interface [edit] edit interfaces [edit interfaces] show
2. Check the Configuration of a Level 1 Router on page 116	[edit protocols isis] show [edit protocols isis] run show isis interface [edit] edit interfaces [edit interfaces] show
3. Check the Configuration of a Level 2 Router on page 117	[edit protocols isis] show [edit protocols isis] run show isis interface [edit] edit interfaces [edit interfaces] show
Display the Status of IS-IS Adjacencies on page 119	
1. Verify Adjacent Routers on page 120	show isis adjacency
2. Examine a Route on page 121	show route <i>destination-prefix</i> show route detail <i>destination-prefix</i> show isis route <i>destination-prefix</i>
3. Examine the Forwarding Table on page 123	show route forwarding-table destination <i>destination-prefix</i>
4. Examine the Link-State Database on page 124	show isis database
5. Examine a Link-State Protocol Data Unit Header on page 126	show isis database extensive

Verify the IS-IS Configuration on a Router in a Network

Purpose For IS-IS to run on a router (intermediate system) in your network, you must enable IS-IS on the router, configure a network entity title (NET) on the loopback interface (lo0), and configure **family iso** on all interfaces on which you want to run IS-IS. When you enable IS-IS on a router, Level 1 and Level 2 are enabled by default.

Figure 7 illustrates an example of routers at different levels in an IS-IS topology.

Figure 7: Levels in an IS-IS Network Topology

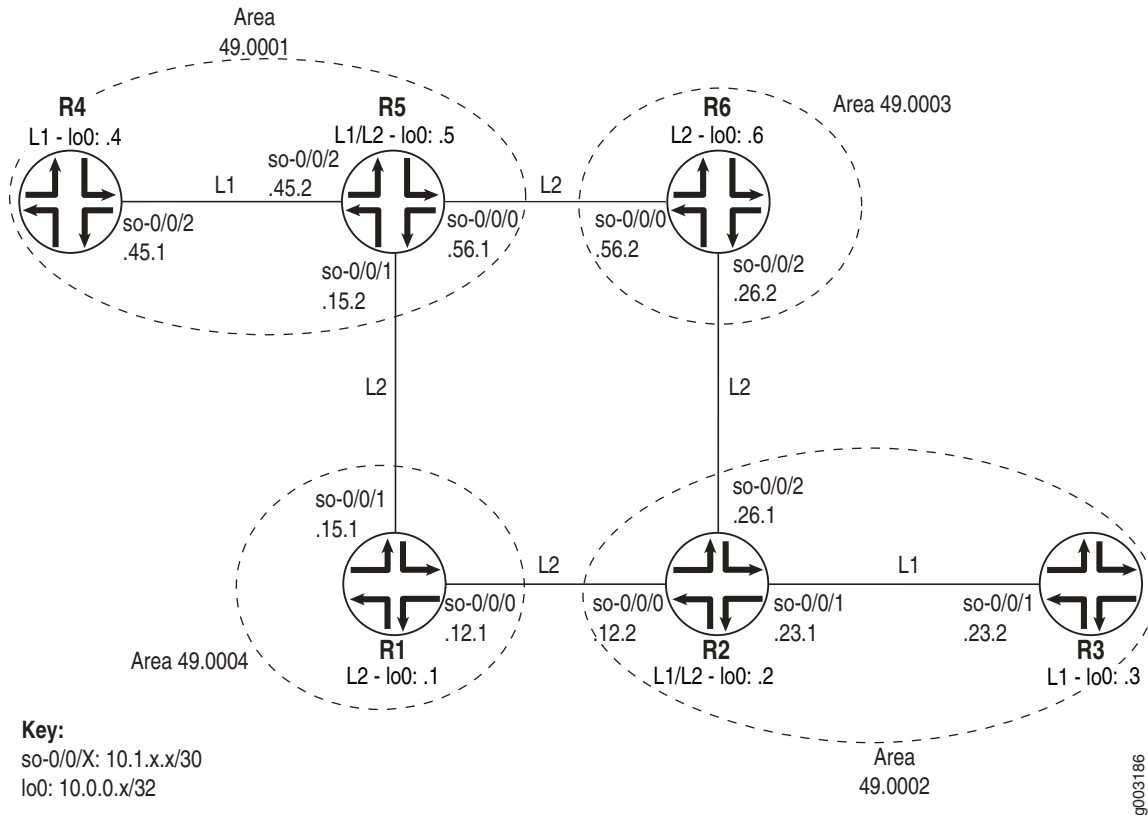


The network in Figure 7 is organized hierarchically and consists of Level 2, Level 1/Level 2, and Level 1 routers in one autonomous system (AS) divided into four areas: 49.0001, 49.0002, 49.0003, and 49.0004. The Level 2 routers route toward other autonomous systems. The Level 1/Level 2 routers route between areas and to other autonomous systems. The Level 1 routers route within an area, and when the destination is outside the local area, they route toward a Level1/Level2 system.

In the following steps, the configuration of the various types of routers is examined.

Figure 8 provides more details about the IS-IS network topology on page 112 so that you can verify the configuration output of the various routers.

Figure 8: IS-IS Network Topology with Details



Steps To Take To verify that IS-IS is configured correctly on routers at different levels, follow these steps:

1. Check the Configuration of a Level 1/Level 2 Router on page 114
2. Check the Configuration of a Level 1 Router on page 116
3. Check the Configuration of a Level 2 Router on page 117

Step 1: Check the Configuration of a Level 1/Level 2 Router

Action To verify the IS-IS configuration of a Level 1/Level 2 router in your network, enter the following JUNOS command-line interface (CLI) commands:

```
user@host# [edit protocols isis] show
user@host# [edit protocols isis] run show isis interface
user@host# [edit] edit interfaces
user@host# [edit interfaces] show
```

Sample output The following output is for an IS-IS configuration on R2, a Level 1/Level 2 router in the network shown in Figure 7:

```
[edit protocols isis]
user@R2# show
interface so-0/0/0.0 {
    level 2 metric 10;
    level 1 disable;
}
interface so-0/0/1.0 {
    level 2 disable;
    level 1 metric 10;
}
interface so-0/0/2.0 {
    level 2 metric 10;
    level 1 disable;
}
interface fxp0.0 {
    disable;
}
interface lo0.0;

[edit protocols isis]
user@R2# run show isis interface
IS-IS interface database:
Interface          L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
lo0.0              0  0x1 Passive           Passive          0/0
so-0/0/0.0         2  0x1 Disabled          Point to Point   10/10
so-0/0/1.0         3  0x1 Point to Point    Point to Point   10/10
so-0/0/2.0         2  0x1 Disabled          Point to Point   10/10

[edit interfaces]
user@R2# show
so-0/0/0 {
    unit 0 {
        family inet {
            address 10.1.12.2/30;
        }
        family iso;
    }
}
so-0/0/1 {
    unit 0 {
        family inet {
            address 10.1.23.1/30;
        }
        family iso;
    }
}
```

```

so-0/0/2 {
  unit 0 {
    family inet {
      address 10.1.26.1/30;
    }
    family iso;
  }
}
lo0 {
  unit 0 {
    family inet {
      address 10.0.0.2/32;
    }
    family iso {
      address 49.0002.1000.0000.0002.00;
    }
  }
}

```

What It Means The sample output shows a basic configuration of IS-IS on R2, a Level 1/Level 2 router. The basic configuration is at the [edit protocols isis] and [edit interfaces] hierarchy levels.

At the [edit protocols isis] level, five interfaces are included: so-0/0/0, so-0/0/1, so-0/0/2, fxp0, and the loopback (lo0) interface. Two interfaces, so-0/0/0.0 and so-0/0/2.0, have Level 1 disabled, making them Level 2 interfaces. One interface, so-0/0/1.0, has Level 2 disabled, making it a Level 1 interface. The management interface (fxp0) is disabled so that IS-IS packets are not sent over it, and the loopback interface (lo0) is included because it becomes a point of connection from the router to the IS-IS network.

At the [edit interfaces] hierarchy level, all of the interfaces included in the [edit protocols isis] hierarchy level are configured with family iso, and the loopback (lo0) interface is configured with the NET address 49.0002.1000.0000.0002.00. Every router in an IS-IS network must have at least one NET address that identifies a point of connection to the IS-IS network. The NET address is generally configured on the loopback (lo0) interface. Routers that participate in multiple areas can have multiple NET addresses.

See the *JUNOS Routing Protocols Configuration Guide* for more information on configuring IS-IS on a router.

Step 2: Check the Configuration of a Level 1 Router

Action To check the configuration of a Level 1 router, enter the following CLI commands:

```
user@host# [edit protocols isis] show
user@host# [edit protocols isis] run show isis interface
user@host# [edit] edit interfaces
user@host# [edit interfaces] show
```

Sample Output: The following sample output is for R4, a Level 1 router in the network shown in Figure 7:

```
[edit protocols isis]
user@R4# show
level 2 disable;
interface so-0/0/2.0 {
    level 1 metric 10;
}
interface fxp0.0 {
    disable;
}
interface lo0.0;

[edit protocols isis]
user@R4# run show isis interface
IS-IS interface database:
Interface          L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
lo0.0              0   0x1 Passive           Passive         0/0
so-0/0/2.0         1   0x1 Point to Point    Disabled        10/10

[edit interfaces]
user@R4# show
so-0/0/2 {
    unit 0 {
        family inet {
            address 10.1.45.1/30;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.0.0.4/32;
        }
        family iso {
            address 49.0001.1000.0000.0004.00;
        }
    }
}
```

What It Means The sample output shows a basic configuration of IS-IS on R4, a Level 1 router. The basic configuration is at the [edit protocols isis] and [edit interfaces] hierarchy levels.

At the [edit protocols isis] hierarchy level, three interfaces are included: so-0/0/2.0, fxp0, and the loopback (lo0) interface. Level 2 is disabled on the router, making it a Level 1 router that sends packets within its local area, 49.0001. When a packet destination is outside the local area, R4 establishes an adjacency with the nearest Level 1/Level 2 router (R5) that forwards the packets. For more information on adjacencies, see “Display the Status of IS-IS Adjacencies” on page 119.

One interface, **so-0/0/2.0**, is configured for IS-IS. The management interface (**fxp0**) is disabled so that IS-IS packets are not sent over it, and the loopback interface (**lo0**) is included because it becomes a point of connection from the router to the IS-IS network.

At the **[edit interfaces]** hierarchy level, the interface included in the **[edit protocols isis]** hierarchy level is also configured with **family iso**, and the loopback (**lo0**) interface is configured with the NET address of **49.0001.1000.0000.0004.00**. Every router in an IS-IS network must have at least one NET address that identifies a point of connection to the IS-IS network. The NET address is generally configured on the loopback (**lo0**) interface. Routers that participate in multiple areas can have multiple NET addresses.

See the *JUNOS Routing Protocols Configuration Guide* for more information on configuring IS-IS on a router.

Step 3: Check the Configuration of a Level 2 Router

Action To check the configuration of a Level 2 router, enter the following CLI commands:

```
user@host# [edit protocols isis] show
user@host# [edit protocols isis] run show isis interface
user@host# [edit] edit interfaces
user@host# [edit interfaces] show
```

Sample Output: The following sample output is for R6, a Level 2 router in the network shown in Figure 7:

```
[edit protocols isis]
user@R6# show
level 1 disable;
interface so-0/0/0.0 {
    level 2 metric 10;
}
interface so-0/0/2.0 {
    level 2 metric 10;
}
interface fxp0.0 {
    disable;
}
interface lo0.0;

[edit protocols isis]
user@R6# run show isis interface
IS-IS interface database:
Interface          L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
lo0.0              0  0x1 Passive           Passive         0/0
so-0/0/0.0         2  0x1 Disabled         Point to Point  10/10
so-0/0/2.0         2  0x1 Disabled         Point to Point  10/10
```

```

[edit interfaces]
user@R6# show
so-0/0/0 {
    unit 0 {
        family inet {
            address 10.1.56.2/30;
        }
        family iso;
    }
}
so-0/0/2 {
    unit 0 {
        family inet {
            address 10.1.26.2/30;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.0.0.6/32;
        }
        family iso {
            address 49.0003.1000.0000.0006.00;
        }
    }
}

```

What It Means The sample output shows a basic configuration of IS-IS on R6, a Level 2 router. The basic configuration is at the `[edit protocols isis]` and `[edit interfaces]` hierarchy levels.

At the `[edit protocols isis]` level, four interfaces are included: `so-0/0/0.0`, `so-0/0/2.0`, `fxp0`, and the loopback (`lo0`) interface. Level 1 is disabled on the two SONET interfaces, making this a Level 2 router that routes between areas and towards other ASs. The management interface (`fxp0`) is disabled so that IS-IS packets are not sent over it, and the loopback interface (`lo0`) is included because it becomes a point of connection from the router to the IS-IS network.

At the `[edit interfaces]` hierarchy level, the interfaces included in the `[edit protocols isis]` hierarchy level are also configured with `family iso`, and the loopback (`lo0`) interface is configured with the NET address of `49.0003.1000.0000.0006.00`. Every router in an IS-IS network must have at least one NET address that identifies a point of connection to the IS-IS network. The NET address is generally configured on the loopback (`lo0`) interface. Routers that participate in multiple areas can have multiple NET addresses.

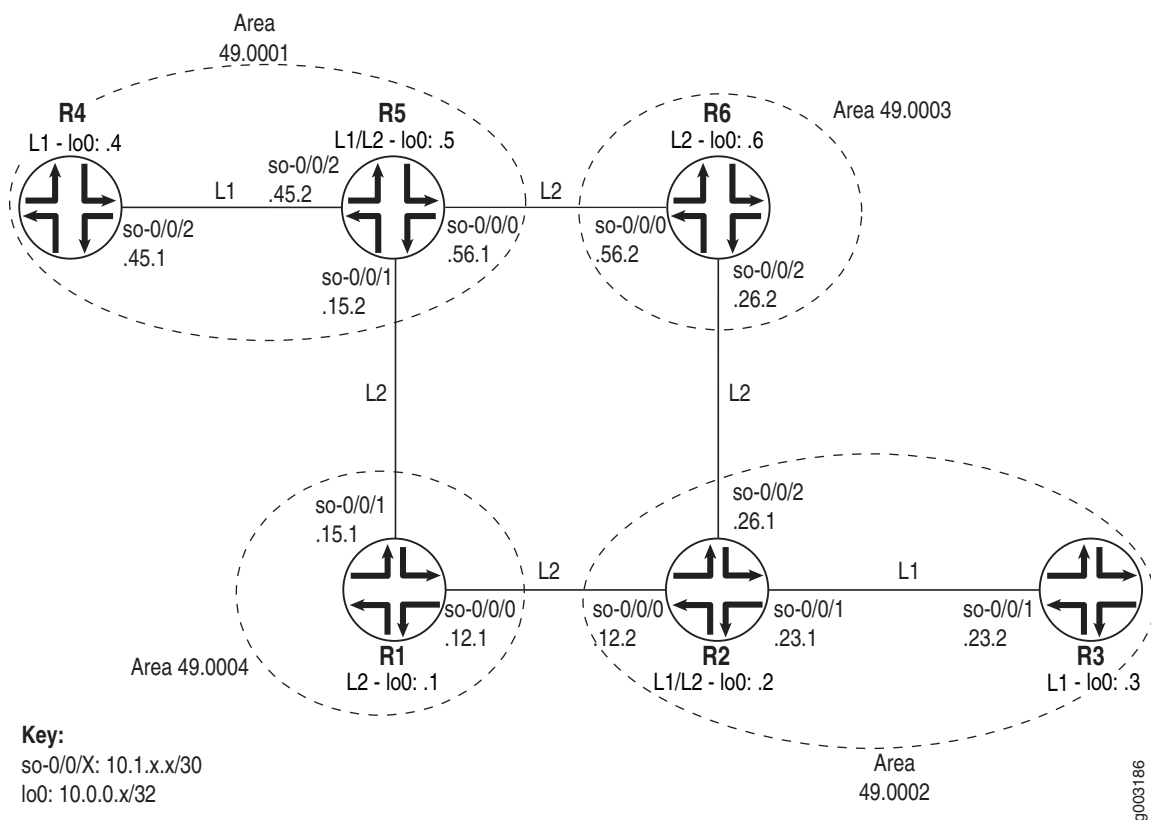
See the *JUNOS Routing Protocols Configuration Guide* for more information on configuring IS-IS on a router.

Display the Status of IS-IS Adjacencies

Purpose Assuming that all the routers are correctly configured for IS-IS, you can verify which neighbors are adjacent and able to exchange IS-IS data. In addition, you can examine the set of routes installed in the forwarding table to verify that the routing protocol process (rpd) has relayed the correct information into the forwarding table.

Figure 9 illustrates the example IS-IS topology used for the procedures in this section.

Figure 9: IS-IS Network Topology



The network consists of Level 1 and Level 2 adjacencies. Level 1 adjacencies are within areas 49.0001 and 49.0002. Level 2 adjacencies occur between all directly connected Level 2 routers regardless of which area they are in. For example, R5 is in area 49.0001, R6 is in area 49.0003, R1 is in area 49.0004, and R2 is in area 49.0002. The network in Figure 9 should have the following adjacencies:

- Level 2 adjacencies between all directly connected Level 2 routers (R1, R2, R5, and R6).
- Level 1 adjacencies between routers in area 49.0001 (R4 and R5) and between routers in area 49.0002 (R2 and R3).

Steps To Take To verify that routers are adjacent and able to exchange IS-IS data, follow these steps:

1. Verify Adjacent Routers on page 120
2. Examine a Route on page 121
3. Examine the Forwarding Table on page 123
4. Examine the Link-State Database on page 124
5. Examine a Link-State Protocol Data Unit Header on page 126

Step 1: Verify Adjacent Routers

Action To verify that routers are adjacent and able to exchange IS-IS data, enter the following CLI operational mode command:

```
user@host> show isis adjacency
```

Sample Output The following sample output shows the adjacencies that formed for all routers shown in Figure 9:

```
user@R1> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
so-0/0/0.0     R2          2 Up         19
so-0/0/1.0     R5          2 Up         18

user@R2> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
so-0/0/0.0     R1          2 Up         19
so-0/0/1.0     R3          1 Up         26
so-0/0/2.0     R6          2 Up         21

user@R3> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
so-0/0/1.0     R2          1 Up         24

user@R4> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
so-0/0/2.0     R5          1 Up         23

user@R5> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
so-0/0/0.0     R6          2 Up         22
so-0/0/1.0     R1          2 Up         20
so-0/0/2.0     R4          1 Up         20

user@R6> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
so-0/0/0.0     R5          2 Up         21
so-0/0/2.0     R2          2 Up         20
```

What It Means The sample output shows the adjacencies that formed in the network illustrated in Figure 9. The Level 1/Level 2 routers (R2 and R5) formed Level 1 adjacencies with Level 1 routers (R3 and R4), and Level 2 adjacencies with the Level 2 routers (R1 and R6). To view the status of the adjacency, examine the **State** column. In this example, all adjacencies in the network are up.

If the state is not **Up** for a particular neighbor, you must first examine the IS-IS configuration for the particular interface. Make sure that the NET address is correct and that the loopback (lo0) interface is configured. Use the **show isis interface** or **show isis interface detail** command to display the IS-IS parameters for all interfaces configured with IS-IS. With these two commands, you can see which interfaces are configured for IS-IS, whether they are configured for Level 1 or Level 2, the IS-IS metric, and other IS-IS information.

Step 2: Examine a Route

Purpose You can determine the cost associated with a route and the selection of a route. In this step, the path from router **R5** to **R3** is examined in two situations. In the first example, all metrics in the network are set to the default of 10; in the second example, all metrics on a transit router (**R6**) are set to 5.

Action To examine a route in an IS-IS network, enter one or all of the following CLI commands:

```
user@host> show route destination-prefix
user@host> show route detail destination-prefix
user@host> show isis route destination-prefix
```

Sample Output 1 The following sample output shows the route from **R5** to **R3** when all metrics across interfaces are set to the default of 10:

```
user@R5> show route 10.0.0.3

inet.0: 28 destinations, 28 routes (28 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.3/32          *[IS-IS/18] 00:02:00, metric 30
                    to 10.1.56.2 via so-0/0/0.0
                    > to 10.1.15.1 via so-0/0/1.0

user@R5> show route detail 10.0.0.3

inet.0: 28 destinations, 28 routes (28 active, 0 holddown, 0 hidden)
10.0.0.3/32 (1 entry, 1 announced)
  *IS-IS Preference: 18
    Level: 2
    Next hop: 10.1.56.2 via so-0/0/0.0
    Next hop: 10.1.15.1 via so-0/0/1.0, selected
    State: <Active Int>
    Age: 34:29      Metric: 30
    Task: IS-IS
    Announcement bits (1): 0-KRT
    AS path: I

user@R5> show isis route 10.0.0.3
  IS-IS routing table                      Current version: L1: 241 L2: 243
Prefix          L Version Metric Type Interface      Via
10.0.0.3/32      2      243    30 int  so-0/0/0.0      R6
                                   so-0/0/1.0      R1
```

The following sample output shows the IS-IS configuration for transit router R6 with the metric on so-0/0/2.0 changed from the default of 10 to 5:

```
[edit protocols isis]
user@R6# show
level 1 disable;
interface so-0/0/0.0 {
level 2 metric 10;
}
interface so-0/0/2.0 {
    level 2 metric 5;
}
interface fxp0.0 {
    disable;
}
interface lo0.0;
```

Sample Output 2 The following sample output shows the route from R5 to R3 after the metric on R6 is changed from the default of 10 to 5:

```
user@R5> show route 10.0.0.3

inet.0: 28 destinations, 28 routes (28 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.3/32          *[IS-IS/18] 00:00:10, metric 25
                    > to 10.1.56.2 via so-0/0/0.0

user@R5> show route detail 10.0.0.3

inet.0: 28 destinations, 28 routes (28 active, 0 holddown, 0 hidden)
10.0.0.3/32 (1 entry, 1 announced)
    *IS-IS Preference: 18
      Level: 2
    Next hop: 10.1.56.2 via so-0/0/0.0, selected
    State: <Active Int>
    Age: 4:57      Metric: 25
    Task: IS-IS
    Announcement bits (1): 0-KRT
    AS path: I

user@R5> show isis route detail 10.0.0.3
  IS-IS routing table                               Current version: L1: 250 L2: 257
Prefix          L Version Metric Type Interface    Via
10.0.0.3/32      2    257    25 int so-0/0/0.0    R6
```

What It Means Sample output 1 shows the cost for each route and the preferred next hop. In this example, there are two next hops, one through R1 and the other through R6. Both have an equal cost (30) to the destination. The cost is indicated in the **Metric** field. The preferred next hop is randomly chosen. In this case, the preferred next hop is through R1, interface so-0/0/1.0. In the output for the **show route** command, the selected next hop is indicated by a forward arrow (>). With the **show route detail** command, the next hop is indicated by the key word **selected**. The output for the **show isis route** command shows the selected interface and indicates that the IS-IS protocol is building the correct routing table from the link-state database.

After the metric on **R6** is changed to a lower value, sample output 2 shows a different cost and next hop. With IS-IS, routes with lower total path metrics are preferred over those with higher path metrics. The path through **R6** (**so-0/0/0.0**) is now lower (25) than the path through **R1** (**so-0/0/1.0**) (30). In the output for the **show route** command, the lower cost (25) is indicated in the **Metric** field, and the preferred path is indicated by the forward arrow (>) and the keyword **selected**. The output for the **show isis route** command shows the selected interface and indicates that the IS-IS protocol is building the correct routing table from the link-state database.

In general, the output for the **show route** commands shows all active entries in the routing table. The information displayed includes the name of the routing table (**inet.0**), the number of destinations for which there are routes in the routing table (28), how the route was learned, and the route preference value, such as **[IS-IS/18]**. In addition, any metric associated with the route (**metric 30**), and the name of the interface through which the route was learned are displayed.

Step 3: Examine the Forwarding Table

Purpose You can display the set of routes installed in the forwarding table to verify that the routing protocol process (**rpd**) has relayed the correct information into the forwarding table. This is especially important when there are network problems, such as connectivity. In this procedure, you verify that the routes displayed in Step 2 appear in the forwarding table for router **R5**.

Action To examine the forwarding table for a router, enter the following CLI command:

```
user@host> show route forwarding-table destination destination-prefix
```

Sample Output

```
user@R5> show route forwarding-table destination 10.0.0.3
Routing table: inet
Internet:
Destination          Type RtRef Next hop          Type Index NhRef Netif
10.0.0.3/32          user   0 10.1.15.0          ucst  285   7 so-0/0/1.0

user@R5> show route forwarding-table destination 10.0.0.3
Routing table: inet
Internet:
Destination          Type RtRef Next hop          Type Index NhRef Netif
10.0.0.3/32          user   0 10.1.56.0          ucst  281   9 so-0/0/0.0
```

What It Means The sample output shows the selected next hop between routers **R5** and **R3** sent from the **inet** routing table and installed into the forwarding table. The first instance shows the route through **R1** and the second instance shows the route through **R6**. In both instances, the preferred route displayed in Step 2 is installed in the forwarding table.

In general, the sample output includes the destination address and destination type, the next-hop address and next-hop type, the number of references to the next hop, an index number into an internal next-hop database, and the interface used to reach the next hop.

Step 4: Examine the Link-State Database

Purpose You can examine the entire link-state database to obtain important information about the network. In this step, you verify that the key word **Attached** appears in the output of the Level 1 routers (R3 and R4) and the Level 1/Level 2 routers (R2 and R5).

Action To examine the link-state database for routers at different levels of the network, enter the following CLI command:

```
user@host> show isis database
```

Sample Output

```
user@R1> show isis database
IS-IS level 1 link-state database:
  0 LSPs

IS-IS level 2 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R1.00-00                0x1b   0xd4b5      583 L1 L2
R2.00-00                0x1d   0x1a4f      1131 L1 L2
R5.00-00                0x1b   0x6245      883 L1 L2
R6.00-00                0x20   0xf7c       919 L1 L2
  4 LSPs

user@R2> show isis database
IS-IS level 1 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R2.00-00                0x56   0xbd8c      862 L1 L2 Attached
R3.00-00                0x58   0xaca1      463 L1
  2 LSPs

IS-IS level 2 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R1.00-00                0x57   0x3e10      871 L1 L2
R2.00-00                0x5e   0x9790     1150 L1 L2
R5.00-00                0x5a   0xadba     1162 L1 L2
R6.00-00                0x56   0xa2b2      484 L1 L2
  4 LSPs

user@R3> show isis database
IS-IS level 1 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R2.00-00                0x56   0xbd8c      792 L1 L2 Attached
R3.00-00                0x58   0xaca1      397 L1
  2 LSPs

IS-IS level 2 link-state database:
  0 LSPs

[edit protocols isis]
user@R4> show isis database
IS-IS level 1 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R4.00-00                0x56   0x34b8     1125 L1
R5.00-00                0x57   0x22bb      795 L1 L2 Attached
  2 LSPs

IS-IS level 2 link-state database:
  0 LSPs
```

```

user@R5> show isis database
IS-IS level 1 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R4.00-00              0x56   0x34b8   1071 L1
R5.00-00              0x57   0x22bb    745 L1 L2 Attached
  2 LSPs

IS-IS level 2 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R1.00-00              0x57   0x3e10    707 L1 L2
R2.00-00              0x5e   0x9790    982 L1 L2
R5.00-00              0x5a   0xadba   1002 L1 L2
R6.00-00              0x57   0xa0b3   1064 L1 L2
  4 LSPs

user@R6> show isis database
IS-IS level 1 link-state database:
  0 LSPs

IS-IS level 2 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R1.00-00              0x1b   0xd4b5    728 L1 L2
R2.00-00              0x1c   0x1c4e    604 L1 L2
R5.00-00              0x1b   0x6245   1032 L1 L2
R6.00-00              0x20   0xf7c    1072 L1 L2
  4 LSPs

```

What It Means The sample output shows the details of the Level 1 and Level 2 IS-IS databases for routers R1, R2, R3, R4, R5, and R6. Whether a router is configured for Level 1, Level 2 or Level 1/Level 2 is indicated by the type of IS-IS link-state database(s) in the output for the **show isis database** command for that router. For example, R3 and R4 are Level 1 routers because they do not have LSPs in the Level 2 link-state database, and the R1 and R6 are Level 2 routers because they do not have LSPs in the Level 1 link-state database. R2 and R5 have LSPs in both Level 1/Level 2 link-state databases, indicating they are Level 1/Level 2 routers.

In addition, the output for R2 shows that it is a Level 1/Level 2 router because it has R3 in its Level 1 database, while R3 does not have the L2 notation in the **Attributes** field, indicating that it is configured for Level 1.

The details of the Level 2 IS-IS database are the same for all Level 2 routers in the network.

The key word **Attached** appears in the Level 1 link-state database for R2, R3, R4, and R5, indicating that the Level 2 routers (R2 and R5) can communicate with other Level 2 systems and act as gateways for the Level 1 routers (R3 and R4) in their respective areas. If the links that connect R2 and R5 to the other Level 2 routers (R1 and R6) are broken, the key word **Attached** will not appear in the output because R2 and R5 will no longer act as gateways for the Level 1 routers.

Step 5: Examine a Link-State Protocol Data Unit Header

Purpose By using the key word **extensive**, you can examine each header field of a link-state protocol data unit (LSP) and gather important details about the network. In this step, you examine the IS-IS database for router R2.

Action To examine an LSP header, enter the following CLI command:

```
user@host> show isis database extensive
```

Sample Output

```
user@R2> show isis database extensive
IS-IS level 1 link-state database:
[...Output truncated...]
Header: LSP ID: R2.00-00, Length: 139 bytes
  Allocated length: 1492 bytes, Router ID: 10.0.0.2
  Remaining lifetime: 1071 secs, Level: 1, Interface: 0
  Estimated free bytes: 1353, Actual free bytes: 1353
  Aging timer expires in: 1071 secs
  Protocols: IP, IPv6

Packet: LSP ID: R2.00-00, Length: 139 bytes, Lifetime : 1198 secs
  Checksum: 0xbb8d, Sequence: 0x57, Attributes: 0xb <L1 L2 Attached>
  NLPID: 0x83, Fixed length: 27 bytes, Version: 1, Sysid length: 0 bytes
  Packet type: 18, Packet version: 1, Max area: 0
[...Output truncated...]
Header: LSP ID: R3.00-00, Length: 139 bytes
  Allocated length: 284 bytes, Router ID: 10.0.0.3
  Remaining lifetime: 823 secs, Level: 1, Interface: 68
  Estimated free bytes: 145, Actual free bytes: 145
  Aging timer expires in: 823 secs
  Protocols: IP, IPv6

Packet: LSP ID: R3.00-00, Length: 139 bytes, Lifetime : 1198 secs
  Checksum: 0xaa2, Sequence: 0x59, Attributes: 0x1 <L1>
  NLPID: 0x83, Fixed length: 27 bytes, Version: 1, Sysid length: 0 bytes
  Packet type: 18, Packet version: 1, Max area: 0
[...Output truncated...]
IS-IS level 2 link-state database:
[...Output truncated...]
Header: LSP ID: R1.00-00, Length: 194 bytes
  Allocated length: 284 bytes, Router ID: 10.0.0.1
  Remaining lifetime: 398 secs, Level: 2, Interface: 67
  Estimated free bytes: 145, Actual free bytes: 90
  Aging timer expires in: 398 secs
  Protocols: IP, IPv6

Packet: LSP ID: R1.00-00, Length: 194 bytes, Lifetime : 1196 secs
  Checksum: 0x3e10, Sequence: 0x57, Attributes: 0x3 <L1 L2>
  NLPID: 0x83, Fixed length: 27 bytes, Version: 1, Sysid length: 0 bytes
  Packet type: 20, Packet version: 1, Max area: 0
[...Output truncated...]
Header: LSP ID: R2.00-00, Length: 236 bytes
  Allocated length: 1492 bytes, Router ID: 10.0.0.2
  Remaining lifetime: 677 secs, Level: 2, Interface: 0
  Estimated free bytes: 1256, Actual free bytes: 1256
  Aging timer expires in: 677 secs
  Protocols: IP, IPv6
```

```

Packet: LSP ID: R2.00-00, Length: 236 bytes, Lifetime : 1198 secs
Checksum: 0x9790, Sequence: 0x5e, Attributes: 0x3 <L1 L2>
NLPID: 0x83, Fixed length: 27 bytes, Version: 1, Sysid length: 0 bytes
Packet type: 20, Packet version: 1, Max area: 0
[...Output truncated...]

```

What It Means The sample output is from router **R2** and shows extensive information about the Level 1 and Level 2 link-state databases, which are identical across all Level 2 routers. Level 1 routers only include information for the Level 1 link-state database, which is identical to the Level 1 information in the output for a Level 2 router. The sample output shows the details of four LSPs. Level 1 and Level 2 LSPs include identical types of information except for the packet type. A Level 2 LSP has a packet type of **20** and a Level 1 LSP has a packet type of **18**.

The first three examples illustrate different values in the **Attributes** field:

- **<L1 L2>** and the key word **Attached** in the first example, **R2.00-00**, indicate that router **R2** is a Level 1/Level 2 router acting as a gateway for Level 1 routers.
- **<L1>** in the second example, **R3.00-00**, indicates that **R3** is a Level 1 router.
- **<L1 L2>** in the third example, **R1.00-00**, indicates that **R1** is a Level 2 router (the fourth example is also for a Level 2 router).

The fourth example, **R2.00-00**, is included to show that a Level 1/Level 2 router appears in both the Level 1 and the Level 2 link-state databases. Note that in the Level 2 link-state database, the key word **Attached** is not included in the **Attributes** field.

It is useful to examine one LSP in greater detail. The third LSP, **R1.00-00**, was originated by **R1** as indicated by the **LSP ID** field. **R1** is the hostname of the router. The first **00** indicates that the LSP is for the router itself, and the final **00** denotes that the LSP is not fragmented. Both IPv4 and IPv6 are supported by this router, as indicated in the **Protocols** field. The **Attributes** field shows that the router is capable of both Level 1 and Level 2 routing, is not connected to Level 1 routers in other areas, and is not overloaded. The key words **Attached** and **Overloaded** would appear in the **Attributes** field if this were the case. The remaining lifetime of the LSP is 1196 seconds. IS-IS lifetimes start at a configured age (1200 seconds by default) and count down to zero.

Chapter 12

Verify the OSPF Protocol and Neighbors

This chapter describes how to check whether the Open Shortest Path First protocol (OSPF) is configured correctly on a Juniper Networks router, the proper adjacencies are formed in a network, and the appropriate link-state advertisements (LSAs) are flooded throughout different parts of the OSPF autonomous system (AS). (See Table 32.)

Table 32: Checklist for Verifying the OSPF Protocol and Neighbors

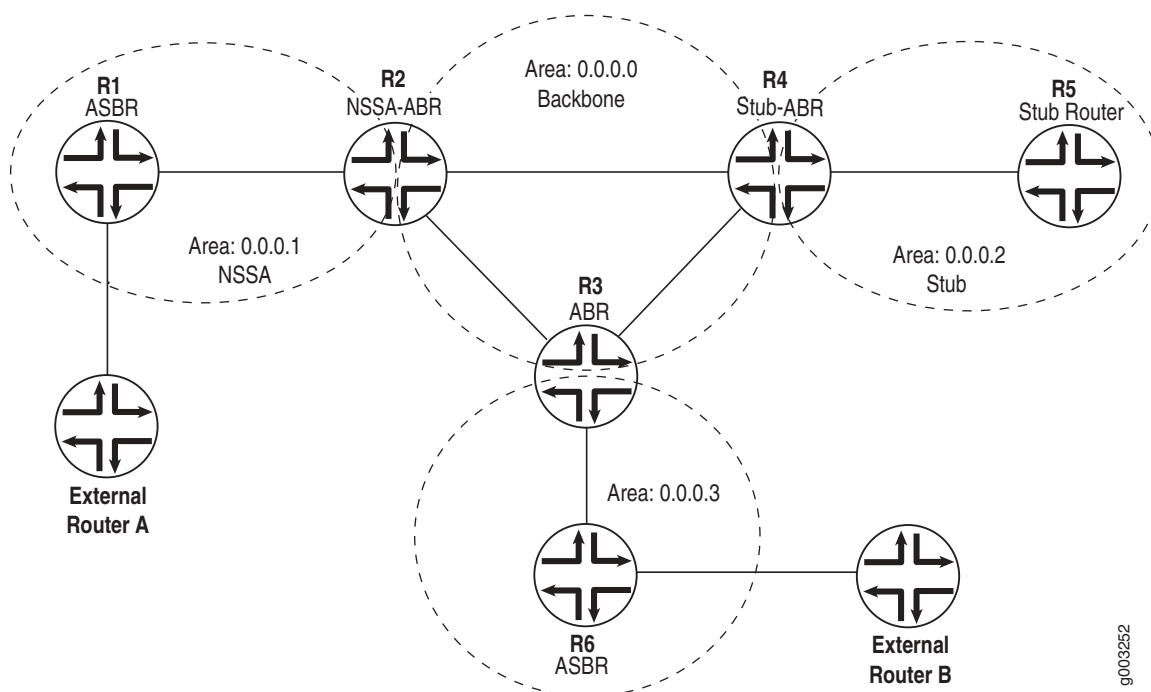
Verify the OSPF Protocol and Neighbors Tasks	Command or Action
Verify the OSPF Protocol on page 130	
1. Check OSPF on an ASBR on page 132	show configuration show ospf interface
2. Check OSPF on an ABR on page 135	show configuration show ospf interface
3. Check OSPF on a Stub Router on page 139	show configuration show ospf interface
Check OSPF Neighbors on page 141	
1. Verify OSPF Neighbors on page 142	show ospf neighbor
2. Examine the OSPF Link-State Database on page 144	show ospf database
3. Examine OSPF Routes on page 148	show route <i>destination-prefix</i> show ospf database
4. Examine the Forwarding Table on page 151	show route <i>destination-prefix</i> extensive show route forwarding-table destination <i>destination-prefix</i>
5. Examine Link-State Advertisements in Detail on page 152	
a. Examine a Type 1 Router LSA on page 152	show ospf database router extensive
b. Examine a Type 3 Summary LSA on page 153	show ospf database netsummary extensive
c. Examine a Type 4 ASBR Summary LSA on page 154	show ospf database asbrsummary extensive
d. Examine a Type 5 AS External LSA on page 155	show ospf database extern extensive
e. Examine Type 7 NSSA External LSA on page 156	show ospf database nssa extensive

Verify the OSPF Protocol

Purpose For OSPF to run on a router in your network, you must include the interfaces that run OSPF at the `[edit protocols ospf]` hierarchy level and, for those interfaces, the `family inet` statement must be included at the `[edit interfaces interface-name unit logical-unit-number]` hierarchy level.

Figure 10 illustrates an example an OSPF autonomous system (AS) consisting of multiple areas and different types of OSPF routers.

Figure 10: Multi-Area OSPF Network Topology



The AS in Figure 10 is organized hierarchically around a backbone area, 0.0.0.0. Portions of the network are designated as separate areas: 0.0.0.1, 0.0.0.2, and 0.0.0.3. The backbone is the connecting point for all other areas, and each area must attach to the backbone in at least one location. OSPF is based on the concept of a link-state database in which each OSPF router attempts to form adjacencies with its OSPF neighbor. Once the adjacencies are in place, each router generates and floods LSAs into the network. The LSAs are placed into the link-state database on each router where the shortest path first (SPF) algorithm is calculated to find the best path to each end node in the network.

All non-backbone areas (0.0.0.1, 0.0.0.2, and 0.0.0.3) contain routers internal to that area (R1, R5, and R6) as well as a single area border router (ABR) (R2, R3, and R4). Internal routers generate LSAs within their area. The ABR translates these internal LSAs into summary LSAs that represent the LSAs within its non-backbone area and floods the summary LSAs to the backbone. The ABR is also responsible for generating summary LSAs that represent the backbone LSAs and injecting them into its attached areas. Because the ABR belongs to more than one area, it maintains a separate topological database for each area to which it is connected.

In Figure 10, the ABRs belong to different non-backbone areas. R2 is in area 0.0.0.1, a not-so-stubby area (NSSA); R3 is in area 0.0.0.3; and R4 is in area 0.0.0.2, a stub area.

The NSSA (0.0.0.1) consists of two routers: R1 and R2. An NSSA allows external routes to be flooded within its area. These routes are then leaked to other areas within the AS. However, external routes learned from other areas within the AS do not enter the NSSA.

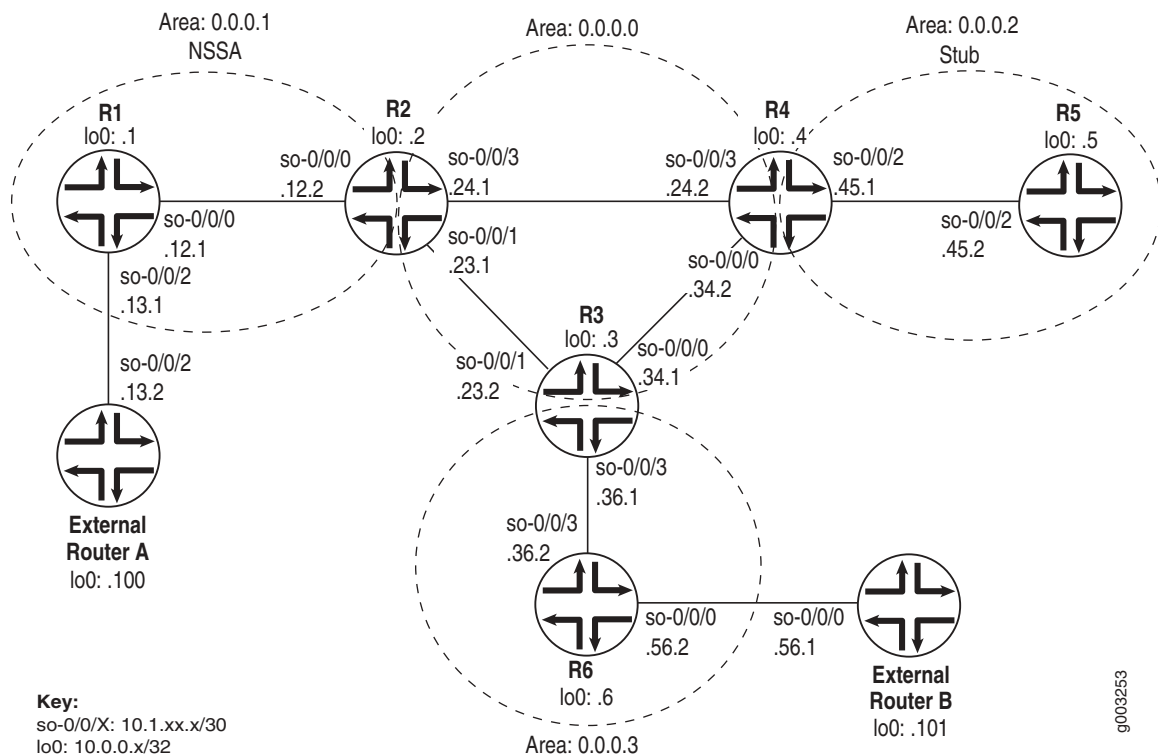
The stub area (0.0.0.2) consists of two routers: R4 and R5. A stub area does not allow external routes to be flooded within its area. A stub area is useful when much of the AS consists of external LSAs because it reduces the size of the topological database within the stub area and subsequently the amount of memory required by the routers in the area.

Area 0.0.0.3 is a non-backbone area consisting of two routers: R3 and R6.

External Routers A and B reside outside the AS. When an OSPF router exchanges routing information with routers in other ASs, that router is called an autonomous system boundary router (ASBR). The ASBRs shown in Figure 10 are R1 and R6.

Figure 11 provides interface and IP address information for the example OSPF network topology used for the procedures in this section.

Figure 11: OSPF Network Topology with Details



Steps To Take To verify that OSPF is configured correctly on routers in different areas of the network, follow these steps:

1. Check OSPF on an ASBR on page 132
2. Check OSPF on an ABR on page 135
3. Check OSPF on a Stub Router on page 139

Step 1: Check OSPF on an ASBR

Action To verify the OSPF configuration on an ASBR router in your network, enter the following JUNOS command-line interface (CLI) operational mode commands:

```
user@host> show configuration
user@host> show ospf interface
```

Sample Output The following sample output is for an OSPF configuration on R1, an ASBR router shown in Figure 11:

```
user@R1> show configuration
[...Output truncated...]
interfaces {
  so-0/0/0 {
    unit 0 {
      family inet {
        address 10.1.12.1/30;
      }
    }
  }
  so-0/0/2 {
    unit 0 {
      family inet {
        address 10.1.13.1/30;
      }
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 10.0.0.1/32;
      }
    }
  }
}
routing-options {
  static {
    [...Output truncated...]
    route 10.0.0.100/32 next-hop 10.1.13.2;
  }
  router-id 10.0.0.1;
}
```

```

protocols {
  ospf {
    export export-to-ospf;
    area 0.0.0.1 {
      nssa;
      interface so-0/0/0.0;
      interface lo0.0 {
        passive;
      }
    }
  }
}
policy-options {
  policy-statement export-to-ospf {
    term external-router {
      from {
        route-filter 10.0.0.100/32 exact;
      }
      then accept;
    }
  }
}

```

```

user@R1> show ospf interface

```

Interface	State	Area	DR ID	BDR ID	Nbrs
lo0.0	DRother	0.0.0.1	0.0.0.0	0.0.0.0	0
so-0/0/0.0	PtToPt	0.0.0.1	0.0.0.0	0.0.0.0	1

The following sample output is for an OSPF configuration on R6, an ASBR router shown in Figure 11:

```

user@R6> show configuration
[...Output truncated...]
interfaces {
  so-0/0/0 {
    unit 0 {
      family inet {
        address 10.1.56.2/30;
      }
    }
  }
  so-0/0/3 {
    unit 0 {
      family inet {
        address 10.1.36.2/30;
      }
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 10.0.0.6/32;
      }
    }
  }
}

```

```

routing-options {
  static {
[...Output truncated...]
    route 10.0.0.101/32 next-hop 10.1.56.1;
  }
  router-id 10.0.0.6;
}
protocols {
  ospf {
    export export-to-ospf;
    area 0.0.0.3 {
      interface so-0/0/3.0;
      interface lo0.0 {
        passive;
      }
    }
  }
}
policy-options {
  policy-statement export-to-ospf {
    term external-router {
      from {
        route-filter 10.0.0.101/32 exact;
      }
      then accept;
    }
  }
}

```

```

user@R6> show ospf interface
Interface      State      Area      DR ID      BDR ID      Nbrs
lo0.0          DRother   0.0.0.3    0.0.0.0    0.0.0.0      0
so-0/0/3.0     PtToPt    0.0.0.3    0.0.0.0    0.0.0.0      1

```

What It Means The sample output shows a basic OSPF configuration at the [edit protocols ospf] and [edit interfaces] hierarchy levels on the R1 and R6 ASBR routers. In addition, both routers have an export policy, **export-to-ospf**, configured. The export policy allows external routes to be injected into the OSPF database and communicated throughout the AS.

R1 has two interfaces included at the [edit protocols ospf] hierarchy level: so-0/0/3 and the loopback interface (lo0). Both interfaces have the **family inet** statement included at the [edit interfaces] hierarchy level and are in area 0.0.0.1. Area 0.0.0.1 is attached to the backbone through R2, an ABR.

In addition, R1 has the **nssa** statement included at the [edit protocols ospf] hierarchy level indicating that it is an ASBR running in an NSSA. An NSSA allows external routes from outside the AS to be flooded within it. In this instance, the routes learned from external router B through the export policy **export-to-ospf** are injected into the R1 OSPF database and communicated throughout the AS. For more information on OSPF routes, see “Examine OSPF Routes” on page 148.

R6 has two interfaces included at the [edit protocols ospf] hierarchy level: so-0/0/3 and the loopback interface (lo0). Both interfaces have the **family inet** statement included at the [edit interfaces] hierarchy level and are in area 0.0.0.3. Area 0.0.0.3 is attached to the backbone through R3, an ABR. In addition, external router B is attached to R6 which has the export policy **export-to-ospf** configured. The export policy allows external routes to be injected into the R6 OSPF database and communicated throughout the AS.

Both routers (R1 and R6) have the router ID configured manually to avoid possible problems when the OSPF router ID (RID) changes: for example, when multiple loopback addresses are configured. The RID uniquely identifies the router within the OSPF network. It is transmitted within the LSAs used to populate the link-state database and calculate the shortest-path tree. In a link-state network, it is important that two routers do not share the same RID value, otherwise IP routing problems may occur.

An ASBR exchanges routing information with routers in other autonomous systems. ASBRs advertise externally learned routes throughout the AS. With the exception of routers in stub areas, any router in the AS—an internal router, an area border router, or a backbone router—can be an ASBR.

See the *JUNOS Routing Protocols Configuration Guide* for more information on configuring OSPF on a router.

Step 2: Check OSPF on an ABR

Action To verify the OSPF configuration on an ABR router in your network, enter the following JUNOS CLI operational mode commands:

```
user@host> show configuration
user@host> show ospf interface
```

Sample Output The following sample output is for an OSPF configuration on R2, an NSSA ABR shown in Figure 11:

```
user@R2> show configuration
[...Output truncated...]
interfaces {
  so-0/0/0 {
    unit 0 {
      family inet {
        address 10.1.12.2/30;
      }
    }
  }
  so-0/0/1 {
    unit 0 {
      family inet {
        address 10.1.23.1/30;
      }
    }
  }
  so-0/0/3 {
    unit 0 {
      family inet {
        address 10.1.24.1/30;
      }
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 10.0.0.2/32;
      }
    }
  }
}
```

```

}
routing-options {
}
  router-id 10.0.0.2;
}
protocols {
  ospf {
    area 0.0.0.1 {
      nssa {
        default-lsa default-metric 10;
      }
      interface so-0/0/0.0;
    }
    area 0.0.0.0 {
      interface so-0/0/3.0;
      interface so-0/0/1.0;
      interface lo0.0 {
        passive;
      }
    }
  }
}
}

```

```

user@R2> show ospf interface

```

Interface	State	Area	DR ID	BDR ID	Nbrs
lo0.0	DRother	0.0.0.0	0.0.0.0	0.0.0.0	0
so-0/0/1.0	PtToPt	0.0.0.0	0.0.0.0	0.0.0.0	1
so-0/0/3.0	PtToPt	0.0.0.0	0.0.0.0	0.0.0.0	1
so-0/0/0.0	PtToPt	0.0.0.1	0.0.0.0	0.0.0.0	1

The following sample output is for an OSPF configuration on R3, an ABR shown in Figure 11:

```

user@R3> show configuration

```

```

interfaces {
  so-0/0/0 {
    unit 0 {
      family inet {
        address 10.1.34.1/30;
      }
    }
  }
  so-0/0/1 {
    unit 0 {
      family inet {
        address 10.1.23.2/30;
      }
    }
  }
  so-0/0/3 {
    unit 0 {
      family inet {
        address 10.1.36.1/30;
      }
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 10.0.0.3/32;
      }
    }
  }
}

```



```

    }
}
routing-options {
    router-id 10.0.0.3;
}
protocols {
    ospf {
        area 0.0.0.0 {
            interface so-0/0/0.0;
            interface so-0/0/1.0;
            interface lo0.0 {
                passive;
            }
        }
        area 0.0.0.3 {
            interface so-0/0/3.0;
        }
    }
}

```

```
user@R3> show ospf interface
```

Interface	State	Area	DR ID	BDR ID	Nbrs
lo0.0	DRother	0.0.0.0	0.0.0.0	0.0.0.0	0
so-0/0/0.0	PtToPt	0.0.0.0	0.0.0.0	0.0.0.0	1
so-0/0/1.0	PtToPt	0.0.0.0	0.0.0.0	0.0.0.0	1
so-0/0/3.0	PtToPt	0.0.0.3	0.0.0.0	0.0.0.0	1

The following sample output is for an OSPF configuration on R4, an ABR shown in Figure 11:

```
user@R4> show configuration
```

```
[...Output truncated...]
```

```

interfaces {
    so-0/0/0 {
        unit 0 {
            family inet {
                address 10.1.34.2/30;
            }
        }
    }
    so-0/0/2 {
        unit 0 {
            family inet {
                address 10.1.45.1/30;
            }
        }
    }
    so-0/0/3 {
        unit 0 {
            family inet {
                address 10.1.24.2/30;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 10.0.0.4/32;
            }
        }
    }
}
routing-options {

```

```

    router-id 10.0.0.4;
  }
  protocols {
    ospf {
      area 0.0.0.0 {
        interface so-0/0/0.0;
        interface so-0/0/3.0;
        interface lo0.0 {
          passive;
        }
      }
      area 0.0.0.2 {
        stub default-metric 10;
        interface so-0/0/2.0;
      }
    }
  }
}

```

```

user@R4> show ospf interface

```

Interface	State	Area	DR ID	BDR ID	Nbrs
lo0.0	DRother	0.0.0.0	0.0.0.0	0.0.0.0	0
so-0/0/0.0	PtToPt	0.0.0.0	0.0.0.0	0.0.0.0	1
so-0/0/3.0	PtToPt	0.0.0.0	0.0.0.0	0.0.0.0	1
so-0/0/2.0	PtToPt	0.0.0.2	0.0.0.0	0.0.0.0	1

What It Means The sample output shows a basic OSPF configuration at the [edit protocols ospf] and [edit interfaces] hierarchy levels on the R2, R3, and R4 ABR routers.

R2 has four interfaces included at the [edit protocols ospf] hierarchy level, and those interfaces have the **family inet** statement included at the [edit interfaces] hierarchy level. Three interfaces—so-0/0/1.0, so-0/0/3.0, and the loopback (lo0) interface—are in the backbone (0.0.0.0). One interface, so-0/0/0.0, is in the NSSA (0.0.0.1). Because R2 has one interface configured for an NSSA, external routes learned from outside the AS (through R1) are redistributed throughout the network. For more information on OSPF routes, see “Examine OSPF Routes” on page 148.

R3 has four interfaces included at the [edit protocols ospf] hierarchy level, and those interfaces have the **family inet** statement included at the [edit interfaces] hierarchy level. Three interfaces—so-0/0/0.0, so-0/0/1.0, and the loopback (lo0) interface—are in the backbone (0.0.0.0). One interface, so-0/0/3.0, is in a non-backbone area (0.0.0.3).

R4 has four interfaces included at the [edit protocols ospf] hierarchy level, and those interfaces have the **family inet** statement included at the [edit interfaces] hierarchy level. Two interfaces, so-0/0/0.0 and so-0/0/3.0, are in the backbone (0.0.0.0). One interface, so-0/0/2.0, is in the stub area (0.0.0.2). Because internal routers within a stub area do not receive external LSA information, they must rely on either direct static routes or a default route to get to external destinations. A default route can be statically configured on the internal router or learned from the stub ABR. To advertise a default LSA from the stub ABR, include the **stub default-metric** statement at the [edit protocols ospf area area-id] hierarchy level to activate the default route.

All routers (R2, R3, and R4) have the router ID configured manually to avoid possible problems when the OSPF router ID (RID) changes; for example, when multiple loopback addresses are configured. The RID uniquely identifies the router within the OSPF network. It is transmitted within the LSAs used to populate the link-state database and calculate the shortest-path tree. In a link-state network, it is important that two routers do not share the same RID value, otherwise IP routing problems may occur.

An ABR belongs to more than one area and maintains a separate topological database for each area to which it is connected. For more information on the OSPF database, see “Examine the OSPF Link-State Database” on page 144.

See the *JUNOS Routing Protocols Configuration Guide* for more information on configuring OSPF on a router.

Step 3: Check OSPF on a Stub Router

Action To verify the OSPF configuration on a stub router in your network, enter the following commands:

```
user@host> show configuration
user@host> show ospf interface
```

Sample Output The following sample output is for an OSPF configuration on R5, a stub router shown in Figure 11:

```
user@R5> show configuration
[...Output truncated...]
interfaces {
  so-0/0/2 {
    unit 0 {
      family inet {
        address 10.1.45.2/30;
      }
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 10.0.0.5/32;
      }
    }
  }
}
routing-options {
  router-id 10.0.0.5;
}
protocols {
  ospf {
    area 0.0.0.2 {
      stub;
      interface so-0/0/2.0;
      interface lo0.0 {
        passive;
      }
    }
  }
}
```

```

user@R5> show ospf interface
Interface      State      Area      DR ID      BDR ID      Nbrs
lo0.0          DRother   0.0.0.2   0.0.0.0    0.0.0.0     0
so-0/0/2.0     PtToPt    0.0.0.2   0.0.0.0    0.0.0.0     1

```

What It Means The sample output shows a basic OSPF configuration at the [edit protocols ospf] and [edit interfaces] hierarchy levels on R5, a stub router.

R5 has two interfaces included at the [edit protocols ospf] hierarchy level, and those interfaces have the family inet statement included at the [edit interfaces] hierarchy level. Both interfaces, so-0/0/2.0 and the loopback interface (lo0), are in the stub area (0.0.0.2).

R5 has the router ID configured manually to avoid possible problems when the OSPF router ID (RID) changes; for example, when multiple loopback addresses are configured. The RID uniquely identifies the router within the OSPF network. It is transmitted within the LSAs used to populate the link-state database and calculate the shortest-path tree. In a link-state network, it is important that two routers do not share the same RID value, otherwise IP routing problems may occur.

A stub area does not allow AS external advertisements to flood within that area. R5 relies on a default route (0.0.0.0/0) to reach destinations outside the AS. The default route can be statically configured on R5 or advertised by an ABR (R4). In this network, the default LSA is advertised by R4.

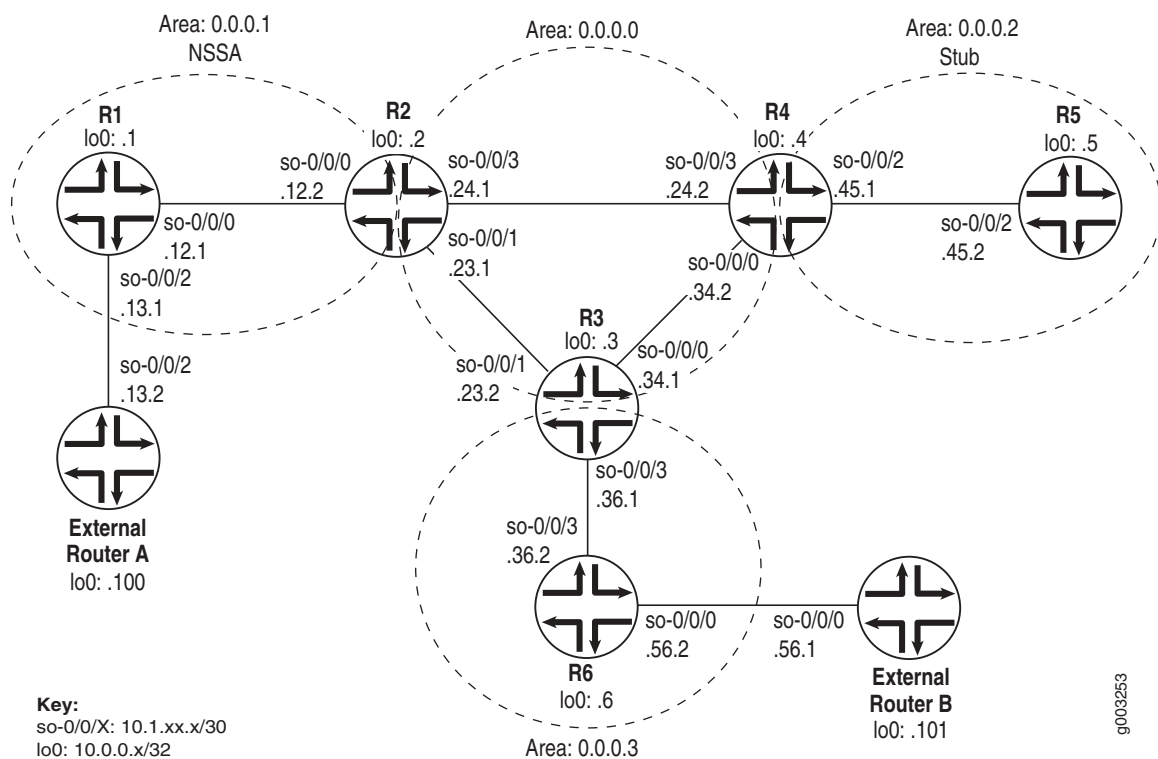
A stub area is useful if you want to reduce the size of the topological database and therefore the amount of memory required from the routers in the stub area. However, some restrictions apply to a stub area. You cannot create a virtual link through a stub area, and a stub area cannot contain an ASBR.

Check OSPF Neighbors

Purpose Assuming that all the routers are correctly configured for OSPF, you can verify which neighbors are adjacent and what type of LSAs are contained in the OSPF link-state database. In addition, you can examine the set of routes installed in the forwarding table to verify that the routing protocol process (rpd) has relayed the correct information into the forwarding table.

Figure 12 illustrates an example OSPF network topology used in this section.

Figure 12: OSPF Network Topology



The network consists of various types of routers that form adjacencies with neighboring OSPF routers. Once these adjacencies are in place, each router generates and floods LSAs into the network. The LSAs are placed into the link-state database on each router where the shortest path first (SPF) algorithm is calculated to find the best path to each router in the network. The network in Figure 12 should have the following adjacencies and LSA distribution:

- ABR routers R2, R3, and R4 should form adjacencies with routers in all areas to which they are connected (0.0.0.0, 0.0.0.1, 0.0.0.2, and 0.0.0.3). See “Check OSPF on an ABR” on page 135.
- Internal routers (R1, R5, and R6) should form adjacencies with routers inside their local area only. See “Check OSPF on a Stub Router” on page 139 and “Check OSPF on an ASBR” on page 132.
- Backbone area 0.0.0.0 should have Type 1, Type 3, Type 4, and Type 5 LSAs.

- NSSA area 0.0.0.1 should have Type 1, Type 3, and Type 7 LSAs.
- Stub area 0.0.0.2 should have Type 1 and Type 3 LSAs.
- Area 0.0.0.3 should have Type 1, Type 3, Type 4, and Type 5 LSAs.

Steps To Take To verify that routers are adjacent and have the correct exchange of LSAs, follow these steps:

1. Verify OSPF Neighbors on page 142
2. Examine the OSPF Link-State Database on page 144
3. Examine OSPF Routes on page 148
4. Examine the Forwarding Table on page 151
5. Examine Link-State Advertisements in Detail on page 152

Step 1: Verify OSPF Neighbors

Action To verify that routers are adjacent and able to exchange OSPF data, enter the following CLI operational mode command:

```
user@host> show ospf neighbor
```

Sample Output The following sample output shows the adjacencies that formed for all routers in Figure 12 on page 141:

```
user@R1> show ospf neighbor
  Address      Interface      State      ID           Pri  Dead
10.1.12.2      so-0/0/0.0     Full       10.0.0.2     128  36

user@R2> show ospf neighbor
  Address      Interface      State      ID           Pri  Dead
10.1.23.2      so-0/0/1.0     Full       10.0.0.3     128  32
10.1.24.2      so-0/0/3.0     Full       10.0.0.4     128  33
10.1.12.1      so-0/0/0.0     Full       10.0.0.1     128  33

user@R3> show ospf neighbor
  Address      Interface      State      ID           Pri  Dead
10.1.34.2      so-0/0/0.0     Full       10.0.0.4     128  36
10.1.23.1      so-0/0/1.0     Full       10.0.0.2     128  38
10.1.36.2      so-0/0/3.0     Full       10.0.0.6     128  33

user@R4> show ospf neighbor
  Address      Interface      State      ID           Pri  Dead
10.1.34.1      so-0/0/0.0     Full       10.0.0.3     128  31
10.1.24.1      so-0/0/3.0     Full       10.0.0.2     128  36
10.1.45.2      so-0/0/2.0     Full       10.0.0.5     128  39

user@R5> show ospf neighbor
  Address      Interface      State      ID           Pri  Dead
10.1.45.1      so-0/0/2.0     Full       10.0.0.4     128  35

user@R6> show ospf neighbor
  Address      Interface      State      ID           Pri  Dead
10.1.36.1      so-0/0/3.0     Full       10.0.0.3     128  31
```

What It Means The sample output shows that ABR routers R2, R3, and R4 have formed adjacencies with routers in all areas to which they are directly connected. Internal routers (R1, R5, and R6) have formed an adjacency with the other router inside their local area.

Adjacencies are formed after OSPF hello packets are sent and received by neighbors. Adjacencies determine the type of LSAs sent and received, and what topological database updates are sent. When adjacencies are established, pairs of adjacent routers synchronize their topological databases.

Table 33 lists and describes the fields in the `show ospf neighbor` command.

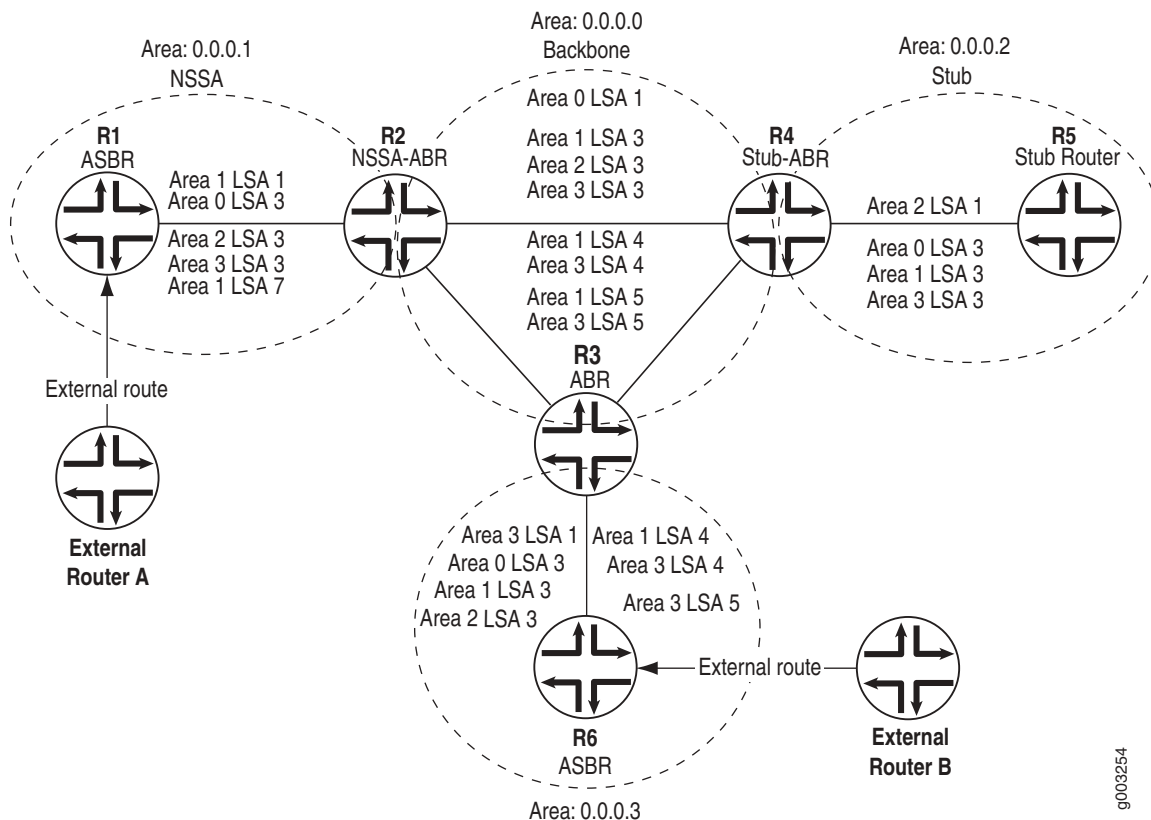
Table 33: Output Fields for the `show ospf neighbor` Command

Field	Description
Address	Address of the neighbor.
Interface	Interface through which the neighbor is reachable.
State	State of the neighbor. It can be Attempt, Down, Exchange, ExStart, Full, Init, Loading, or 2 Way.
ID	Router ID of the neighbor.
Pri	Priority of the neighbor to become the designated router. Only used on broadcast networks during designated router elections. By default, set to 128, indicating the highest priority and the most likely router to be elected designated router.
Dead	Number of seconds until the neighbor becomes unreachable.

Step 2: Examine the OSPF Link-State Database

Purpose You can determine if the correct types of LSAs are sent and received throughout the OSPF network by examining the entire OSPF link-state database. Figure 13 illustrates the flooding scope of LSAs generated and flooded in the example OSPF network.

Figure 13: LSA Flooding Scopes



This network should have the following distribution of LSAs:

- Backbone area 0.0.0.0 should have Type 1, Type 3, Type 4, and Type 5 LSAs.
- NSSA area 0.0.0.1 should have Type 1, Type 3, and Type 7 LSAs.
- Stub area 0.0.0.2 should have Type 1 and Type 3 LSAs.
- Area 0.0.0.3 should have Type 1, Type 3, Type 4, and Type 5 LSAs.

Because all routers in this network have SONET interfaces configured for Point-to-Point (PPP) encapsulation, all OSPF adjacencies are point-to-point, which results in Type 2 network LSAs not appearing in this network or being described in the following sections. Type 2 network LSAs are only advertised by a designated router, which is only present on broadcast or non-broadcast multiaccess (NBMA) networks.

Action To determine if the correct LSAs appear in the different areas of the OSPF AS, enter the following CLI operational mode command:

```
user@host> show ospf database
```

Sample Output user@R2> show ospf database

```

      OSPF link state database, area 0.0.0.0
Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router    *10.0.0.2      10.0.0.2    0x80000049 1555 0x2  0xd72a 84
Router    10.0.0.3      10.0.0.3    0x80000038 1395 0x2  0xef0e 84
Router    10.0.0.4      10.0.0.4    0x80000041 914  0x2  0x46a9 84
Summary   *10.0.0.1      10.0.0.2    0x80000047 1855 0x2  0xf509 28
Summary   10.0.0.5      10.0.0.4    0x8000003c 2114 0x2  0xd72c 28
Summary   10.0.0.6      10.0.0.3    0x80000033 1995 0x2  0xe527 28
Summary   *10.1.12.0     10.0.0.2    0x80000047 786  0x2  0x5d98 28
Summary   10.1.36.0     10.0.0.3    0x80000035 2426 0x2  0x727c 28
Summary   10.1.45.0     10.0.0.4    0x8000003d 1021 0x2  0xf8e3 28
ASBRSum   *10.0.0.1      10.0.0.2    0x80000046 355  0x2  0xe915 28
ASBRSum   10.0.0.6      10.0.0.3    0x80000032 1526 0x2  0xd933 28

```

```

      OSPF link state database, area 0.0.0.1
Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router    10.0.0.1      10.0.0.1    0x80000058 858  0x0  0x5c26 60
Router    *10.0.0.2      10.0.0.2    0x80000048 1986 0x0  0xecbd 48
Summary   *10.0.0.2      10.0.0.2    0x80000039 1686 0x0  0x1cf2 28
Summary   *10.0.0.3      10.0.0.2    0x80000038 2286 0x0  0x1eef 28
Summary   *10.0.0.4      10.0.0.2    0x80000038 955  0x0  0x14f8 28
Summary   *10.0.0.5      10.0.0.2    0x80000038 186  0x0  0x14f6 28
Summary   *10.0.0.6      10.0.0.2    0x80000038 2155 0x0  0xaaff 28
Summary   *10.1.23.0     10.0.0.2    0x80000046 655  0x0  0x4e9 28
Summary   *10.1.24.0     10.0.0.2    0x80000046 486  0x0  0xf8f3 28
Summary   *10.1.34.0     10.0.0.2    0x80000039 1255 0x0  0xae40 28
Summary   *10.1.36.0     10.0.0.2    0x80000039 55  0x0  0x9854 28
Summary   *10.1.45.0     10.0.0.2    0x80000039 1086 0x0  0x35ae 28
NSSA      *0.0.0.0      10.0.0.2    0x80000044 2455 0x0  0xd821 36
NSSA      10.0.0.100    10.0.0.1    0x80000051 2916 0x8  0x797c 36

```

```

      OSPF AS SCOPE link state database
Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Extern    *10.0.0.100    10.0.0.2    0x8000005e 1386 0x2  0xcf20 36
Extern    10.0.0.101    10.0.0.6    0x8000002b 333  0x2  0x9791 36

```

```
user@R3> show ospf database
```

```

      OSPF link state database, area 0.0.0.0
Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router    10.0.0.2      10.0.0.2    0x80000049 1668 0x2  0xd72a 84
Router    *10.0.0.3      10.0.0.3    0x80000038 1506 0x2  0xef0e 84
Router    10.0.0.4      10.0.0.4    0x80000041 1027 0x2  0x46a9 84
Summary   10.0.0.1      10.0.0.2    0x80000047 1968 0x2  0xf509 28
Summary   10.0.0.5      10.0.0.4    0x8000003c 2227 0x2  0xd72c 28
Summary   *10.0.0.6      10.0.0.3    0x80000033 2106 0x2  0xe527 28
Summary   10.1.12.0     10.0.0.2    0x80000047 900  0x2  0x5d98 28
Summary   *10.1.36.0     10.0.0.3    0x80000036 6  0x2  0x707d 28
Summary   10.1.45.0     10.0.0.4    0x8000003d 1134 0x2  0xf8e3 28
ASBRSum   10.0.0.1      10.0.0.2    0x80000046 468  0x2  0xe915 28
ASBRSum   *10.0.0.6      10.0.0.3    0x80000032 1638 0x2  0xd933 28

```

```

      OSPF link state database, area 0.0.0.3
Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router    *10.0.0.3      10.0.0.3    0x80000036 2406 0x2  0x3452 48
Router    10.0.0.6      10.0.0.6    0x8000002f 445  0x2  0x1850 60

```

```

Summary *10.0.0.1      10.0.0.3      0x80000036   906 0x2 0x1cf1 28
Summary *10.0.0.2      10.0.0.3      0x80000036   738 0x2 0x806 28
Summary *10.0.0.3      10.0.0.3      0x80000033  1806 0x2 0xf917 28
Summary *10.0.0.4      10.0.0.3      0x80000033  1038 0x2 0xf915 28
Summary *10.0.0.5      10.0.0.3      0x80000033   306 0x2 0xf913 28
Summary *10.1.12.0     10.0.0.3      0x80000036   606 0x2 0x8381 28
Summary *10.1.23.0     10.0.0.3      0x80000036   438 0x2 0xffffa 28
Summary *10.1.24.0     10.0.0.3      0x80000036  1338 0x2 0xfef9 28
Summary *10.1.34.0     10.0.0.3      0x80000036   138 0x2 0x8669 28
Summary *10.1.45.0     10.0.0.3      0x80000033  1206 0x2 0x1dc9 28
ASBRSum *10.0.0.1      10.0.0.3      0x80000035  2238 0x2 0x10fd 28
ASBRSum *10.0.0.2      10.0.0.3      0x80000035  1938 0x2 0xfb12 28

```

OSPF AS SCOPE link state database

Type	ID	Adv Rtr	Seq	Age	Opt	Cksum	Len
Extern	10.0.0.100	10.0.0.2	0x8000005e	1500	0x2	0xcf20	36
Extern	10.0.0.101	10.0.0.6	0x8000002b	445	0x2	0x9791	36

```
user@R4> show ospf database
```

OSPF link state database, area 0.0.0.0

Type	ID	Adv Rtr	Seq	Age	Opt	Cksum	Len
Router	10.0.0.2	10.0.0.2	0x80000049	1711	0x2	0xd72a	84
Router	10.0.0.3	10.0.0.3	0x80000038	1550	0x2	0xef0e	84
Router	*10.0.0.4	10.0.0.4	0x80000041	1068	0x2	0x46a9	84
Summary	10.0.0.1	10.0.0.2	0x80000047	2011	0x2	0xf509	28
Summary	*10.0.0.5	10.0.0.4	0x8000003c	2268	0x2	0xd72c	28
Summary	10.0.0.6	10.0.0.3	0x80000033	2150	0x2	0xe527	28
Summary	10.1.12.0	10.0.0.2	0x80000047	942	0x2	0x5d98	28
Summary	10.1.36.0	10.0.0.3	0x80000036	50	0x2	0x707d	28
Summary	*10.1.45.0	10.0.0.4	0x8000003d	1175	0x2	0xf8e3	28
ASBRSum	10.0.0.1	10.0.0.2	0x80000046	511	0x2	0xe915	28
ASBRSum	10.0.0.6	10.0.0.3	0x80000032	1681	0x2	0xd933	28

OSPF link state database, area 0.0.0.2

Type	ID	Adv Rtr	Seq	Age	Opt	Cksum	Len
Router	*10.0.0.4	10.0.0.4	0x8000003f	875	0x0	0x5913	48
Router	10.0.0.5	10.0.0.5	0x8000002e	1263	0x0	0x5a03	60
Summary	*0.0.0.0	10.0.0.4	0x80000019	768	0x0	0x4be3	28
Summary	*10.0.0.1	10.0.0.4	0x80000040	575	0x0	0x20e4	28
Summary	*10.0.0.2	10.0.0.4	0x80000040	468	0x0	0xcf8	28
Summary	*10.0.0.3	10.0.0.4	0x8000003f	275	0x0	0x401	28
Summary	*10.0.0.4	10.0.0.4	0x8000003d	168	0x0	0xf313	28
Summary	*10.0.0.6	10.0.0.4	0x8000003d	2075	0x0	0xf30f	28
Summary	*10.1.12.0	10.0.0.4	0x8000003f	1968	0x0	0x8973	28
Summary	*10.1.23.0	10.0.0.4	0x8000003f	1775	0x0	0x10e1	28
Summary	*10.1.24.0	10.0.0.4	0x8000003d	1668	0x0	0xfef4	28
Summary	*10.1.34.0	10.0.0.4	0x8000003d	1475	0x0	0x9059	28
Summary	*10.1.36.0	10.0.0.4	0x8000003d	1368	0x0	0x8462	28

OSPF AS SCOPE link state database

Type	ID	Adv Rtr	Seq	Age	Opt	Cksum	Len
Extern	10.0.0.100	10.0.0.2	0x8000005e	1542	0x2	0xcf20	36
Extern	10.0.0.101	10.0.0.6	0x8000002b	488	0x2	0x9791	36

What It Means The sample output shows that all the ABRs have the correct distribution of LSAs. Area 0.0.0.0 for all routers has Type 1 router, Type 3 summary, and Type 4 ASBR summary LSAs. Each ABR has an OSPF AS scope link-state database that includes Type 5 external LSAs.

Note that Type 2 network LSAs are not found in this topology because both broadcast or NMBA network types are not present.

NSSA area 0.0.0.1, in the output for R2, has Type 1 router, Type 3 summary, and Type 7 NSSA LSAs. Stub area 0.0.0.2, in the output for R4, has Type 1 router and Type 3 summary LSAs. Non-backbone area 0.0.0.3, in the output for R3, has Type 1 router, Type 3 summary, Type 4 ASBR, and Type 5 external LSAs.

All areas have a Type 1 router LSA because the Type 1 LSA is generated for each router that has interfaces in that area. Because this LSA has an area flooding scope, it remains within its own particular area and is not seen in other areas. For example, in the link-state database for area 0.0.0.2, there are two router LSAs: one for R4 and one for R5.

The ABR for that area places the routing information contained within the Type 1 LSA into a Type 3 summary or Type 4 ASBR summary LSA and forwards it across the area boundary. Whether the area receives a Type 3 or Type 4 summary LSA depends on whether the area is a stub area. Type 3 summary LSAs appear in all areas, but Type 4 LSAs only appear in non-stub areas as indicated in the link-state databases for areas 0.0.0.1, 0.0.0.2, and 0.0.0.3.

Each ABR router has a Type 5 AS external LSA used to advertise any networks external to the OSPF AS. This LSA is flooded by the ABRs to each non-stub router in the entire AS. For example, within area 0.0.0.0, Type 5 LSAs exist for areas 0.0.0.1 and 0.0.0.3. Both of these areas are connected to routers (external router A and external router B) from other ASs, which results in the injection of external routes into the OSPF AS. However, there are no Type 5 LSAs in stub areas 0.0.0.1 and 0.0.0.2.

A Type 7 NSSA external LSA appears in NSSA area 0.0.0.1 and is used within the NSSA to advertise an external router. This LSA is flooded to each router in the NSSA and is not sent to other adjacent areas. For example, only area 0.0.0.1 has a Type 7 LSA. Because a Type 7 LSA does not traverse area boundaries, the ABR in the NSSA (R2) translates the Type 7 LSA into a Type 5 LSA that is forwarded to all areas (with the exception of stub areas).

The sample output shows that each router has two databases, indicating that it is an ABR between the backbone and a non-backbone, stub, or NSSA area. All of the addresses preceded by an asterisk (*) are LSAs that originated with the router from which the output was taken.

Step 3: Examine OSPF Routes

Purpose You can determine if the LSAs that appear in the link-state database of a router are correct by examining the route to the destination. In this step, three routes are examined. The first example shows the route from R5 to external router A, the second shows the route from R6 to external router A, and the third shows the route from R4 to R6.

Action To examine a route in an OSPF AS, enter one or all of the following CLI commands:

```
user@host> show route destination-prefix
user@host> show ospf database
```

Sample Output 1 The following sample output shows the path from R5 to external router A:

```
user@R5> show route 10.0.0.100

inet.0: 23 destinations, 25 routes (23 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[OSPF/10] 01:58:42, metric 11
                   > via so-0/0/2.0

user@R5> show ospf database

        OSPF link state database, area 0.0.0.2
Type      ID          Adv Rtr          Seq          Age  Opt  Cksum  Len
Router    10.0.0.4          10.0.0.4          0x8000002b    140  0x0  0x81fe  48
Router    *10.0.0.5          10.0.0.5          0x8000001f    526  0x0  0x78f3  60
Summary   0.0.0.0            10.0.0.4          0x80000005     32  0x0  0x73cf  28
Summary   10.0.0.1            10.0.0.4          0x8000002b   2132  0x0  0x4acf  28
Summary   10.0.0.2            10.0.0.4          0x8000002b   1940  0x0  0x36e3  28
Summary   10.0.0.3            10.0.0.4          0x8000002a   1832  0x0  0x2eeb  28
Summary   10.0.0.4            10.0.0.4          0x80000028   1640  0x0  0x1efd  28
Summary   10.0.0.6            10.0.0.4          0x80000029   1340  0x0  0x1cfa  28
Summary   10.1.12.0           10.0.0.4          0x8000002b   1232  0x0  0xb15f  28
Summary   10.1.23.0           10.0.0.4          0x8000002b   1040  0x0  0x38cd  28
Summary   10.1.24.0           10.0.0.4          0x80000029    932  0x0  0x27e0  28
Summary   10.1.34.0           10.0.0.4          0x80000029    740  0x0  0xb845  28
Summary   10.1.36.0           10.0.0.4          0x80000029    632  0x0  0xac4e  28
```

Sample Output 2 The following sample output shows the route from R6 to external router A:

```
user@R6> show route 10.0.0.100

inet.0: 29 destinations, 31 routes (29 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.100/32      *[OSPF/150] 16:52:11, metric 0, tag 0
                   > via so-0/0/3.0
```

```
user@R6> show ospf database
```

```

      OSPF link state database, area 0.0.0.3
  Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router  10.0.0.3        10.0.0.3    0x8000001d  502  0x2  0x6639  48
Router  *10.0.0.6        10.0.0.6    0x80000019  807  0x2  0x443a  60
Summary 10.0.0.1          10.0.0.3    0x8000001c  1570 0x2  0x50d7  28
Summary 10.0.0.2          10.0.0.3    0x8000001c  1402 0x2  0x3ceb  28
Summary 10.0.0.3          10.0.0.3    0x80000019  2470 0x2  0x2efc  28
Summary 10.0.0.4          10.0.0.3    0x80000019  1702 0x2  0x2efa  28
Summary 10.0.0.5          10.0.0.3    0x80000019  970  0x2  0x2ef8  28
Summary 10.1.12.0         10.0.0.3    0x8000001c  1270 0x2  0xb767  28
Summary 10.1.23.0         10.0.0.3    0x8000001c  1102 0x2  0x34e0  28
Summary 10.1.24.0         10.0.0.3    0x8000001c  2002 0x2  0x33df  28
Summary 10.1.34.0         10.0.0.3    0x8000001c  802  0x2  0xba4f  28
Summary 10.1.45.0         10.0.0.3    0x80000019  1870 0x2  0x51af  28
ASBRSum 10.0.0.1          10.0.0.3    0x8000001c  370  0x2  0x42e4  28
ASBRSum 10.0.0.2          10.0.0.3    0x8000001c  70  0x2  0x2ef8  28

      OSPF AS SCOPE link state database
  Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Extern  10.0.0.100        10.0.0.2    0x80000042  384  0x2  0x804  36
Extern  *10.0.0.101        10.0.0.6    0x80000015  807  0x2  0xc37b  36
Extern  10.1.13.0          10.0.0.2    0x80000041  234  0x2  0x481e  36
Extern  10.1.15.0          10.0.0.2    0x80000041  233  0x2  0x3232  36
Extern  100.168.64.0       10.0.0.2    0x80000041  82  0x2  0xe0f7  36

```

Sample Output 3 The following sample output shows the route from R4 to R6:

```
user@R4> show route 10.0.0.6
```

```

inet.0: 27 destinations, 31 routes (27 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

```

```

10.0.0.6/32          *[OSPF/10] 17:02:28, metric 2
                    > via so-0/0/0.0

```

```
user@R4> show ospf database
```

```

      OSPF link state database, area 0.0.0.0
  Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router  10.0.0.2          10.0.0.2    0x8000002f  632  0x2  0xc10  84
Router  10.0.0.3          10.0.0.3    0x8000001e  2271 0x2  0x24f3  84
Router  *10.0.0.4          10.0.0.4    0x80000022  1582 0x2  0x848a  84
Summary 10.0.0.1          10.0.0.2    0x8000002d  789  0x2  0x2aee  28
Summary *10.0.0.5          10.0.0.4    0x8000001e  982  0x2  0x140e  28
Summary 10.0.0.6          10.0.0.3    0x8000001a  302  0x2  0x180e  28
Summary 10.1.12.0         10.0.0.2    0x8000002c  1847 0x2  0x937d  28
Summary 10.1.36.0         10.0.0.3    0x8000001c  771  0x2  0xa463  28
Summary *10.1.45.0         10.0.0.4    0x8000001f  1789 0x2  0x35c5  28
ASBRSum 10.0.0.1          10.0.0.2    0x8000002b  1533 0x2  0x20f9  28
ASBRSum 10.0.0.6          10.0.0.3    0x80000018  2402 0x2  0xe19  28

```

```

      OSPF link state database, area 0.0.0.2
Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router    *10.0.0.4    10.0.0.4    0x80000020 1282 0x0  0x97f3  48
Router    10.0.0.5     10.0.0.5    0x80000018 1685 0x0  0x86ec  60
Summary   *10.0.0.1     10.0.0.4    0x80000021 1189 0x0  0x5ec5  28
Summary   *10.0.0.2     10.0.0.4    0x80000021  889 0x0  0x4ad9  28
Summary   *10.0.0.3     10.0.0.4    0x80000020  682 0x0  0x42e1  28
Summary   *10.0.0.4     10.0.0.4    0x8000001e 1489 0x0  0x32f3  28
Summary *10.0.0.6     10.0.0.4    0x8000001f  589 0x0  0x30f0  28
Summary   *10.1.12.0    10.0.0.4    0x80000021  382 0x0  0xc555  28
Summary   *10.1.23.0    10.0.0.4    0x80000021  289 0x0  0x4cc3  28
Summary   *10.1.24.0    10.0.0.4    0x80000020   82 0x0  0x39d7  28
Summary   *10.1.34.0    10.0.0.4    0x8000001f 2089 0x0  0xcc3b  28
Summary   *10.1.36.0    10.0.0.4    0x8000001f 1882 0x0  0xc044  28

      OSPF AS SCOPE link state database
Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Extern    10.0.0.100    10.0.0.2    0x80000042  484 0x2  0x804   36
Extern    10.0.0.101    10.0.0.6    0x80000015  910 0x2  0xc37b  36
Extern    10.1.13.0     10.0.0.2    0x80000041  333 0x2  0x481e  36
Extern    10.1.15.0     10.0.0.2    0x80000041  332 0x2  0x3232  36
Extern    100.168.64.0  10.0.0.2    0x80000041  182 0x2  0xe0f7  36

```

What It Means Sample output 1 shows an OSPF default route (0.0.0.0/0) with a preference value of 10. In the area 0.0.0.2 link-state database, a Type 3 summary LSA advertises the default route.

Sample output 2 shows an OSPF route with a preference value of 150. In the AS scope link-state database, an external Type 5 LSA indicates that the route from R6 to external router A is through R2, the advertising router. By default, routes resulting from OSPF external LSAs are installed with a preference value of 150.

Sample output 3 shows an OSPF route with a preference value of 10. In the area 0.0.0.0 link-state database, a summary Type 3 LSA indicates that the route from R4 to R6 is through R3, the advertising router.

The LSAs placed into the link-state database are used by the router to run the Dijkstra algorithm (also called the shortest path first algorithm). This computation uses the link-state database as a source, resulting in a loop-free topology using the best metric from the local router to all nodes in the OSPF network.

Step 4: Examine the Forwarding Table

Purpose You can examine the set of routes installed in the forwarding table to verify that the routing protocol process (rpd) has relayed the correct information into the forwarding table.

Action To examine the forwarding table, enter the following CLI commands:

```
user@host> show route destination-prefix extensive
user@host> show route forwarding-table destination destination-prefix
```

Sample Output user@R2> show route 10.0.0.3 extensive

```
inet.0: 19 destinations, 24 routes (19 active, 0 holddown, 0 hidden)
10.0.0.3/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 10.0.0.3/32 -> {so-0/0/1.0}
    *OSPF   Preference: 10
           Next hop: via so-0/0/1.0, selected
           State: <Active Int>
           Local AS: 65002
           Age: 5d 16:07:37           Metric: 1
           Area: 0.0.0.0
           Task: OSPF
           Announcement bits (2): 0-KRT 4-Resolve inet.0
           AS path: I
```

```
user@R2> show route forwarding-table destination 10.0.0.3
Routing table: inet
Internet:
Destination      Type RtRef Next hop           Type Index NhRef Netif
10.0.0.3/32      user   1          so-0/0/1.0         ucst  294   3 so-0/0/1.0
```

What It Means The sample output shows that the same next hop appears in the output for the `show route destination-prefix extensive` and the `show route forwarding-table destination destination-prefix` commands, indicating that the routing protocol process (rpd) is relaying the correct next hop to the forwarding table.

The `show route destination-prefix extensive` command displays very detailed route information about the active entries for the specified address or range of addresses.

The `show route forwarding-table destination destination-prefix` command displays the route entries in the kernel's forwarding table. This is the version of the forwarding table in the Routing Engine. The Routing Engine copies this table to the Packet Forwarding Engine.



NOTE: The `show route forwarding-table` command is an independent command, not a filter that selects specific information that is displayed from the routing tables. You cannot use this command in conjunction with any of the `show route` filter options.

For more information about the `show route` command, see the *JUNOS Routing Protocols and Policies Command Reference*.

Step 5: Examine Link-State Advertisements in Detail

Purpose You can obtain important information about the routers in your network by examining LSAs in detail.

Steps To Take To examine OSPF LSAs, follow these steps:

1. Examine a Type 1 Router LSA on page 152
2. Examine a Type 3 Summary LSA on page 153
3. Examine a Type 4 ASBR Summary LSA on page 154
4. Examine a Type 5 AS External LSA on page 155
5. Examine Type 7 NSSA External LSA on page 156

Examine a Type 1 Router LSA

Action To examine a Type 1 router LSA, enter the following CLI operational mode command:

```
user@host> show ospf database router extensive
```

Sample Output user@R1> show ospf database router extensive

```

      OSPF link state database, area 0.0.0.1
Type      ID          Adv Rtr          Seq          Age  Opt  Cksum  Len
Router  *10.0.0.1      10.0.0.1        0x8000005a    1180 0x0  0x5828  60
  bits 0x2, link count 3
  id 10.0.0.1, data 255.255.255.255, Type Stub (3)
  TOS count 0, TOS 0 metric 0
  id 10.0.0.2, data 10.1.12.1, Type PointToPoint (1)
  TOS count 0, TOS 0 metric 1
  id 10.1.12.0, data 255.255.255.252, Type Stub (3)
  TOS count 0, TOS 0 metric 1
  Gen timer 00:30:19
  Aging timer 00:40:19
  Installed 00:19:40 ago, expires in 00:40:20, sent 00:19:38 ago
  Ours
Router   10.0.0.2      10.0.0.2        0x8000004b    679 0x0  0xe6c0  48
  bits 0x3, link count 2
  id 10.0.0.1, data 10.1.12.2, Type PointToPoint (1)
  TOS count 0, TOS 0 metric 1
  id 10.1.12.0, data 255.255.255.252, Type Stub (3)
  TOS count 0, TOS 0 metric 1
  Aging timer 00:48:40
  Installed 00:11:16 ago, expires in 00:48:41, sent 3w0d 23:33:12 ago

```

What It Means The sample output shows the details of two router LSAs: the first for R1 (*10.0.0.1) and the second for R2 (10.0.0.2). The asterisk (*) indicates that the LSA was generated by R1. You can also determine ownership of the LSA by the last line of the output in this case, *ours*.

Each time the LSA is updated, the sequence (*seq*) field increments, indicating that the router has the most recent version of the LSA. Values range from 0x80000001 to 0xFFFFFFFF. If the sequence field is not incrementing, there may be problems with the connection.

The **bits** field is set to 0x2 in the first LSA and 0x3 in the second LSA. When the **bits** field is set to 0x2, the originating router (R1) is an ASBR. When the **bits** field is set to 0x3, the originating router (R2) is both ABR and ASBR.

R1 has three links connected to area 0.0.0.1 shown by the link count field that is set to a value of 3. The **Type** field shows that R1 has a single point-to-point link to R2 and two links advertised as stub networks.

Each OSPF router generates a single Type 1 LSA to describe the status and cost (metric) of all links on the router. This LSA is flooded to each router in the OSPF area. It is defined as having an area scope, so it is not flooded across an area boundary.

Examine a Type 3 Summary LSA

Action To examine a Type 3 summary LSA, enter the following CLI operational mode command:

```
user@host> show ospf database netsummary extensive
```

Sample Output user@R2> show ospf database netsummary extensive

```

      OSPF link state database, area 0.0.0.0
Type      ID          Adv Rtr      Seq          Age  Opt  Cksum  Len
Summary *10.0.0.1      10.0.0.2      0x80000043   529  0x2  0xfd05  28
mask 255.255.255.255
TOS 0x0, metric 1
Gen timer 00:34:13
Aging timer 00:51:10
Installed 00:08:49 ago, expires in 00:51:11, sent 00:08:47 ago
Ours,
[...Output truncated...]
      OSPF link state database, area 0.0.0.1
[...Output truncated...]
Summary *10.0.0.5      10.0.0.2      0x80000047   2198  0x0  0xf506  28
mask 255.255.255.255
TOS 0x0, metric 2
Gen timer 00:07:19
Aging timer 00:23:22
Installed 00:36:38 ago, expires in 00:23:22, sent 00:36:36 ago
Ours,

```

What It Means The sample output shows that R2 is an ABR because it contains two databases: one for the backbone area 0.0.0.0 and one for area 0.0.0.1. Within the backbone area, the summary LSA *10.0.0.1 is generated from R2 as indicated by the asterisk (*) next to the link-state ID field, and **ours** in the last line of the LSA. The cost to transmit data out of the interface is 1, as indicated by the **metric** field.

Within area 0.0.0.1, the summary LSA *10.0.0.5 is generated by R2 and has a metric of 2, which is the cost to R5 from R2. Before calculating the SPF algorithm, the local router (R2) must add an additional metric of 1 to the existing metric of 1. The additional metric of 1 must be added because there is another router between R2 and R5 (R4).

Each time the LSA is updated, the sequence (**seq**) field increments, indicating that the router has the most recent version of the LSA. Values range from **0x80000001** to **0x7FFFFFFF**. If the sequence field is not incrementing, there may be problems with the connection.

Examine a Type 4 ASBR Summary LSA

Action To examine a Type 4 ASBR summary LSA, enter the following CLI operational mode command:

```
user@host> show ospf database asbrsummary extensive
```

Sample Output user@R3> show ospf database asbrsummary extensive

```

      OSPF link state database, area 0.0.0.0
[...Output truncated...]
ASBRSum *10.0.0.6          10.0.0.3          0x80000042  1023  0x2  0xb943  28
  mask 0.0.0.0
  TOS 0x0, metric 1
  Gen timer 00:27:57
  Aging timer 00:42:57
  Installed 00:17:03 ago, expires in 00:42:57, sent 00:17:01 ago
  Ours,
[...Output truncated...]

```

What It Means The sample output shows that an LSA within the backbone area, ***10.0.0.6**, is generated by ASBR R3, as indicated by the asterisk (*) next to the link-state ID field and **ours** in the last line of the LSA.

Each time the LSA is updated, the sequence (**seq**) field increments, indicating that the router has the most recent version of the LSA. Values range from **0x80000001** to **0x7FFFFFFF**. If the sequence field is not incrementing, there may be problems with the connection.

Because the router ID of all the ASBR summary LSAs is a full 32-bit value, the network mask is not needed and is set to a value of **0.0.0.0**. The metric for the LSA within the backbone area is set to 1, which is the cost to the advertising router (R3) from the originating router (R6). The metric is calculated before the SPF algorithm is calculated.

In general, each ABR that must transmit information about an ASBR from one OSPF area into another generates a Type 4 LSA. This LSA is flooded to each router in the OSPF area. A Type 4 LSA is defined as having an area scope so that another ABR does not reflood it across the area boundary.

Examine a Type 5 AS External LSA

Action To examine a Type 5 AS external LSA, enter the following CLI operational mode command:

```
user@host> show ospf database extern extensive
```

Sample Output

```
user@R2> show ospf database extern extensive
      OSPF AS SCOPE link state database
      Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Extern *10.0.0.100      10.0.0.2    0x80000047  1377  0x2  0xfd09  36
      mask 255.255.255.255
      Type 2, TOS 0x0, metric 0, fwd addr 10.0.0.1, tag 0.0.0.0
      Gen timer 00:21:02
      Aging timer 00:37:02
      Installed 00:22:57 ago, expires in 00:37:03, sent 00:22:55 ago
      Ours,
      [...Output truncated...]
```

What It Means The sample output shows one Type 5 external LSA, *10.0.0.100. The status of the router represented by this LSA is indicated by the **fwd addr** field, which shows that it does not belong to any particular OSPF area. The forwarding address provides the address toward which packets should be sent to reach the external router (10.0.0.1). R1 is the ASBR with the connection to external router A.

The **mask** field represents the subnet mask associated with the advertised router. It is used with the link-state ID field (10.0.0.100), which encapsulates the network address in a Type 5 LSA. This LSA has a metric value of 0, the default value, indicating that this is a Type 2 external metric. Thus, any local router should use the default metric (0) when performing an SPF algorithm.

Each time the LSA is updated, the sequence (**seq**) field increments, indicating that the router has the most recent version of the LSA. Values range from 0x80000001 to 0xFFFFFFFF. If the sequence field is not incrementing, there may be problems with the connection.

In general, each ASBR generates a Type 5 LSA to advertise any routers external to the OSPF AS. This LSA is flooded to each non-stub router in the entire AS.

Examine Type 7 NSSA External LSA

Action To examine a Type 7 NSSA external LSA, enter the following CLI operational mode command:

```
user@host> show ospf database nssa extensive
```

Sample Output user@R1> show ospf database nssa extensive

```

      OSPF link state database, area 0.0.0.1
Type      ID              Adv Rtr          Seq          Age  Opt  Cksum  Len
[...Output truncated...]
NSSA      *10.0.0.100      10.0.0.1        0x8000003b    843  0x8  0xa566  36
mask 255.255.255.255
Type 2, TOS 0x0, metric 0, fwd addr 10.0.0.1, tag 0.0.0.0
Gen timer 00:35:56
Aging timer 00:45:56
Installed 00:14:03 ago, expires in 00:45:57, sent 00:14:01 ago
Ours

```

What It Means The sample output shows that the LSA belongs to a single NSSA, 0.0.0.1, and was generated by R1. This router has a metric value of 0, which is the default, and is listed as a Type 2 external metric. Any local router must use the default metric as the total cost for the route when performing an SPF calculation. The default metric of the route must be added to the cost to reach the advertising ASBR. This value then represents the total cost for the route.

In general, each ASBR within the NSSA generates a Type 7 LSA to advertise any routers external to the OSPF AS. This LSA is flooded to each router within the NSSA (R2). Because the LSA has only an area flooding scope, it is not sent into other adjacent areas. For each Type 7 LSA received, the ABR (R2) translates the information into a Type 5 LSA and sends the information into the backbone. The other backbone routers do not know that the original information came from an NSSA. The Type 5 LSA is then flooded to each non-stub router in the entire AS.

Chapter 13

Verify the BGP Protocol and Peers

This chapter describes how to check whether the Border Gateway Protocol (BGP) is configured correctly on a Juniper Networks router in your network, the internal Border Gateway Protocol (IBGP) and exterior Border Gateway Protocol (EBGP) sessions are properly established, the external routes are advertised and received correctly, and the BGP path selection process is working properly. (See Table 34.)

Table 34: Checklist for Verifying the BGP Protocol and Peers

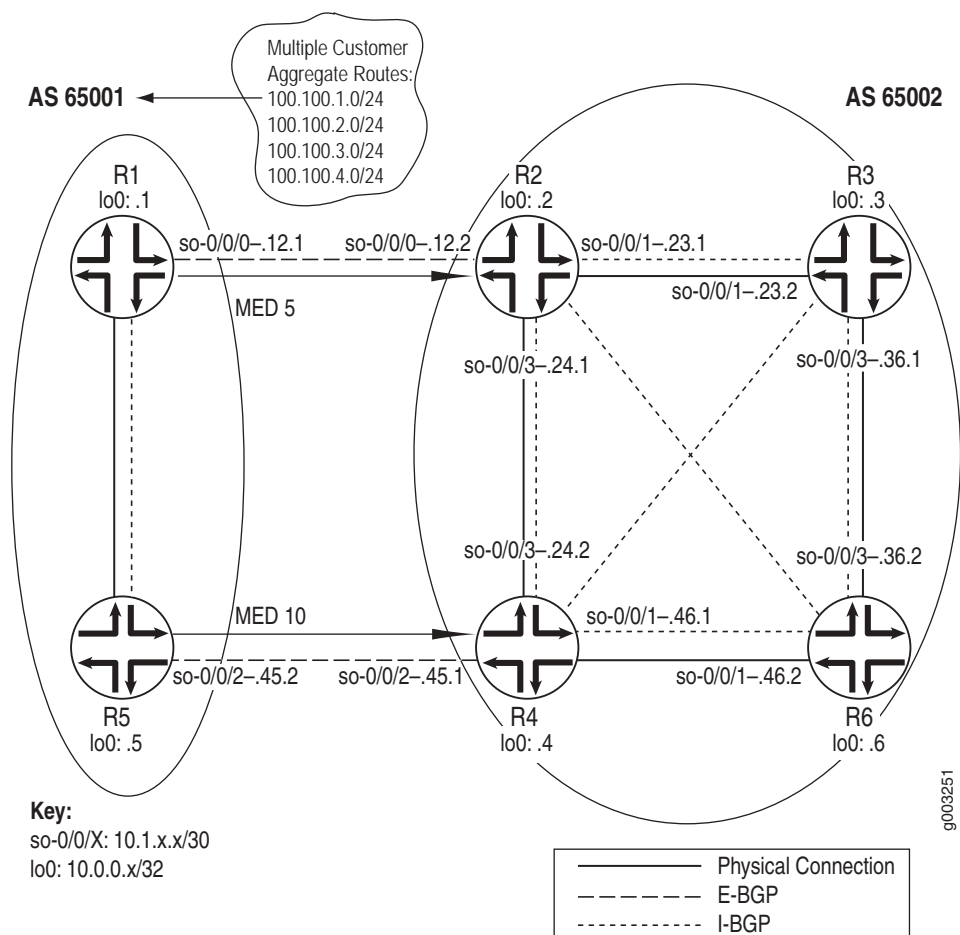
Verify the BGP Protocol and Peers Tasks	Command or Action
Verify the BGP Protocol on page 158	
1. Verify BGP on an Internal Router on page 159	show configuration
2. Verify BGP on a Border Router on page 162	show configuration
Verify BGP Peers on page 165	
1. Check That BGP Sessions Are Up on page 166	show bgp summary
2. Verify That a Neighbor is Advertising a Particular Route on page 169	show route advertising-protocol bgp <i>neighbor-address</i>
3. Verify That a Particular BGP Route Is Received on Your Router on page 170	show route receive-protocol bgp <i>neighbor-address</i>
Examine BGP Routes and Route Selection on page 171	
1. Examine the Local Preference Selection on page 173	show route <i>destination-prefix</i> < detail >
2. Examine the Multiple Exit Discriminator Route Selection on page 174	show route <i>destination-prefix</i> < detail >
3. Examine the EBGP over IBGP Selection on page 175	show route <i>destination-prefix</i> < detail >
4. Examine the IGP Cost Selection on page 176	show route <i>destination-prefix</i> < detail >
Examine Routes in the Forwarding Table on page 177	show route forwarding-table

Verify the BGP Protocol

Purpose For BGP to run on a router in your network, you must define the local autonomous system (AS) number, configure at least one group, and include information about at least one peer in the group. If the peer is an EBGP peer, include the peer's AS number. For all peers, include either the peer's interface IP address or loopback (lo0) IP address. When configuring BGP on an interface, you must also include the family inet statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level.

Figure 14 illustrates the example configurations used in this section.

Figure 14: BGP Configuration Topology



The network in Figure 14 consists of two directly connected ASes. IP addresses included in the network diagram are as follows:

- 10.1.12.1—AS 650001 external IP address on R1
- 10.1.45.2—AS 650001 external IP address on R5
- 10.0.0.1—Internal loopback (lo0) IP address for R1

- 10.0.0.5—Internal loopback (lo0) IP address for R5
- 10.1.12.2—AS 65002 external IP address on R2
- 10.1.45.1—AS 65002 external IP address on R5
- 10.0.0.2—Internal loopback (lo0) address for R2
- 10.0.0.3—Internal loopback (lo0) address for R3
- 10.0.0.4—Internal loopback (lo0) address for R4
- 10.0.0.6—Internal loopback (lo0) address for R6

All routers within each AS maintain an IBGP session between each router in that AS. R1 and R5 have an IBGP session through their loopback (lo0) interfaces: 10.0.0.1 and 10.0.0.5. R2, R3, R4, and R6 maintain IBGP sessions between each other through their loopback (lo0) interfaces: 10.0.0.2, 10.0.0.3, 10.0.0.4, and 10.0.0.6.

The two routers in AS 65001 each contain one EBGP link to AS 65002 (R2 and R4) over which they announce aggregated prefixes: 100.100/16. Routers at the edge of a network that communicate directly with routers in other networks are called border routers. Border routers use EBGP to exchange routing information between networks.

Adjacent BGP routers are referred to as neighbors or peers. Peers can be internal or external to the AS. Internal and external peers are configured slightly differently. In general, internal peers communicate using the loopback (lo0) interface, and external peers communicate through the shared interface. See Figure 14 for the loopback (lo0) and interface information.

Steps To Take To verify the BGP configuration of a router in your network, follow these steps:

1. Verify BGP on an Internal Router on page 159
2. Verify BGP on a Border Router on page 162

Step 1: Verify BGP on an Internal Router

Action To verify the BGP configuration of an internal router, enter the following JUNOS command-line interface (CLI) command:

```
user@host> show configuration
```

Sample Output The following sample output is for a BGP configuration on R3 in the network shown in Figure 14:

```
user@R3> show configuration
[...Output truncated...]
interfaces {
  so-0/0/1 {
    unit 0 {
      family inet {
        address 10.1.23.2/30;
      }
      family iso;
```

```

    }
  }
  so-0/0/3 {
    unit 0 {
      family inet {
        address 10.1.36.1/30;
      }
      family iso;
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 10.0.0.3/32;
      }
      family iso {
        address 49.0002.1000.0000.0003.00;
      }
    }
  }
}
routing-options {
  [...Output truncated...]
  router-id 10.0.0.3;
  autonomous-system 65002;
}
protocols {
  bgp {
    group internal {
      type internal;
      local-address 10.0.0.3;
      neighbor 10.0.0.2;
      neighbor 10.0.0.4;
      neighbor 10.0.0.6;
    }
  }
  isis {
    level 1 disable;
    interface all {
      level 2 metric 10;
    }
    interface lo0.0;
  }
}

user@R6> show configuration |
[Output truncated...]
interfaces {
  so-0/0/1 {
    unit 0 {
      family inet {
        address 10.1.46.2/30;
      }
      family iso;
    }
  }
  so-0/0/3 {
    unit 0 {
      family inet {
        address 10.1.36.2/30;
      }
      family iso;
    }
  }
}

```



```

    }
    lo0 {
        unit 0 {
            family inet {
                address 10.0.0.6/32;
            }
            family iso {
                address 49.0003.1000.0000.0006.00;
            }
        }
    }
}
routing-options {
    [Output truncated...]
    router-id 10.0.0.6;
    autonomous-system 65002;
}
protocols {
    bgp {
        group internal {
            type internal;
            local-address 10.0.0.6;
            neighbor 10.0.0.2;
            neighbor 10.0.0.3;
            neighbor 10.0.0.4;
        }
    }
    isis {
        level 1 disable;
        interface all {
            level 2 metric 10;
        }
        interface lo0.0;
    }
}

```

What It Means The sample output shows a basic BGP configuration on routers R3 and R6. The local AS (65002) and one group (internal) are configured on both routers. R3 has three internal peers—10.0.0.2, 10.0.0.4, and 10.0.0.6—included at the [protocols bgp group group] hierarchy level. R6 also has three internal peers: 10.0.0.2, 10.0.0.3, and 10.0.0.4. The underlying IGP protocol is Intermediate System-to-Intermediate System (IS-IS), and relevant interfaces are configured to run IS-IS.

Note that in this configuration the router ID is manually configured to avoid any duplicate router ID problems.

Step 2: Verify BGP on a Border Router

Action To verify the BGP configuration of a border router, enter the following JUNOS CLI operational mode command:

```
user@host> show configuration
```

Sample Output The following sample output is for a BGP configuration on two border routers from AS 65002 (R2 and R4) shown in Figure 14:

```
user@R2> show configuration
[...Output truncated...]
interfaces {
  so-0/0/0 {
    unit 0 {
      family inet {
        address 10.1.12.2/30;
      }
      family iso;
    }
  }
  so-0/0/1 {
    unit 0 {
      family inet {
        address 10.1.23.1/30;
      }
      family iso;
    }
  }
  so-0/0/3 {
    unit 0 {
      family inet {
        address 10.1.24.1/30;
      }
      family iso;
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 10.0.0.2/32;
      }
      family iso {
        address 49.0002.1000.0000.0002.00;
      }
    }
  }
}
routing-options {
  [...Output truncated...]
  router-id 10.0.0.2;
  autonomous-system 65002;
}
protocols {
  bgp {
    group internal {
      type internal;
      export next-hop-self;
      neighbor 10.0.0.3;
      neighbor 10.0.0.4;
      neighbor 10.0.0.6;
```

```

    }
    group toR1 {
        type external;
        import import-toR1;
        peer-as 65001;
        neighbor 10.1.12.1;
    }
}
isis {
    level 1 disable;
    interface all {
        level 2 metric 10;
    }
    interface lo0.0;
}
}
policy-options {
    policy-statement next-hop-self {
        term change-next-hop {
            from neighbor 10.1.12.1;
            then {
                next-hop self;
            }
        }
    }
    policy-statement import-toR1 {
        term 1 {
            from {
                route-filter 100.100.1.0/24 exact;
            }
            then {
                local-preference 200;
            }
        }
    }
}

```

```

user@R4> show configuration
[...Output truncated...]
interfaces {
    so-0/0/1 {
        unit 0 {
            family inet {
                address 10.1.46.1/30;
            }
            family iso;
        }
    }
    so-0/0/2 {
        unit 0 {
            family inet {
                address 10.1.45.1/30;
            }
            family iso;
        }
    }
    so-0/0/3 {
        unit 0 {
            family inet {
                address 10.1.24.2/30;
            }
            family iso;
        }
    }
}

```

```

    lo0 {
        unit 0 {
            family inet {
                address 10.0.0.4/32;
            }
            family iso {
                address 49.0001.1000.0000.0004.00;
            }
        }
    }
}
routing-options {
    [...Output truncated...]
    router-id 10.0.0.4;
    autonomous-system 65002;
}
protocols {
    bgp {
        group internal {
            type internal;
            local-address 10.0.0.4;
            export next-hop-self;
            neighbor 10.0.0.2;
            neighbor 10.0.0.3;
            neighbor 10.0.0.6;
        }
        group toR5 {
            type external;
            peer-as 65001;
            neighbor 10.1.45.2;
        }
    }
    isis {
        level 1 disable;
        interface all {
            level 2 metric 10;
        }
        interface lo0.0;
    }
}
policy-options {
    policy-statement next-hop-self {
        term change-next-hop {
            from neighbor 10.1.45.2;
            then {
                next-hop self;
            }
        }
    }
}

```

What It Means The sample output shows a basic BGP configuration on border routers R2 and R4. Both routers have the AS (65002) included at the [routing-options] hierarchy level. Each router has two groups included at the [protocols bgp group group] hierarchy level. External peers are included in the external group, either toR1 or toR5, depending on the router. Internal peers are included in the internal group. The underlying IGP protocol is IS-IS on both routers, and relevant interfaces are configured to run IS-IS.

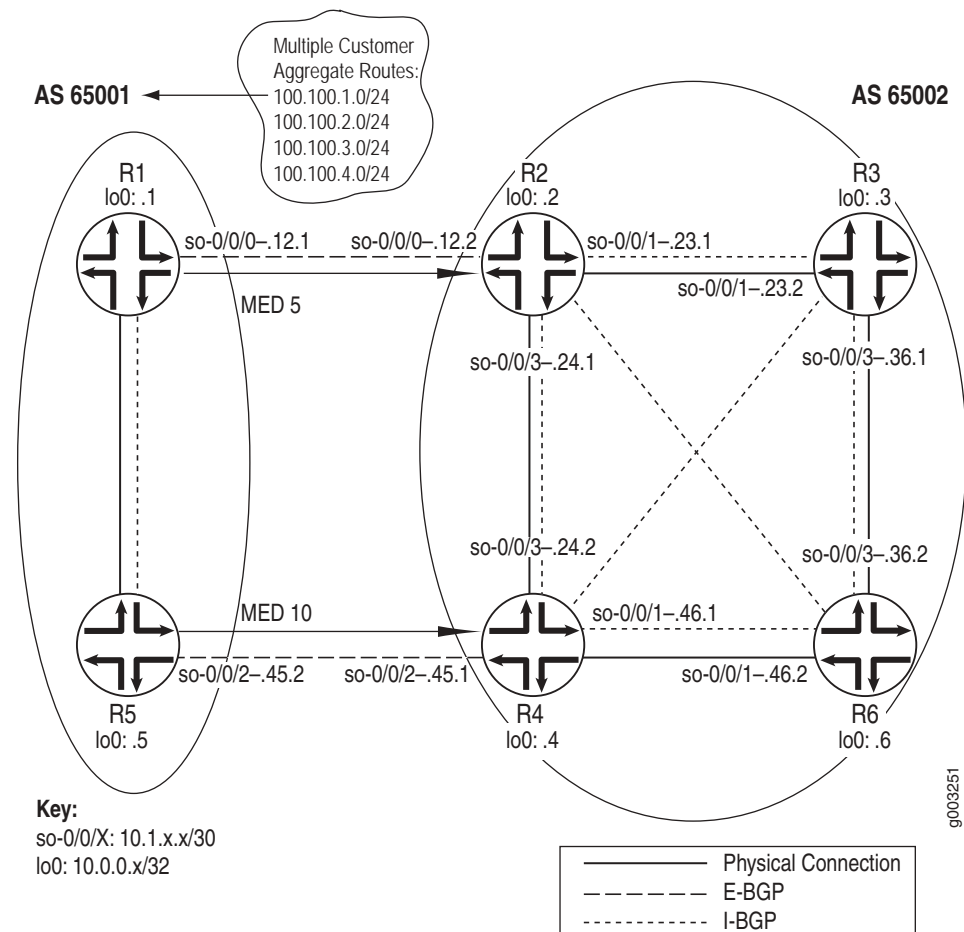
Note that in the configuration on both routers, the router ID is manually configured to avoid duplicate router ID problems, and the **next-hop-self** statement is included to avoid any BGP next-hop reachability problems.

Verify BGP Peers

Purpose Assuming that all the routers are correctly configured for BGP, you can verify if IBGP and EBGP sessions are properly established, external routes are advertised and received correctly, and the BGP path selection process is working properly.

Figure 15 illustrates an example BGP network topology used in this section

Figure 15: BGP Network Topology



The network consists of two directly connected ASes consisting of external and internal peers. The external peers are directly connected through a shared interface and are running EBGP. The internal peers are connected through their loopback (lo0) interfaces through IBGP. AS 65001 is running OSPF and AS 65002 is running IS-IS as its underlying IGP. IBGP routers do not have to be directly connected, the underlying IGP allows neighbors to reach one another.

The two routers in AS 65001 each contain one EBGP link to AS 65002 (R2 and R4) over which they announce aggregated prefixes: 100.100.1.0, 100.100.2.0, 100.100.3.0, and 100.100.4.0. Also, R1 and R5 are injecting multiple exit discriminator (MED) values of 5 and 10, respectively, for some routes.

The internal routers in both ASs are using a full mesh IBGP topology. A full mesh is required because the networks are not using confederations or route reflectors, so any routes learned through IBGP are not distributed to other internal neighbors. For example, when R3 learns a route from R2, R3 does not distribute that route to R6 because the route is learned through IBGP, so R6 must have a direct BGP connection to R2 to learn the route.

In a full mesh topology, only the border router receiving external BGP information distributes that information to other routers within its AS. The receiving router does not redistribute that information to other IBGP routers in its own AS.

From the point of view of AS 65002, the following sessions should be up:

- The four routers in AS 65002 should have IBGP sessions established with each other.
- R2 should have an EBGP session established with R1.
- R4 should have an EBGP session established with R5.

To verify BGP peers, follow these steps:

1. Check That BGP Sessions Are Up on page 166
2. Verify That a Neighbor is Advertising a Particular Route on page 169
3. Verify That a Particular BGP Route Is Received on Your Router on page 170

Step 1: Check That BGP Sessions Are Up

Action To check that all IBGP and EBGP sessions are properly established, enter the following JUNOS CLI operational mode command:

```
user@host> show bgp summary
```

Sample Output 1 The following sample output from R2 shows four peers that are *not* established:

```
user@R2> show bgp summary
Groups: 2 Peers: 4 Down peers: 1
Table Tot Paths Act Paths Suppressed History Damp State Pending
inet.0 6 4 0 0 0 0 0
Peer AS InPkt OutPkt OutQ Flaps Last Up/Dwn State|#Active/Received/Damped...
10.0.0.3 65002 86 90 0 2 42:54 0/0/0 0/0/0
10.0.0.4 65002 90 91 0 1 42:54 0/2/0 0/0/0
10.0.0.6 65002 87 90 0 3 3 Active
10.1.12.1 65001 89 89 0 1 42:54 4/4/0 0/0/0
```

Sample Output 2 The following sample output shows that all peers are established:

```
user@R2> show bgp summary
Groups: 2 Peers: 4 Down peers: 0
Table Tot Paths Act Paths Suppressed History Damp State Pending
inet.0 6 4 0 0 0 0 0
Peer AS InPkt OutPkt OutQ Flaps Last Up/Dwn State|#Active/Received/Damped..
10.0.0.3 65002 54440 54457 0 1 2w4d21h 0/0/0 0/0/0
10.0.0.4 65002 51505 51507 0 0 2w3d21h 0/2/0 0/0/0
```

```

10.0.0.6      65002      14066      14070      0      1 4d 21:15:14 0/0/0      0/0/0
10.1.12.1    65001      88580      88587      0      0      4w2d18h 4/4/0      0/0/0

```

```
user@R3> 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	0
Peer	AS	InPkt	OutPkt	OutQ	Flaps	Last Up/Dwn	State #Active/Received/Damped..
10.0.0.2	65002	54476	54460	0	1	7:17	4/4/0 0/0/0
10.0.0.4	65002	51527	51511	0	0	2w3d21h	0/2/0 0/0/0
10.0.0.6	65002	54459	54459	0	0	2w4d21h	0/0/0 0/0/0

```
user@R4> show bgp summary
```

```
Groups: 2 Peers: 4 Down peers: 0
```

Table	Tot Paths	Act Paths	Suppressed	History	Damp	State	Pending
inet.0	8	4	0	0	0	0	0
Peer	AS	InPkt	OutPkt	OutQ	Flaps	Last Up/Dwn	State #Active/Received/Damped...
10.0.0.2	65002	51530	51532	0	1	8:59	2/4/0 0/0/0
10.0.0.3	65002	51512	51531	0	0	2w3d21h	0/0/0 0/0/0
10.0.0.6	65002	51515	51530	0	0	2w3d21h	0/0/0 0/0/0
10.1.45.2	65001	51520	51530	0	0	2w3d21h	2/4/0 0/0/0

```
user@R6> 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	0
Peer	AS	InPkt	OutPkt	OutQ	Flaps	Last Up/Dwn	State #Active/Received/Damped..
10.0.0.2	65002	88610	88587	0	2	10:30	2/4/0 0/0/0
10.0.0.3	65002	54464	54468	0	1	2w4d22h	0/0/0 0/0/0
10.0.0.4	65002	51532	51520	0	0	2w3d21h	2/2/0

What It Means Sample output 1 shows a peer that is not established, while sample output 2 shows that all IBGP and EBGP sessions shown in the network diagram in Figure 15 are established.

Sample output 1 shows one peer (10.0.0.6) is not established, as indicated by the **Down Peers: 1** field. The **State|#Active/Received/Damped** column also displays **Active**, indicating that the peer is in the **Active** state and not yet established.

The **State|#Active/Received/Damped** column is a dual purpose field, as indicated by the pipe (|). When a peer is not established (as in our example), the field indicates the state of the peering session, such as **active**, **connect**, or **idle**. When the peer is in the **Established** state, the field displays the *number* of active, received, or dampened routes received from the neighbor. For example, peer 10.0.0.6 is an example of the last column displaying the **Active** state, and peer 10.1.12.1 is an example of the last column showing the number of active, received, and dampened routes (4/4/0). For more detailed information on the **show bgp summary** command, see the *JUNOS Routing Protocols and Policies Command Reference*.

If the BGP neighbor session is not established, use the **ping** and **show route** commands to verify network connectivity to the BGP neighbor. Also, use the **show log messages** command to look for any errors pertaining to the peer in question.

Sample output 2 shows that all IBGP and EBGP sessions shown in the network diagram in Figure 15 are established, as indicated by the **Down Peers: 0** field and the last column that shows the number of routes.

Following is a description of the output for all established BGP peers, R2, R3, R4, and R6.

The route information for border router R2 shows the following:

- 0/0/0 for internal peers 10.0.0.3 and 10.0.0.6, indicating that no BGP routes are received or active in the routing table from those peers. No BGP routes are damped.
- 0/2/0 for internal peer 10.0.0.4, indicating that two received BGP routes are not active in the routing table. No BGP routes are damped.
- 4/4/0 for external peer 10.1.12.1, indicating that four received BGP routes are active in the routing table. No BGP routes are damped.

The route information for internal router R3 shows the following:

- 0/0/0 for internal peer 10.0.0.6, indicating that no BGP routes are received or active in the routing table from that peer. No BGP routes are damped.
- 0/2/0 for internal peer 10.0.0.4, indicating that two received BGP routes are not active in the routing table. No BGP routes are damped.
- 4/4/0 for internal peer 10.0.0.2, indicating that four received BGP routes are active in the routing table. No BGP routes are damped.

The route information for border router R4 shows the following:

- 0/0/0 for internal peers 10.0.0.3 and 10.0.0.6, indicating that no BGP routes are received or active in the routing table from those peers. No BGP routes are damped.
- 2/4/0 for internal peer 10.0.0.2 and external peer 10.1.45.1, indicating that two BGP routes are active in the routing table, but four are received. No BGP routes are damped.

The route information for internal router R6 shows the following:

- 0/0/0 for internal peer 10.0.0.3, indicating that no BGP routes are received or active in the routing table from that peer. No BGP routes are damped.
- 2/4/0 for internal peers 10.0.0.2, and 10.0.0.4, indicating that of the four received BGP routes, two are active in the routing table. No BGP routes are damped.

Other information in the sample output includes the following:

- Number of configured BGP groups: R2 has two groups configured (internal and toR1), and R4 also has two BGP groups configured (internal and toR5).
- Number of BGP peers to which the router is linked: R2 and R4 have four (one EBGP and three IBGP), and R3 and R6 have three IBGP. One peer is down (R6) in sample output 1.

- The name of the routing table storing the BGP routes, all routers are using `inet.0`.
- The total number of BGP paths, for example, **R4** has a total of eight BGP paths from all of its peers.
- The number of active BGP routes, for example, **R4** has a total of four active BGP routes from all of its peers.
- The second column of values (0/0/0) in the `State|#Active/Received/Damped` field indicates the number of multiprotocol BGP (MBGP) routes. All routers do not have any received, active, or damped MBGP routes.

Step 2: Verify That a Neighbor is Advertising a Particular Route

Purpose You can determine if a particular route that you have configured is being advertised by a neighbor.

Action To verify that a neighbor is advertising a particular route, enter the following JUNOS CLI operational mode command:

```
user@host> show route advertising-protocol bgp neighbor-address
```

Sample Output user@R2> show route advertising-protocol bgp 10.0.0.4

```
inet.0: 20 destinations, 22 routes (20 active, 0 holddown, 0 hidden)
  Prefix            Nexthop          MED      Lc1pref    AS path
* 100.100.1.0/24     Self             5         200        65001 I
* 100.100.2.0/24     Self             5         100        65001 I
* 100.100.3.0/24     Self             100       65001 I
* 100.100.4.0/24     Self             100       65001 I
```

What It Means The sample output shows the BGP routes advertised from **R2** to its neighbor, **10.0.0.4 (R4)**. Out of 22 total routes in the `inet.0` routing table, 20 are active destinations. No routes are **hidden** or in the **hold-down** state. Routes reside in the **hold-down** state prior to being declared active, and routes rejected by a routing policy can be placed into the **hidden** state. The information displayed reflects the routes that the routing table exported to the BGP routing protocol.

Step 3: Verify That a Particular BGP Route Is Received on Your Router

Purpose Display the routing information as it is received through a particular BGP neighbor and advertised by the local router to the neighbor.

Action To verify that a particular BGP route is received on your router, enter the following JUNOS CLI operational mode command:

```
user@host> show route receive-protocol bgp neighbor-address
```

Sample Output user@R6> show route receive-protocol bgp 10.0.0.2

```
inet.0: 18 destinations, 20 routes (18 active, 0 holddown, 0 hidden)
  Prefix                Nexthop          MED      Lclpref    AS path
* 100.100.1.0/24         10.0.0.2          5         200        65001 I
* 100.100.2.0/24         10.0.0.2          5         100        65001 I
  100.100.3.0/24         10.0.0.2          100       65001 I
  100.100.4.0/24         10.0.0.2          100       65001 I
```

```
iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

```
user@R6> show route receive-protocol bgp 10.0.0.4
```

```
inet.0: 18 destinations, 20 routes (18 active, 0 holddown, 0 hidden)
  Prefix                Nexthop          MED      Lclpref    AS path
* 100.100.3.0/24         10.0.0.4          100       65001 I
* 100.100.4.0/24         10.0.0.4          100       65001 I
```

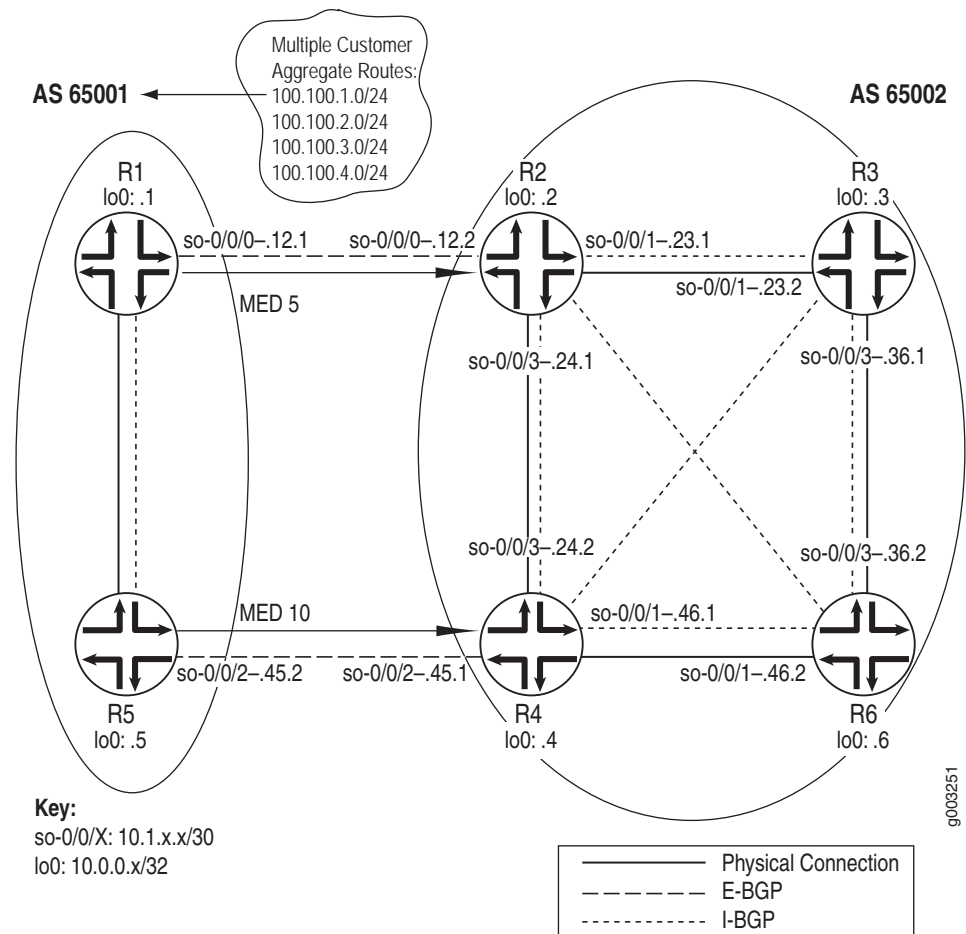
```
iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

What It Means The sample output shows four BGP routes from R2 and two from R4. Of the four routes from R2, only two are active in the routing table, as indicated by the asterisk (*), while both routes received from R4 are active in the routing table. All BGP routes came through AS 65001.

Examine BGP Routes and Route Selection

Purpose You can examine the BGP path selection process to determine the single, active path when BGP receives multiple routes to the same destination prefix.

Figure 16: BGP Network Topology



The network in Figure 16 shows that R1 and R5 announce the same aggregate routes to R2 and R4, which results in R2 and R4 receiving two routes to the same destination prefix. The route selection process on R2 and R4 determines which of the two BGP routes received is active and advertised to the other internal routers (R3 and R6).

Before the router installs a BGP route, it must make sure that the BGP **next-hop** attribute is reachable. If the BGP next hop cannot be resolved, the route is not installed. When a BGP route is installed in the routing table, it must go through a path selection process if multiple routes exist to the same destination prefix. The BGP path selection process proceeds in the following order:

1. Route preference in the routing table is compared. For example, if an OSPF and a BGP route exist for a particular destination, the OSPF route is selected as the active route because the OSPF route has a default preference of 10, while the BGP route has a default preference of 170.
2. Routes are compared for local preference. The route with the highest local preference is preferred. For example, see “Examine the Local Preference Selection” on page 173.
3. The AS path attribute is evaluated. The shorter AS path is preferred.
4. The origin code is evaluated. The lowest origin code is preferred (I (IGP) < E (EGP) < ? (Incomplete)).
5. The MED value is evaluated. By default, if any of the routes are advertised from the same neighboring AS, the lowest MED value is preferred. The absence of a MED value is interpreted as a MED of 0. For an example, see “Examine the Multiple Exit Discriminator Route Selection” on page 174.
6. The route is evaluated as to whether it is learned through EBGp or IBGP. EBGp learned routes are preferred to IBGP learned routes. For an example, see “Examine the EBGp over IBGP Selection” on page 175.
7. If the route is learned from IBGP, the route with the lowest IGP cost is preferred. For an example, see “Examine the IGP Cost Selection” on page 176. The physical next hop to the IBGP peer is installed according to the following three rules:
 - a. After BGP examines the `inet.0` and `inet.3` routing tables, the physical next hop of the route with the lowest preference is used.
 - b. If the preference values in the `inet.0` and the `inet.3` routing tables are a tie, the physical next hop of the route in the `inet.3` routing table is used.
 - c. When a preference tie exists in the same routing table, the physical next hop of the route with more paths is installed.
8. The route reflection cluster list attribute is evaluated. The shortest length cluster list is preferred. Routes without a cluster list are considered to have a cluster list length of 0.
9. The router ID is evaluated. The route from the peer with the lowest router ID is preferred (usually the loopback address).
10. The peer address value is examined. The peer with the lowest peer IP address is preferred.

Steps To Take To determine the single, active path when BGP receives multiple routes to the same destination prefix, enter the following JUNOS CLI operational mode command:

```
user@host> show route destination-prefix < detail >
```

The following steps illustrate the inactive reason displayed when BGP receives multiple routes to the same destination prefix and one route is selected as the single, active path:

1. Examine the Local Preference Selection on page 173
2. Examine the Multiple Exit Discriminator Route Selection on page 174
3. Examine the EBGp over IBGP Selection on page 175
4. Examine the IGP Cost Selection on page 176

Step 1: Examine the Local Preference Selection

Action To examine a route to determine if local preference is the selection criteria for the single, active path, enter the following JUNOS CLI operational mode command:

```
user@host> show route destination-prefix < detail >
```

Sample Output user@R4> show route 100.100.1.0 detail

```
inet.0: 20 destinations, 24 routes (20 active, 0 holddown, 0 hidden)
100.100.1.0/24 (2 entries, 1 announced)
  *BGP      Preference: 170/-201
    Source: 10.0.0.2
    Next hop: 10.1.24.1 via so-0/0/3.0, selected
    Protocol next hop: 10.0.0.2 Indirect next hop: 8644000 277
    State: <Active Int Ext>
    Local AS: 65002 Peer AS: 65002
    Age: 2:22:34 Metric: 5 Metric2: 10
    Task: BGP_65002.10.0.0.2+179
    Announcement bits (3): 0-KRT 3-BGP.0.0.0.0+179 4-Resolve inet.0
    AS path: 65001 I
    Localpref: 200
    Router ID: 10.0.0.2
  BGP      Preference: 170/-101
    Source: 10.1.45.2
    Next hop: 10.1.45.2 via so-0/0/2.0, selected
    State: <Ext>
    Inactive reason: Local Preference
    Local AS: 65002 Peer AS: 65001
    Age: 2w0d 1:28:31 Metric: 10
    Task: BGP_65001.10.1.45.2+179
    AS path: 65001 I
    Localpref: 100
    Router ID: 10.0.0.5
```

What It Means The sample output shows that R4 received two instances of the 100.100.1.0 route: one from 10.0.0.2 (R2) and one from 10.1.45.2 (R5). R4 selected the path from R2 as its active path, as indicated by the asterisk (*). The selection is based on the local preference value contained in the **Localpref** field. The path with the *highest* local preference is preferred. In the example, the path with the higher local preference value is the path from R2, 200.

The reason that the route from R5 is not selected is in the **Inactive reason** field, in this case, **Local Preference**.

Note that the two paths are from the same neighboring network: AS 65001.

Step 2: Examine the Multiple Exit Discriminator Route Selection

Action To examine a route to determine if the MED is the selection criteria for the single, active path, enter the following JUNOS CLI operational mode command:

```
user@host> show route destination-prefix < detail >
```

Sample Output user@R4> show route 100.100.2.0 detail

```
inet.0: 20 destinations, 24 routes (20 active, 0 holddown, 0 hidden)
100.100.2.0/24 (2 entries, 1 announced)
  *BGP      Preference: 170/-101
             Source: 10.0.0.2
             Next hop: 10.1.24.1 via so-0/0/3.0, selected
             Protocol next hop: 10.0.0.2 Indirect next hop: 8644000 277
             State: <Active Int Ext>
             Local AS: 65002 Peer AS: 65002
             Age: 2:32:01      Metric: 5      Metric2: 10
             Task: BGP_65002.10.0.0.2+179
             Announcement bits (3): 0-KRT 3-BGP.0.0.0.0+179 4-Resolve inet.0
             AS path: 65001 I
             Localpref: 100
             Router ID: 10.0.0.2
  BGP      Preference: 170/-101
             Source: 10.1.45.2
             Next hop: 10.1.45.2 via so-0/0/2.0, selected
             State: <NotBest Ext>
             Inactive reason: Not Best in its group
             Local AS: 65002 Peer AS: 65001
             Age: 2w0d 1:37:58      Metric: 10
             Task: BGP_65001.10.1.45.2+179
             AS path: 65001 I
             Localpref: 100
             Router ID: 10.0.0.5
```

What It Means The sample output shows that R4 received two instances of the 100.100.2.0 route: one from 10.0.0.2 (R2), and one from 10.1.45.2 (R5). R4 selected the path from R2 as its active route, as indicated by the asterisk (*). The selection is based on the MED value contained in the **Metric** field. The path with the lowest MED value is preferred. In the example, the path with the lowest MED value (5) is the path from R2. Note that the two paths are from the same neighboring network: AS 65001.

The reason that the inactive path is not selected is displayed in the **Inactive reason** field, in this case, **Not Best in its group**. The wording is used because the JUNOS software uses the process of deterministic MED selection, by default.

Step 3: Examine the EBGp over IBGP Selection

Action To examine a route to determine if EBGp is selected over IBGP as the selection criteria for the single, active path, enter the following JUNOS CLI operational mode command:

```
user@host> show route destination-prefix < detail >
```

Sample Output user@R4> show route 100.100.3.0 detail

```
inet.0: 20 destinations, 24 routes (20 active, 0 holddown, 0 hidden)
100.100.3.0/24 (2 entries, 1 announced)
  *BGP    Preference: 170/-101
          Source: 10.1.45.2
          Next hop: 10.1.45.2 via so-0/0/2.0, selected
          State: <Active Ext>
          Local AS: 65002 Peer AS: 65001
          Age: 5d 0:31:25
          Task: BGP_65001.10.1.45.2+179
          Announcement bits (3): 0-KRT 3-BGP.0.0.0+179 4-Resolve inet.0
          AS path: 65001 I
          Localpref: 100
          Router ID: 10.0.0.5
  BGP    Preference: 170/-101
          Source: 10.0.0.2
          Next hop: 10.1.24.1 via so-0/0/3.0, selected
          Protocol next hop: 10.0.0.2 Indirect next hop: 8644000 277
          State: <NotBest Int Ext>
          Inactive reason: Interior > Exterior > Exterior via Interior
          Local AS: 65002 Peer AS: 65002
          Age: 2:48:18   Metric2: 10
          Task: BGP_65002.10.0.0.2+179
          AS path: 65001 I
          Localpref: 100
          Router ID: 10.0.0.2
```

What It Means The sample output shows that R4 received two instances of the 100.100.3.0 route: one from 10.1.45.2 (R5) and one from 10.0.0.2 (R2). R4 selected the path from R5 as its active path, as indicated by the asterisk (*). The selection is based on a preference for routes learned from an EBGp peer over routes learned from an IBGP. R5 is an EBGp peer.

You can determine if a path is received from an EBGp or IBGP peer by examining the **Local As** and **Peer As** fields. For example, the route from R5 shows the local AS is 65002 and the peer AS is 65001, indicating that the route is received from an EBGp peer. The route from R2 shows that both the local and peer AS is 65002, indicating that it is received from an IBGP peer.

The reason that the inactive path is not selected is displayed in the **Inactive reason** field, in this case, **Interior > Exterior > Exterior via Interior**. The wording of this reason shows the order of preferences applied when the same route is received from two routers. The route received from a strictly internal source (IGP) is preferred first, the route received from an external source (EBGP) is preferred next, and any route which comes from an external source and is received internally (IBGP) is preferred last.

Step 4: Examine the IGP Cost Selection

Action To examine a route to determine if EBGP is selected over IBGP as the selection criteria for the single, active path, enter the following JUNOS CLI operational mode command:

```
user@host> show route destination-prefix < detail >
```

Sample Output user@R6> show route 100.100.4.0 detail

```
inet.0: 18 destinations, 20 routes (18 active, 0 holddown, 0 hidden)
100.100.4.0/24 (2 entries, 1 announced)
  *BGP      Preference: 170/-101
            Source: 10.0.0.4
            Next hop: 10.1.46.1 via so-0/0/1.0, selected
            Protocol next hop: 10.0.0.4 Indirect next hop: 864c000 276
            State: <Active Int Ext>
            Local AS: 65002 Peer AS: 65002
            Age: 2:16:11    Metric2: 10
            Task: BGP_65002.10.0.0.4+4120
            Announcement bits (2): 0-KRT 4-Resolve inet.0
            AS path: 65001 I
            Localpref: 100
            Router ID: 10.0.0.4
  BGP      Preference: 170/-101
            Source: 10.0.0.2
            Next hop: 10.1.46.1 via so-0/0/1.0, selected
            Next hop: 10.1.36.1 via so-0/0/3.0
            Protocol next hop: 10.0.0.2 Indirect next hop: 864c0b0 278
            State: <NotBest Int Ext>
            Inactive reason: IGP metric
            Local AS: 65002 Peer AS: 65002
            Age: 2:16:03    Metric2: 20
            Task: BGP_65002.10.0.0.2+179
            AS path: 65001 I
            Localpref: 100
            Router ID: 10.0.0.2
```

What It Means The sample output shows that R6 received two instances of the 100.100.4.0 route: one from 10.0.0.4 (R4) and one from 10.0.0.2 (R2). R6 selected the path from R4 as its active route, as indicated by the asterisk (*). The selection is based on the IGP metric, displayed in the **Metric2** field. The route with the lowest IGP metric is preferred. In the example, the path with the lowest IGP metric value is the path from R4, with an IGP metric value of 10, while the path from R2 has an IGP metric of 20. Note that the two paths are from the same neighboring network: AS 65001.

The reason that the inactive path was not selected is displayed in the **Inactive reason** field, in this case, **IGP metric**.

Examine Routes in the Forwarding Table

Purpose When you run into problems, such as connectivity problems, you may need to examine routes in the forwarding table to verify that the routing protocol process has relayed the correct information into the forwarding table.

Action To display the set of routes installed in the forwarding table, enter the following JUNOS CLI operational mode command:

```
user@host> show route forwarding-table
```

Sample Output

```
user@R2> show route forwarding-table
Routing table: inet
Internet:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm  0
10.0.0.2/32      intf  0 10.0.0.2          locl  256  1
10.0.0.3/32      user  1 10.1.23.0         ucst  282  4 so-0/0/1.0
10.0.0.4/32      user  1 10.1.24.0         ucst  290  7 so-0/0/3.0
10.0.0.6/32      user  1 10.1.24.0         ucst  290  7 so-0/0/3.0
10.1.12.0/30     intf  1 ff.3.0.21        ucst  278  6 so-0/0/0.0
10.1.12.0/32     dest  0 10.1.12.0         recv  280  1 so-0/0/0.0
10.1.12.2/32     intf  0 10.1.12.2         locl  277  1
10.1.12.3/32     dest  0 10.1.12.3         bcst  279  1 so-0/0/0.0
10.1.23.0/30     intf  0 ff.3.0.21        ucst  282  4 so-0/0/1.0
10.1.23.0/32     dest  0 10.1.23.0         recv  284  1 so-0/0/1.0
10.1.23.1/32     intf  0 10.1.23.1         locl  281  1
10.1.23.3/32     dest  0 10.1.23.3         bcst  283  1 so-0/0/1.0
10.1.24.0/30     intf  0 ff.3.0.21        ucst  290  7 so-0/0/3.0
10.1.24.0/32     dest  0 10.1.24.0         recv  292  1 so-0/0/3.0
10.1.24.1/32     intf  0 10.1.24.1         locl  289  1
10.1.24.3/32     dest  0 10.1.24.3         bcst  291  1 so-0/0/3.0
10.1.36.0/30     user  0 10.1.23.0         ucst  282  4 so-0/0/1.0
10.1.46.0/30     user  0 10.1.24.0         ucst  290  7 so-0/0/3.0
100.100.1.0/24   user  0 10.1.12.0         ucst  278  6 so-0/0/0.0
100.100.2.0/24   user  0 10.1.12.0         ucst  278  6 so-0/0/0.0
100.100.3.0/24   user  0 10.1.12.0         ucst  278  6 so-0/0/0.0
100.100.4.0/24   user  0 10.1.12.0         ucst  278  6 so-0/0/0.0
[...Output truncated...]
```

What It Means The sample output shows the network-layer prefixes and their next hops installed in the forwarding table. The output includes the same next-hop information as in the `show route detail` command (the next-hop address and interface name). Additional information includes the destination type, the next-hop type, the number of references to this next hop, and an index into an internal next-hop database. (The internal database contains additional information used by the Packet Forwarding Engine to ensure proper encapsulation of packets sent out an interface. This database is not accessible to the user.)

For detailed information about the meanings of the various flags and types fields, see the *JUNOS Routing Protocols and Policies Command Reference*.

Chapter 14

Verify the Routing Engine CPU Memory

This chapter describes how to verify the Routing Engine CPU memory on your Juniper Networks router. (See Table 35.)

Table 35: Checklist for Verifying the Routing Engine CPU Memory

Verify the Routing Engine CPU Memory Tasks	Command or Action
Check the Routing CPU Memory Usage on page 180	
1. Check Overall CPU and Memory Usage on page 180	show system process extensive
2. Check Routing Protocol Process (rpd) Memory Usage on page 183	show route summary show task memory detail
3. Display Tasks on page 185	show task show task memory show task <i>task-name</i>

Check the Routing CPU Memory Usage

Purpose Software processes on the router can consume a considerable amount of CPU and memory. The routing protocol process (rpd) can consume enormous amounts of memory to store information needed for the operation of routing and related protocols, such as Border Gateway Protocol (BGP), Open Shortest Path First (OSPF), Intermediate System-to-Intermediate System (ISIS), Resource Reservation Protocol (RSVP), Label Distribution Protocol (LDP), and Multiprotocol Label Switching (MPLS).

Steps To Take To verify the traffic passing through the router and check memory utilization, follow these steps:

1. Check Overall CPU and Memory Usage on page 180
2. Check Routing Protocol Process (rpd) Memory Usage on page 183
3. Display Tasks on page 185

Step 1: Check Overall CPU and Memory Usage

Purpose You can display exhaustive system process information about software processes that are running on the router and have controlling terminals. This command is equivalent to the UNIX **top** command. However, the UNIX **top** command shows real-time memory usage, with the memory values constantly changing, while the **show system processes extensive** command provides a snapshot of memory usage in a given moment.

Action To check overall CPU and memory usage, enter the following JUNOS command-line interface (CLI) command:

```
user@host> show system processes extensive
```

Sample Output

```
user@R1> show system processes extensive
last pid: 5251; load averages: 0.00, 0.00, 0.00 up 4+20:22:16 10:44:41
58 processes: 1 running, 57 sleeping

Mem: 57M Active, 54M Inact, 17M Wired, 184K Cache, 35M Buf, 118M Free
Swap: 512M Total, 512M Free
```

PID	USERNAME	PRI	NICE	SIZE	RES	STATE	TIME	WCPU	CPU	COMMAND
4480	root	2	0	3728K	1908K	select	231:17	2.34%	2.34%	chassisd
4500	root	2	0	1896K	952K	select	0:36	0.00%	0.00%	fud
4505	root	2	0	1380K	736K	select	0:35	0.00%	0.00%	irsd
4481	root	2	0	1864K	872K	select	0:32	0.00%	0.00%	alarmd
4488	root	2	0	8464K	4600K	kqread	0:28	0.00%	0.00%	rpd
4501	root	2	-15	1560K	968K	select	0:21	0.00%	0.00%	ppmd
4510	root	2	0	1372K	812K	select	0:13	0.00%	0.00%	bfdd
5	root	18	0	0K	0K	syncer	0:09	0.00%	0.00%	syncer
4485	root	2	0	3056K	1776K	select	0:07	0.00%	0.00%	snmpd
4499	root	2	0	3688K	1676K	select	0:05	0.00%	0.00%	kmd
4486	root	2	0	3760K	1748K	select	0:05	0.00%	0.00%	mib2d
4493	root	2	0	1872K	928K	select	0:03	0.00%	0.00%	pfed
4507	root	2	0	1984K	1052K	select	0:02	0.00%	0.00%	fsad
4518	root	2	0	3780K	2400K	select	0:02	0.00%	0.00%	dcd
8	root	-18	0	0K	0K	psleep	0:02	0.00%	0.00%	vmuncachedaemo

```

  4 root      -18  0    0K    0K psleep  0:02  0.00%  0.00% bufdaemon
4690 root      2  0    0K    0K peer_s  0:01  0.00%  0.00% peer proxy
4504 root      2  0  1836K  968K select  0:01  0.00%  0.00% dfwd
4477 root      2  0   992K  320K select  0:01  0.00%  0.00% watchdog
4354 root      2  0  1116K  604K select  0:01  0.00%  0.00% syslogd
4492 root     10  0  1004K  400K nanslp  0:01  0.00%  0.00% tnp.sntpd
4446 root     10  0  1108K  616K nanslp  0:01  0.00%  0.00% cron
4484 root      2  0 15716K  7468K select  0:01  0.00%  0.00% mgd
4494 root      2 15  2936K  2036K select  0:01  0.00%  0.00% sampled
5245 remote    2  0  8340K  3472K select  0:01  0.00%  0.00% cli
  2 root     -18  0    0K    0K psleep  0:00  0.00%  0.00% pagedaemon
4512 root      2  0  2840K 1400K select  0:00  0.00%  0.00% l2tpd
  1 root     10  0   852K  580K wait   0:00  0.00%  0.00% init
5244 root      2  0  1376K  784K select  0:00  0.00%  0.00% telnetd
4509 root     10  0  1060K  528K nanslp  0:00  0.00%  0.00% eccd
4508 root      2  0  2264K 1108K select  0:00  0.00%  0.00% spd
2339 root     10  0   514M 17260K mfsidl  0:00  0.00%  0.00% newfs
4497 root      2  0  2432K 1152K select  0:00  0.00%  0.00% cosd
4490 root      2 -15  2356K 1020K select  0:00  0.00%  0.00% apsd
4496 root      2  0  2428K 1108K select  0:00  0.00%  0.00% rmopd
4491 root      2  0  2436K 1104K select  0:00  0.00%  0.00% vrrpd
4487 root      2  0 15756K  7648K sbwait  0:00  0.00%  0.00% mgd
5246 root      2  0 15776K  8336K select  0:00  0.00%  0.00% mgd
  0 root     -18  0    0K    0K sched  0:00  0.00%  0.00% swapper
5251 root     30  0 21732K  840K RUN    0:00  0.00%  0.00% top
4511 root      2  0  1964K  908K select  0:00  0.00%  0.00% pgmd
4502 root      2  0  1960K  956K select  0:00  0.00%  0.00% lmpd
4495 root      2  0  1884K  876K select  0:00  0.00%  0.00% ilmid
4482 root      2  0  1772K  776K select  0:00  0.00%  0.00% craftd
4503 root     10  0  1040K  492K nanslp  0:00  0.00%  0.00% smartd
  6 root     28  0    0K    0K sleep  0:00  0.00%  0.00% netdaemon
4498 root      2  0  1736K  932K select  0:00  0.00%  0.00% nasd
4506 root      2  0  1348K  672K select  0:00  0.00%  0.00% rtspd
4489 root      2  0  1160K  668K select  0:00  0.00%  0.00% inetd
4478 root      2  0  1108K  608K select  0:00  0.00%  0.00% tnetd
4483 root      2  0  1296K  540K select  0:00  0.00%  0.00% ntpd
4514 root      3  0  1080K  540K ttyin  0:00  0.00%  0.00% getty
4331 root      2  0   416K  232K select  0:00  0.00%  0.00% pccardd
  7 root      2  0    0K    0K pfeacc  0:00  0.00%  0.00% if_pfe_listen
 11 root      2  0    0K    0K picacc  0:00  0.00%  0.00% if_pic_listen
  3 root     18  0    0K    0K psleep  0:00  0.00%  0.00% vmdaemon
  9 root      2  0    0K    0K scs_ho  0:00  0.00%  0.00% scs_housekeepi
 10 root      2  0    0K    0K cb-pol  0:00  0.00%  0.00% cb_pol1

```

What It Means The sample output shows the amount of virtual memory used by the Routing Engine and software processes. For example, 118 MB of physical memory is free and 512 MB of the swap file is free, indicating that the router is not short of memory. The **processes** field shows that most of the 58 processes are in the **sleeping** state, with 1 in the **running** state. The process or command that is running is the **top** command.

The **commands** column lists the processes that are currently running. For example, the chassis process (chassisd) has a process identifier (PID) of 4480, with a current priority (PRI) of 2. A lower priority number indicates a higher priority.

The processes are listed according to level of activity, with the most active process at the top of the output. For example, the chassis (chassisd) process is consuming the largest amount of CPU resource at 2.34 percent.

The memory field (**Mem**) shows the virtual memory managed by the Routing Engine and used by processes. The value in the memory field is in KB and MB, and is broken down as follows:

- **Active**—Memory that is allocated and actually in use by programs.
- **Inact**—Memory that is either allocated but not recently used or memory that was freed by programs. Inactive memory is still mapped in the address space of one or more processes and, therefore, counts toward the resident set size of those processes.
- **Wired**—Memory that is not eligible to be swapped, and is usually used for Routing Engine memory structures or memory physically locked by a process.
- **Cache**—Memory that is not associated with any program and does not need to be swapped before being reused.
- **Buf**—The size of the memory buffer used to hold data recently called from disk.
- **Free**—Memory that is not associated with any programs. Memory freed by a process can become **Inactive**, **Cache**, or **Free**, depending on the method used by the process to free the memory.

When the system is under memory pressure, the pageout process reuses memory from the free, cache, inactive and, if necessary, active pages.

The **Swap** field shows the total swap space available and how much is unused. In the example, the output shows 512 MB of total swap space and 512 MB of free swap space.

Finally, the memory usage of each process is listed. The **SIZE** field indicates the size of the virtual address space, and the **RES** field indicates the amount of the program in physical memory, which is also known as RSS or Resident Set Size. In the sample output, the chassis (chassisd) process has 3728 KB of virtual address space and 1908 KB of physical memory.

For additional information about the **show system processes extensive** command, see “Stop and Start JUNOS Software” on page 37.

Step 2: Check Routing Protocol Process (rpd) Memory Usage

Purpose When you notice a lot of memory usage, you can obtain detailed information about the memory utilization of routing tasks to get an idea of what is going on. The routing process (rpd) is the main task that uses Routing Engine memory.

Action To check routing process memory usage, enter the following JUNOS CLI operational mode commands:

```
user@host> show route summary
user@host> show task memory detail
```

Sample Output

```
user@host> show route summary
Autonomous system number: 209
Router ID: 205.175.0.170

inet.0: 179783 destinations, 898393 routes (179771 active, 146 holddown, 157
hidden)
    Direct:    17 routes,    17 active
    Local:     18 routes,    18 active
    BGP: 896632 routes, 178010 active
    Static:    32 routes,    31 active
    IS-IS:    1694 routes,   1694 active

inet.2: 8766 destinations, 22700 routes (8766 active, 124 holddown, 73 hidden)
    Direct:    17 routes,    17 active
    Local:     18 routes,    18 active
    BGP: 20939 routes,  7006 active
    Static:    32 routes,    31 active
    IS-IS:    1694 routes,   1694 active

inet.3: 1614 destinations, 1719 routes (1614 active, 0 holddown, 0 hidden)
    IS-IS:    1613 routes,   1551 active
    RSVP:     45 routes,     45 active
    LDP:      61 routes,     18 active

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
    Direct:    1 routes,     1 active

mpls.0: 371 destinations, 371 routes (371 active, 0 holddown, 0 hidden)
    MPLS:      3 routes,      3 active
    RSVP:    303 routes,    303 active
    LDP:     65 routes,     65 active

user@R1> show task memory detail
----- Overall Memory Report -----
Size TP   Allocs  Mallocs  AllocBytes  MaxAllocs  MaxBytes  FreeBytes
  12      8140   186959   2341188    200824     2409888   54972
  16     4061    182     67888     4586      73376    5840
  16 T      -      -        -     393571   6297136      -
  20    688588    51   13772780   713704   14274080   423956
[...Output truncated...]
8192 P      91      -    745472     195    1597440      -
12288 P      -      -        -        1     12288      -
block      5      -   137200     14    137732    6160
pool     50      -     896    100     1792    3200
alloc      -      8   383744     10    397365    9472
-----
                                389169664                578341705    72977920
```

----- Allocator Memory Report -----						
Name	Size	Alloc	DTP	Alloc	Alloc	MaxAlloc
		Size		Blocks	Bytes	Blocks
					Bytes	Bytes
patricia_root_struct	8	12		7741	92892	8130
sockaddr_un.i802	8	12		2	24	2
sockaddr_un.tag	8	12		371	4452	995
if_addr_entry	8	12		-	-	1
gw_entry_list	8	12		1	12	1
isis_proto_list	8	12		25	300	30
struct krt_scb	12	16		4	64	6
ldp_rt_data	12	16		61	976	133
config_list	12	16		2353	37648	2353
TED NodeInfo	12	16		845	13520	907
isis_area_addr	12	16		544	8704	612
isis_nh_list	12	16		237	3792	922
isis_tsi	12	16		17	272	19
bgp_use_block	12	16		-	-	112
isis_route_walk_cont	12	16	T	-	-	1
bgpg_rtinfo_entry	12	16	T	-	-	393571
task_floating_socket	16	20		1	20	1
[...Output truncated...]						
rt_parse_memory	4092	4096	TP	-	-	1
noblock_buffer_blk	4092	4096	TP	5	20480	811
bgp_buffer	4100	8192	P	91	745472	100
bgp_outbuf	4104	8192	P	-	-	94
ldp_buffer	4108	8192	P	-	-	7
RPD SNMP	8268	12288	P	-	-	1
LDP config	various			1	896	1
					349037508	543172620
----- Malloc Usage Report -----						
Name	Allocs	Bytes	MaxAllocs	MaxBytes	FuncCalls	
MGMT.local	1	12	1	12	1	
RSVP	-	-	1	2048	156084	
BGP_Group_Tweak-RTClie	2	24	2	24	2	
[...Output truncated...]						
LDP	2	24	2	24	2	
KRT Request	-	-	1	16	446888	
BGP_Group_Packet-Design	2	24	2	24	38	
[...Output truncated...]						
MPLS	22272	1221656	22274	1221784	228522	
BGP.0.0.0.0+179	186419	2237028	192292	2307504	282141191	
IS-IS I/O./var/run/ppmd	1	66536	43	103916	695536231	
IS-IS	2407	361372	5887	446076	889294754	
BGP RT Background	3	66556	3	66556	3	
SNMP Subagent./var/run/	-	24	1	9144	3677022	
KRT	2	205616	3	207900	10	
ASPaths	13901	1581544	18023	2067605	293868769	
RT	27	556	28	580	2815	
Scheduler	194	2604	199	2684	41382	
--Anonymous--	4294944918	4293764616	4294967294	4294967292	45560848	
--System--	38565	35474324	38684	35487048	235115763	
				40015436	41923181	
Dynamically allocated memory:	485789696		Maximum:	541736960		
Program data+BSS memory:	2101248		Maximum:	2101248		
Page data overhead:	3039232		Maximum:	3039232		
Page directory size:	512000		Maximum:	512000		

Total bytes in use: 491442176 (70% of available memory)						

What It Means The sample output shows summary statistics about the entries in the routing table (`show route summary` command) and the memory usage breakdown (`show task memory detail` command) for the routing process (rpd). The two commands provide a comprehensive picture of the memory utilization of the routing protocol process.

The `show route summary` command shows the number of routes in the various routing tables. In the sample output, the routing tables represented are `inet.0`, `inet.2`, `inet.3`, `iso.0`, and `mpls.0`. Within each routing table, all of the active, hold-down, and hidden destinations and routes are summarized for all the protocols from which routes are learned. Routes are in the **hold-down** state prior to being declared inactive, and **hidden** routes are not used because of routing policy. Routes in the **hold-down** and **hidden** states are still using memory because they appear in the routing table.

In addition, routes are summarized in the following categories: those directly connected to the network (**Direct**), local routes (**Local**), and routes learned from configured routing protocols, such as BGP and IS-IS.

The `show task memory detail` command lists the data structures within the tasks run by the routing protocol process (rpd). Tasks are enabled depending on the router's configuration. For example, `isis_area_addr` is a data structure resulting from the IS-IS configuration. The `AllocBytes` field indicates the highest amount of memory used by the data structure. For example, the `isis_area_addr` data structure has 544 blocks of allocated memory, each block is allocated a value of 16 bytes, resulting in allocated bytes of 8704. The maximum allocated blocks and bytes are high-water marks for a data structure. For more information on displaying task-related information, see "Display Tasks" on page 185.

The `Total bytes in use` field shows the total amount of memory used by the routing protocol process (rpd).

Step 3: Display Tasks

Purpose You can display information about tasks to further your investigation of a memory problem on the router.

Action To display a list of tasks that are enabled on the router, enter the following JUNOS CLI operational mode commands:

```
user@host> show task
user@host> show task memory
user@host> show task task-name
```

Sample Output

```
user@R1> show task
Pri Task Name                               Pro  Port So  Flags
10 LMP Client                               17  <>
10 IF
15 INET6
15 INET
15 ISO
15 Memory
20 RPD Unix Domain Server./var/run/rpd_serv.local 21  <>
20 RPD Unix Domain Server./var/run/rpd_serv.local 20  <>
20 RPD Unix Domain Server./var/run/rpd_serv.local 19  <>
20 RPD Unix Domain Server./var/run/rpd_server_communication 16  <Accept>
20 RPD Server.0.0.0.0+666                    666 15  <Accept>
```

```

20 Aggregate
20 RT
30 ICMP 1
30 Router-Advertisement
30 ICMPv6 58 9 <>
39 OSPFv2 I/O./var/run/ppmd_control 12 <>
40 12vpn global task
40 BGP RT Background <LowPrio>
40 BGP.::+179 179 23 <Accept LowPrio>
40 BGP.0.0.0+179 179 22 <Accept LowPrio>
40 BFD I/O./var/run/bfdd_control 11 <>
40 OSPF 89
50 BGP_65001.10.0.0.5+3531 3531 18 <LowPrio>
50 BGP_65002.10.1.12.2+1224 1224 25 <LowPrio>
50 BGP_Group_internal <LowPrio>
50 BGP_Group_toR2 <LowPrio>
50 TED
50 ASPaths
51 Resolve inet.0 <LowPrio>
60 KStat 13 <>
60 KRT Request 7 <>
60 KRT Ifstate 255 6 <>
60 KRT 255 5 <>
60 Redirect
70 MGMT.local 24 <>
70 MGMT_listen./var/run/rpd_mgmt 14 <Accept>
70 SNMP Subagent./var/run/snmpd_stream 10 <>
80 IF Delete

```

user@R1> show task memory

Memory	Size (kB)	%Available	When
Currently In Use:	3490	1%	now
Maximum Ever Used:	3535	1%	04/02/04 11:54:46
Available:	220623	100%	now

user@R1> show task io

Task Name	Reads	Writes	Rcvd	Sent	Dropped
LMP Client	1	1	0	0	0
IF	0	0	0	0	0
INET6	0	0	0	0	0
INET	0	0	0	0	0
ISO	0	0	0	0	0
Memory	0	0	0	0	0
RPD Unix Domain Server./var/ru	1	0	0	0	0
RPD Unix Domain Server./var/ru	1	0	0	0	0
RPD Unix Domain Server./var/ru	0	0	0	0	0
RPD Unix Domain Server./var/ru	3	0	0	0	0
RPD Server.0.0.0.0+666	0	0	0	0	0
Aggregate	0	0	0	0	0
RT	0	0	0	0	0
ICMP	0	0	0	0	0
Router-Advertisement	0	0	0	0	0
ICMPv6	0	0	0	0	0
OSPFv2 I/O./var/run/ppmd_contr	31167	1	0	0	0
12vpn global task	0	0	0	0	0
BGP RT Background	0	0	0	0	0
BGP.::+179	0	0	0	0	0
BGP.0.0.0.0+179	8	0	0	0	0
BFD I/O./var/run/bfdd_control	30731	1	0	0	0
OSPF	0	0	0	0	0
BGP_65001.10.0.0.5+3531	20486	0	0	0	0
BGP_65002.10.1.12.2+1224	20489	6	0	0	0
BGP_Group_internal	0	0	0	0	0

BGP_Group_toR2	0	0	0	0	0
TED	0	0	0	0	0
ASPaths	0	0	0	0	0
Resolve inet.0	0	0	0	0	0
KStat	0	0	0	0	0
KRT Request	0	0	57	0	0
KRT Ifstate	18	0	16	0	0
KRT	0	0	2	0	0
Redirect	0	0	0	0	0
MGMT.local	0	0	0	0	0
MGMT_Listen./var/run/rpd_mgmt	23	0	0	0	0
SNMP Subagent./var/run/snmpd_s	23	0	0	0	0
IF Delete	0	0	0	0	0

What It Means The sample output shows a list of routing, routing protocol, and interface tasks that are currently running on the router (**show task**), a summary of memory utilization (**show task memory**), and the memory utilization of a particular task (**show task io**). Tasks can be baseline tasks performed regardless of the router configuration, and other tasks that depend on the router configuration. For example, the **BGP_Group_internal** task is the result of the configuration of BGP on the router, while the **INET6** task is a base task associated with the routing process (rpd).

Each task in the **show task** command output has a priority and a task name. For example, the current priority is 10 for **LMP Client** and 80 for **IF Delete**. A lower number indicates a higher priority.

Some tasks have flags attached to them. For example, the **BGP0.0.0.0+179** task has two flags, **Accept** and **LowPrio**. The **Accept** flag indicates that the task is waiting for incoming connections, and the **LowPrio** flag indicates that the task will be dispatched to read its socket after other, higher priority tasks. Two additional flags are **Connect**, which indicates that a task is waiting for a connection to complete, and **Delete**, which indicates that a task has been deleted and is being cleaned up.

The **show task io** command shows the statistics gathered for each IO operation. The counters show the following:

- **Reads**—This counter increments when a datagram arrives on a connected socket of the task and the task's read callback is called.
- **Writes**—This counter increments when a connected socket of a task becomes writable and the task's callback is called.
- **Rcvd**—This counter increments when the task calls the Routing Engine to read a datagram from a socket which may or may not be connected.
- **Sent**—This counter increments when a task attempts to read or write a datagram on an existing or nonexisting socket.
- **Drops**—This counter increments when a task attempts to read or write a datagram through the Routing Engine on a prebuilt socket, but the request fails for any reason.

Chapter 15

Verify Traffic and Packets through the Router

This chapter describes how to verify traffic and packets entering and passing through your Juniper Networks router. (See Table 36.)

Table 36: Checklist for Verifying Traffic and Packets through the Router

Verify Traffic and Packets Tasks	Command or Action
Monitor Traffic through the Router on page 190	
1. Display Real-Time Statistics about All Interfaces on the Router on page 190	monitor interface traffic
2. Display Real-Time Statistics about an Interface on page 191	monitor interface <i>interface-name</i>
Verify Packets on page 193	
1. Monitor Packets Sent from and Received by the Routing Engine on page 193	monitor traffic interface <i>interface-name</i>
2. Display Key IP Header Information on page 194	show firewall log
3. Show Packet Count When a Firewall Filter Is Configured with the Count Option on page 195	show firewall filter <i>filter-name</i>
4. Display Traffic from the Point of View of the Packet Forwarding Engine on page 196	show pfe statistics traffic

Monitor Traffic through the Router

Purpose To continue your diagnosis of a problem, display real-time statistics about the traffic passing through physical interfaces on the router.

Steps To Take To display real-time statistics about physical interfaces, follow these steps:

1. Display Real-Time Statistics about All Interfaces on the Router on page 190
2. Display Real-Time Statistics about an Interface on page 191

Step 1: Display Real-Time Statistics about All Interfaces on the Router

Action To display real-time statistics about traffic passing through all interfaces on the router, use the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> monitor interface traffic
```

Sample Output

```
user@host> monitor interface traffic
host name                               Seconds: 15                               Time: 12:31:09
```

Interface	Link	Input packets	(pps)	Output packets	(pps)
so-1/0/0	Down	0	(0)	0	(0)
so-1/1/0	Down	0	(0)	0	(0)
so-1/1/1	Down	0	(0)	0	(0)
so-1/1/2	Down	0	(0)	0	(0)
so-1/1/3	Down	0	(0)	0	(0)
t3-1/2/0	Down	0	(0)	0	(0)
t3-1/2/1	Down	0	(0)	0	(0)
t3-1/2/2	Down	0	(0)	0	(0)
t3-1/2/3	Down	0	(0)	0	(0)
so-2/0/0	Up	211035	(1)	36778	(0)
so-2/0/1	Up	192753	(1)	36782	(0)
so-2/0/2	Up	211020	(1)	36779	(0)
so-2/0/3	Up	211029	(1)	36776	(0)
so-2/1/0	Up	189378	(1)	36349	(0)
so-2/1/1	Down	0	(0)	18747	(0)
so-2/1/2	Down	0	(0)	16078	(0)
so-2/1/3	Up	0	(0)	80338	(0)
at-2/3/0	Up	0	(0)	0	(0)
at-2/3/1	Down	0	(0)	0	(0)

Bytes=b, Clear=c, Delta=d, Packets=p, Quit=q or ESC, Rate=r, Up=^U, Down=^D

What It Means The sample output displays traffic data for active interfaces and the amount that each field has changed since the command started or since the counters were cleared by using the **C** key. In this example, the **monitor interface** command has been running for 15 seconds since the command was issued or since the counters last returned to zero.

Step 2: Display Real-Time Statistics about an Interface

Action To display real-time statistics about traffic passing through an interface on the router, use the following JUNOS CLI operational mode command:

```
user@host> monitor interface interface-name
```

Sample Output user@R1> **monitor interface** so-0/0/1

```
Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'
R1
```

```
Interface: so-0/0/1, Enabled, Link is Up
Encapsulation: PPP, Keepalives, Speed: OC3 Traffic statistics:
  Input bytes:          5856541 (88 bps)
  Output bytes:         6271468 (96 bps)
  Input packets:        157629 (0 pps)
  Output packets:       157024 (0 pps)
Encapsulation statistics:
  Input keepalives:     42353
  Output keepalives:    42320
  LCP state: Opened
Error statistics:
  Input errors:         0
  Input drops:          0
  Input framing errors: 0
  Input runts:          0
  Input giants:         0
  Policed discards:     0
  L3 incompletes:       0
  L2 channel errors:    0
  L2 mismatch timeouts: 0
  Carrier transitions:  1
  Output errors:        0
  Output drops:         0
  Aged packets:         0
Active alarms : None
Active defects: None
SONET error counts/seconds:
  LOS count             1
  LOF count             1
  SEF count             1
  ES-S                  77
  SES-S                 77
SONET statistics:
  BIP-B1                0
  BIP-B2                0
  REI-L                 0
  BIP-B3                0
  REI-P                 0
Received SONET overhead: F1      : 0x00 J0      : 0xZ
```

What It Means The sample output shows the input and output packets for a particular SONET interface (so-0/0/1). The information can include common interface failures, such as SONET/SDH and T3 alarms, loopbacks detected, and increases in framing errors. For more information, see “Track Error Conditions” on page 273.

To control the output of the command while it is running, use the keys shown in Table 37.

Table 37: Monitor Interface Output Control Keys

Action	Key
Display information about the next interface. The monitor interface command scrolls through the physical or logical interfaces in the same order that they are displayed by the show interfaces terse command.	N
Display information about a different interface. The command prompts you for the name of a specific interface.	I
Freeze the display, halting the display of updated statistics.	F
Thaw the display, resuming the display of updated statistics.	T
Clear (zero) the current delta counters since monitor interface was started. It does not clear the accumulative counter.	C
Stop the monitor interface command.	Q

See the *JUNOS System Basics and Services Command Reference* for details on using match conditions with the **monitor traffic** command.

Verify Packets

Purpose You can check the flow of packets to and from the router to further your investigation of issues on the router.

Steps To Take To verify packets, follow these steps:

1. Monitor Packets Sent from and Received by the Routing Engine on page 193
2. Display Key IP Header Information on page 194
3. Show Packet Count When a Firewall Filter Is Configured with the Count Option on page 195
4. Display Traffic from the Point of View of the Packet Forwarding Engine on page 196

Step 1: Monitor Packets Sent from and Received by the Routing Engine

Action To print packet headers transmitted through network interfaces sent from or received by the Routing Engine, enter the following JUNOS CLI operational mode command:

```
user@host> monitor traffic interface interface-name
```

Sample Output

```
user@R1> monitor traffic interface so-0/0/1
verbose output suppressed, use <detail> or <extensive> for full protocol decode
Listening on so-0/0/1, capture size 96 bytes

11:23:01.666720 In IP 10.1.15.2 > OSPF-ALL.MCAST.NET: OSPFv2 Hello length: 48
11:23:01.666884 Out IP 10.1.15.1 > OSPF-ALL.MCAST.NET: OSPFv2 Hello length: 48
11:23:01.681330 Out IP 10.0.0.1.bgp > 10.0.0.5.3813: P 3821434885:3821434904(19)
ack 165811073 win 16417 <nop,nop,timestamp 42120056 42108995>: BGP, length: 19
11:23:01.682041 In IP 10.0.0.5.3813 > 10.0.0.1.bgp: P 1:20(19) ack 19 win 16398
<nop,nop,timestamp 42111985 42120056>: BGP, length: 19
11:23:01.781132 Out IP 10.0.0.1.bgp > 10.0.0.5.3813: . ack 20 win 16398
<nop,nop,timestamp 42120066 42111985>
11:23:03.996629 In LCP echo request (type 0x09 id 0x67 len 0x0008)
11:23:03.996645 Out LCP echo reply (type 0x0a id 0x67 len 0x0008)
11:23:04.801130 Out LCP echo request (type 0x09 id 0x6d len 0x0008)
11:23:04.801694 In LCP echo reply (type 0x0a id 0x6d len 0x0008)
^C
11 packets received by filter
0 packets dropped by kernel
```

What It Means The sample output shows the actual packets entering and leaving the Routing Engine, not the transit packets passing through the router. You can use this information to diagnose issues such as Point-to-Point Protocol negotiation, Border Gateway Protocol negotiation, and Open Shortest Path First hellos.

The `monitor traffic` command is similar to the UNIX `tcpdump` command. For more information about the `monitor traffic` command, see the *JUNOS System Basics and Services Command Reference*.



CAUTION: Use the `monitor traffic` command to diagnose problems on your router. Do not to leave this command on because it consumes Routing Engine resources.

Step 2: Display Key IP Header Information

Action To display key IP header information if you have a firewall configured with a `log` action, enter the following JUNOS CLI operational mode command:

```
user@host> show firewall log
```

Sample Output

```
user@R1> show firewall log
```

Time	Filter	A Interface	Pro	Source address	Destination address
16:08:04	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:24373
16:08:03	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:29531
16:08:02	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:27265
16:08:01	pfe	A so-1/1/0.0	OSP	123.168.10.65	212.0.0.5:48
16:08:01	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:43943
16:08:00	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:58572
16:07:59	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:56307
16:07:58	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:60185
16:07:57	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:1600
16:07:56	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:6502
16:07:55	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:17548
16:07:54	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:5298
16:07:53	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:24536
16:07:52	sample-test	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:24373
16:07:52	sample-test	A local	ICM	123.168.10.66	123.168.10.65:22325
16:07:52	pfe	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:27900
16:07:51	pfe	A so-1/1/0.0	OSP	123.168.10.65	212.0.0.5:48
16:07:51	sample-test	A so-1/1/0.0	ICM	123.168.10.65	123.168.10.66:29531
16:07:51	sample-test	A local	ICM	123.168.10.66	123.168.10.65:27483

What It Means The sample output shows key IP header information about firewall filters on the router. The source and destination addresses of packets provide important information when you investigate problems on the router.

The **Filter** field contains information about how a packet traveled through the router before it was handled by either the Routing Engine or the Packet Forwarding Engine.

- If the filter name appears in the **Filter** field, the Routing Engine handled the packet. For example, `sample-test` is a firewall filter configured at the `[edit firewall]` hierarchy level.

- If the word **pfe** appears in the **Filter** field, the Packet Forwarding Engine handled the packet. The Packet Forwarding Engine receives information about the name of the firewall filter.

All packets were accepted (A). Other actions are discard (D) and reject (R).

The **Interface** column shows that all packets came through **so-1/1/0.0**, and **icmp** or **osp** are the represented protocols. Other possible protocol names are: **egp**, **gre**, **ipip**, **pim**, **resp**, **tcp**, or **udp**.

Step 3: Show Packet Count When a Firewall Filter Is Configured with the Count Option

Action To show the packet count when a firewall filter is configured with the count option, enter the following JUNOS CLI operational mode command:

```
user@host> show firewall filter filter-name
```

Sample Output 1 The following sample output shows the **icmp** filter incrementing:

```
user@R1> show firewall filter icmp
Filter: icmp
Counters:
Name                                     Bytes      Packets
count-icmp                             252         3
```

Sample Output 2 The following sample output shows a configuration of the count option:

```
[edit]
user@R1# show firewall filter icmp
term a {
    from {
        protocol icmp;
    }
    then count count-icmp;
}
term b {
    then accept;
}
```

What It Means The sample output shows that the packet matched a criteria in the **icmp** filter and the filter had a count action applied to it.

Step 4: Display Traffic from the Point of View of the Packet Forwarding Engine

Action To display traffic from the point of view of the Packet Forwarding Engine, enter the following JUNOS CLI operational mode command:

```
user@host> show pfe statistics traffic
```

Sample Output 1 The following sample output was taken before packets were sent:

```
user@R2> show pfe statistics traffic
PFE Traffic statistics:
    635392 packets input  (0 packets/sec)
    829862 packets output (0 packets/sec)

PFE Local Traffic statistics:
    579278 local packets input
    773747 local packets output
    0 software input high drops
    0 software input medium drops
    0 software input low drops
    1 software output drops
    0 hardware input drops

PFE Local Protocol statistics:
    0 hdlc keepalives
    0 atm oam
    0 fr lmi
    254613 ppp lcp/ncp
    0 ospf hello
    0 rsvp hello
    107203 isis iih

PFE Hardware Discard statistics:
    0 timeout
    0 truncated key
    0 bits to test
    0 data error
    0 stack underflow
    0 stack overflow
    0 normal discard
    0 extended discard
    0 invalid iif
    0 info cell drops
    0 fabric drops
```

Sample Output 2 The following sample output was taken after 100 packets were sent to router R2:

```
user@R2> show pfe statistics traffic
PFE Traffic statistics:
    635595 packets input  (2 packets/sec)
    829990 packets output (2 packets/sec)

PFE Local Traffic statistics:
    579373 local packets input
    773869 local packets output
    0 software input high drops
    0 software input medium drops
    0 software input low drops
    1 software output drops
    0 hardware input drops
```

PFE Local Protocol statistics:

```

0 hdlc keepalives
0 atm oam
0 fr lmi
254655 ppp lcp/ncp
0 ospf hello
0 rsvp hello
107220 isis iih

```

PFE Hardware Discard statistics:

```

0 timeout
0 truncated key
0 bits to test
0 data error
0 stack underflow
0 stack overflow
100 normal discard
0 extended discard
0 invalid iif
0 info cell drops
0 fabric drops

```

What It Means The sample output shows the number and rate of packets entering and leaving the Packet Forwarding Engine. For example, the 100 packets sent to R2 were discarded due to a route that had a discard next hop configured, as shown in the **PFE Hardware Discard statistics** field. All counters increased as a result of the 100 packets.

Chapter 16

Use the ping and traceroute Commands

This chapter describes how to use the `ping` command to check the availability of various routers in a network topology, and how to use the `traceroute` command to check the path that packets travel between routers. (See Table 38.)

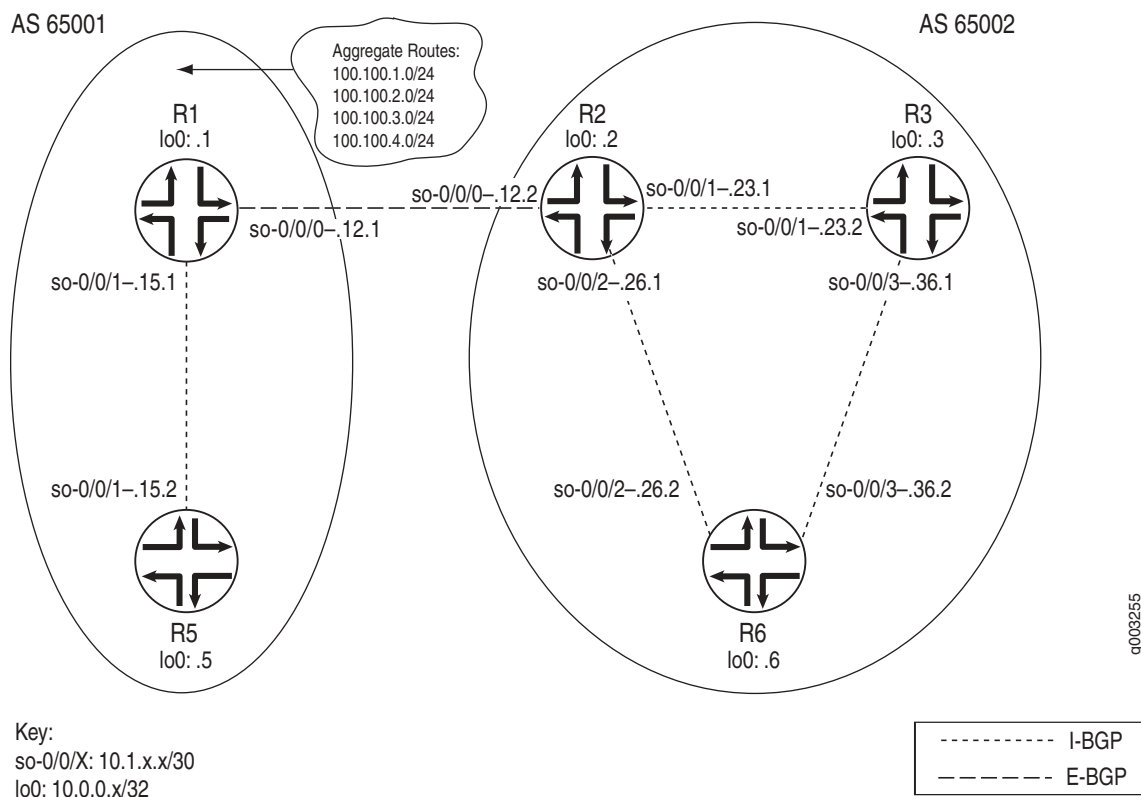
Table 38: Checklist for Using the ping and traceroute Commands

ping and traceroute Command Tasks	Command or Action
Check the Accessibility of Two Routers on the Edge on page 200	
Use Loopback Addresses on page 201	<code>ping remote-host count requests</code> <code>traceroute remote-host</code>
Use Interface Addresses on page 202	<code>ping interface-address count requests</code> <code>traceroute interface-address</code>
Examples of Unsuccessful ping and traceroute Commands on page 203	

Check the Accessibility of Two Routers on the Edge

Purpose This section provides examples of how to use the **ping** command to check the reachability of various routers in a network topology, and how to use the **traceroute** command to check the path that packets travel between routers. The topology shown in Figure 17 illustrates these commands.

Figure 17: Topology for ping and traceroute Command Examples



The network in Figure 17 consists of two autonomous systems (ASs). AS 65001 includes two routers, and AS 65002 includes three routers. The border router (R1) in AS 65001 announces aggregated prefixes **100.100/24** to the AS 65002 network.

Steps To Take To check the reachability of routers and the path to the routers, follow these steps:

1. Use Loopback Addresses on page 201
2. Use Interface Addresses on page 202

Step 1: Use Loopback Addresses

Purpose You can ping one router from another router by specifying the other router's loopback address as the IP address in the **ping** and **traceroute** commands. In this step, R6 and R5 both ping and traceroute each other.

Action To ping and traceroute between R5 and R6, enter the following JUNOS command-line interface (CLI) operational mode commands:

```
user@host> ping remote-host count requests
user@host> traceroute remote-host
```

Sample Output The following sample output is from R6 to R5, as shown in the network topology in Figure 17:

```
user@R6> ping 10.0.0.5 count 3
PING 10.0.0.6 (10.0.0.6): 56 data bytes
64 bytes from 10.0.0.6: icmp_seq=0 ttl=255 time=0.298 ms
64 bytes from 10.0.0.6: icmp_seq=1 ttl=255 time=0.237 ms
64 bytes from 10.0.0.6: icmp_seq=2 ttl=255 time=0.273 ms

--- 10.0.0.6 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.237/0.269/0.298/0.025 ms

user@R6> traceroute 10.0.0.5
traceroute to 10.0.0.5 (10.0.0.5), 30 hops max, 40 byte packets
 1  10.1.26.1 (10.1.26.1)  0.626 ms  0.530 ms  0.489 ms
 2  10.1.12.1 (10.1.12.1)  0.546 ms  0.534 ms  0.507 ms
 3  10.0.0.5 (10.0.0.5)  0.749 ms  0.694 ms  0.686 ms

user@R5> ping 10.0.0.6 count 3
PING 10.0.0.6 (10.0.0.6): 56 data bytes
64 bytes from 10.0.0.6: icmp_seq=0 ttl=253 time=0.875 ms
64 bytes from 10.0.0.6: icmp_seq=1 ttl=253 time=0.815 ms
64 bytes from 10.0.0.6: icmp_seq=2 ttl=253 time=0.819 ms

--- 10.0.0.6 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.815/0.836/0.875/0.027 ms

user@R5> traceroute 10.0.0.6
traceroute to 10.0.0.6 (10.0.0.6), 30 hops max, 40 byte packets
 1  10.1.15.1 (10.1.15.1)  0.635 ms  39.951 ms  0.526 ms
 2  10.1.12.2 (10.1.12.2)  0.555 ms  0.535 ms  0.515 ms
 3  10.0.0.6 (10.0.0.6)  0.769 ms  0.720 ms  0.674 ms
```

What It Means The sample output shows a successful ping and traceroute between the R6 and R5 loopback (lo0) addresses. The ping is successful because the loopback addresses of both routers are advertised to their directly connected neighbors.

The output for the **traceroute** command shows the path from R6 to R5, which is through R2.



NOTE: A ping command might lose packets due to rate limiting of Internet Message Control Protocol (ICMP) packets on the specified host.

Step 2: Use Interface Addresses

Purpose You can ping interfaces on remote routers.

Action To ping and traceroute between R5 and R6, enter the following JUNOS CLI operational mode commands:

```
user@host> ping interface-address count requests
user@host> traceroute interface-address
```

Sample Output

```
user@R6> ping 10.1.15.2 count 3
PING 10.1.15.2 (10.1.15.2): 56 data bytes
64 bytes from 10.1.15.2: icmp_seq=0 ttl=253 time=2.738 ms
64 bytes from 10.1.15.2: icmp_seq=1 ttl=253 time=0.858 ms
64 bytes from 10.1.15.2: icmp_seq=2 ttl=253 time=0.849 ms

--- 10.1.15.2 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.849/1.482/2.738/0.888 ms

user@R6> traceroute 10.1.15.2
traceroute to 10.1.15.2 (10.1.15.2), 30 hops max, 40 byte packets
 1  10.1.26.1 (10.1.26.1)  0.617 ms  0.534 ms  0.500 ms
 2  10.1.12.1 (10.1.12.1)  3.500 ms  0.543 ms  0.508 ms
 3  10.1.15.2 (10.1.15.2)  0.699 ms  0.700 ms  0.672 ms

user@R5> ping 10.1.36.2 count 3
PING 10.1.36.2 (10.1.36.2): 56 data bytes
64 bytes from 10.1.36.2: icmp_seq=0 ttl=253 time=0.890 ms
64 bytes from 10.1.36.2: icmp_seq=1 ttl=253 time=0.857 ms
64 bytes from 10.1.36.2: icmp_seq=2 ttl=253 time=3.264 ms

--- 10.1.36.2 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.857/1.670/3.264/1.127 ms

user@R5> traceroute 10.1.36.2
traceroute to 10.1.36.2 (10.1.36.2), 30 hops max, 40 byte packets
 1  10.1.15.1 (10.1.15.1)  0.636 ms  7.979 ms  0.497 ms
 2  10.1.12.2 (10.1.12.2)  0.544 ms  0.547 ms  0.512 ms
 3  10.1.36.2 (10.1.36.2)  0.729 ms  0.696 ms  0.672 ms
```

What It Means The sample output shows a successful ping and traceroute between the interfaces on R6 and R5. The ping is successful because the interface addresses of both routers are advertised to their directly connected neighbors.

The output for the `traceroute` command shows the path from R6 to R5, which is through R2.



NOTE: A ping command might lose packets due to rate limiting of ICMP packets on the specified host.

Examples of Unsuccessful ping and traceroute Commands

Purpose When the ping or traceroute commands are unsuccessful, it is useful to understand the output.

Action To ping and traceroute between R5 and R6, enter the following JUNOS CLI operational mode commands:

```
user@host> ping interface-address count requests
user@host> traceroute interface-address
```

Sample Output 1

```
user@R6> ping 10.1.15.2 count 3
PING 10.1.15.2 (10.1.15.2): 56 data bytes
36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS Len ID Flg off TTL Pro cks Src Dst
 4 5 00 0054 3648 0 0000 01 01 465c 10.1.26.2 10.1.15.2

36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS Len ID Flg off TTL Pro cks Src Dst
 4 5 00 0054 364b 0 0000 01 01 4659 10.1.26.2 10.1.15.2

36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS Len ID Flg off TTL Pro cks Src Dst
 4 5 00 0054 364f 0 0000 01 01 4655 10.1.26.2 10.1.15.2

^C
--- 10.1.15.2 ping statistics ---
3 packets transmitted, 0 packets received, 100% packet loss

user@R6> ping 10.0.0.5 count 3
PING 10.0.0.5 (10.0.0.5): 56 data bytes
ping: sendto: No route to host
ping: sendto: No route to host
ping: sendto: No route to host
^C
--- 10.0.0.5 ping statistics ---
3 packets transmitted, 0 packets received, 100% packet loss

user@R6> ping 10.1.15.2
PING 10.1.15.2 (10.1.15.2): 56 data bytes
^C
--- 10.1.15.2 ping statistics ---
4 packets transmitted, 0 packets received, 100% packet loss
```

Sample Output 2

```
user@R6> traceroute 10.1.15.2
traceroute to 10.1.15.2 (10.1.15.2), 30 hops max, 40 byte packets
 1 10.1.26.1 (10.1.26.1) 0.626 ms 0.526 ms 0.494 ms
 2 10.1.26.2 (10.1.26.2) 0.521 ms 0.529 ms 0.509 ms
 3 10.1.26.1 (10.1.26.1) 0.516 ms 0.536 ms 0.523 ms
 4 10.1.26.2 (10.1.26.2) 0.528 ms 0.547 ms 0.524 ms
 5 10.1.26.1 (10.1.26.1) 0.532 ms 0.549 ms 0.535 ms
 6 10.1.26.2 (10.1.26.2) 0.547 ms 0.566 ms 0.543 ms
 7 10.1.26.1 (10.1.26.1) 0.551 ms 0.569 ms 0.538 ms
 8 10.1.26.2 (10.1.26.2) 0.557 ms 0.580 ms 0.567 ms
 9 10.1.26.1 (10.1.26.1) 0.570 ms 0.598 ms 0.570 ms
```

```

user@R6> traceroute 10.1.15.2
traceroute to 10.1.15.2 (10.1.15.2), 30 hops max, 40 byte packets
 1  10.1.36.1 (10.1.36.1)  0.651 ms  7.834 ms  0.506 ms
 2  10.1.23.1 (10.1.23.1)  0.536 ms  0.538 ms  0.504 ms
 3  * * *
 4  * * *
 5  *^C

```

What It Means Sample output 1 shows three instances of the `ping` command not succeeding. In the first instance, the packets exceed the time-to-live value, which is decremented to `1`, indicating that packets are being rejected possibly because of a loop. In the second instance, the local router does not know the route to the host. In the third instance, there is no route to the IP address, which might be due to packets being lost on a remote router.

Sample output 2 shows two instances of the `traceroute` command not succeeding. In the first instance, there is a loop between shared interfaces on `R6` and `R2`, as indicated by the `10.1.26.1` and `10.1.26.2` appearing repeatedly. In the second instance, the path goes through `R3` (`10.1.36.1`) to `R2` (`10.1.23.1`) when it times out, as indicated by the asterisk (*). The timeout might be due to the absence of a route to the remote interface.

Chapter 17

Use MIBs

This chapter describes how to determine which Management Information Bases (MIBs) are supported by a Juniper Networks release, and how to query enterprise-specific and standard MIBs to retrieve management information for the various hardware and software components of a Juniper Networks router. (See Table 39.)

Table 39: Checklist for Using MIBs

Use MIBs Tasks	Command or Action
Determine Which MIBs Are Supported by a Juniper Release on page 206	http://www.juniper.net/techpubs/software/index.html
Run Snmpwalk from an NMS System to a Juniper Router on page 207	<code>snmpwalk [common arguments] hostname community object-id</code>
Use SNMP Trace Operations to Monitor a Router on page 209	
1. Configure Trace Operations for SNMP on page 209	<code>[edit]</code> <code>edit snmp</code> <code>set traceoptions flag pdu</code> <code>commit and-quit</code>
2. Query a MIB With SNMPGet on page 210	<code>snmpget hostname community oid</code>
3. Display the Output for SNMP Trace Operations on page 211	<code>show log snmp</code>
Monitor Memory Usage on a Router on page 212	
1. Check Memory Utilization on Chassis Components on page 212	<code>snmpwalk [common arguments] hostname community object-id</code> <code>show chassis routing-engine</code>
2. Check Memory Utilization per Process on page 215	<code>snmpwalk [common arguments] hostname community object-id</code>
Monitor CPU Utilization on page 218	
1. Check CPU Utilization on page 218	<code>snmpwalk [common arguments] hostname community object-id</code>
2. Check CPU Utilization per Process on page 220	<code>snmpwalk [common arguments] hostname community object-id</code>
Retrieve Version Information about Router Software Components on page 223	<code>snmpwalk [common arguments] hostname community object-id</code>

Determine Which MIBs Are Supported by a Juniper Release

Purpose When you update the router software, you might also want to update the corresponding MIBs. Links to the MIBs related to a release are located in the *JUNOS Internet Software Network Management Configuration Guide*. This guide lists the Juniper-specific enterprise MIBs, and provides a link to Simple Network Management Protocol (SNMP) standards that list the standard MIBs supported by the JUNOS software.

In addition, a tar file that contains all the Juniper Networks enterprise-specific MIBs is included in the JUNOS software package for each release.

Action To determine MIBs supported by a Juniper release, follow these steps;

1. Enter the following URL into the address line of your browser:

`http://www.juniper.net/techpubs/software/index.html`
2. Select the release you are interested in.
3. From **JUNOS Configuration Guides**, select **Network Management**.
4. From the table of contents, select **JUNOS Networks Enterprise-Specific MIBs**.
5. From the table of contents, select **SNMP Overview**.
6. From **SNMP Overview**, select **SNMP Standards**.

Run Snmpwalk from an NMS System to a Juniper Router

Purpose Snmpwalk is an SNMP application that you can use to query a MIB for information about the functioning of a router in your network. Snmpwalk uses **GetNext** requests to retrieve the specified information. Object identifiers (OIDs) are used to query the MIB. If the OID argument is not present, Snmpwalk searches MIB-2.

Action To run Snmpwalk for a specific OID, from a management station that has access to the router, and using a tool such as Snmpwalk, enter the following command:

```
user-nms# snmpwalk [common arguments] hostname community object-id
```

Sample Output

```
user-nms % snmpwalk -Os -M /volume/~mibs -m all tp1 public .1.3.6.1.2.1.4
ipForwarding.0 = forwarding(1)
ipDefaultTTL.0 = 64
ipInReceives.0 = Counter32: 9262713
ipInHdrErrors.0 = Counter32: 0
ipInAddrErrors.0 = Counter32: 0
ipForwDatagrams.0 = Counter32: 614171
ipInUnknownProtos.0 = Counter32: 0
ipInDiscards.0 = Counter32: 0
ipInDelivers.0 = Counter32: 8648408
ipOutRequests.0 = Counter32: 1226483
ipOutDiscards.0 = Counter32: 0
ipOutNoRoutes.0 = Counter32: 0
ipReasmTimeout.0 = 60
ipReasmReqds.0 = Counter32: 0
ipReasmOKs.0 = Counter32: 0
ipReasmFails.0 = Counter32: 0
ipFragOKs.0 = Counter32: 0
ipFragFails.0 = Counter32: 0
ipFragCreates.0 = Counter32: 0
ipAdEntAddr.10.0.0.1 = IpAddress: 10.0.0.1
ipAdEntAddr.10.1.12.1 = IpAddress: 10.1.12.1
ipAdEntAddr.10.1.15.1 = IpAddress: 10.1.15.1
ipAdEntAddr.10.168.70.143 = IpAddress: 10.168.70.143
[...Output truncated...]
```

What It Means The sample output shows that the user is on a network management station (user-nms %) that has access to the router, tp1. In the command, the following options are used:

- **Os**—Deletes all but the last symbolic part of the OID **sysUpTime.0**. For example, **Timeticks: (14096763) 1 day, 15:09:27.63**.
- **-M**—Compiles the MIB and gives a path or location to the MIBs.
- **-m**—Uses the files in the directory pointed to by the **-M** option.
- **all**—Uses all the files in the directory pointed to by the **-M** option.

In addition, the command includes the hostname **tp1**, the community string **public**, and the OID **.1.3.6.1.2.1.4**.

The OID in this example is from RFC 2096, *IP Forwarding Table MIB*, which displays multipath IP routes that have the same network number but different network masks.

Before you can retrieve SNMP information from a router, you must have the minimum SNMP configuration for that router. Following is the minimum SNMP configuration required:

```
[edit]
snmp {
  community public {
    authorization read-only;
  }
}
```

With this configuration, the system responds to SNMP **Get**, **GetNext**, and **GetBulk** commands that contain the community string **public**.

For more detailed information on configuring SNMP on a router, see the *JUNOS Network Management Configuration Guide*.

Use SNMP Trace Operations to Monitor a Router

Purpose Tracing operations record more detailed messages about the operation of SNMP, such as the various types of routing protocol packets sent and received, and routing policy actions. In this section, traceoptions are configured on a router, a MIB object is queried through a network management station, and the action of the query is verified with a log file on the router.



NOTE: Traceoptions, in general, requires extra router resources. It is recommended that you do not leave it on permanently.

Steps To Take To use SNMP traceoptions to monitor a router, follow these steps:

1. Configure Trace Operations for SNMP on page 209
2. Query a MIB With SNMPGet on page 210
3. Display the Output for SNMP Trace Operations on page 211

Step 1: Configure Trace Operations for SNMP

Purpose Define tracing for SNMP to access more granular information about the packets sent and received through SNMP.

Action To configure SNMP tracing operations, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@R1# edit snmp
```

2. Configure trace operations:

```
[edit snmp]
user@R1# set traceoptions flag pdu
```

3. Commit the configuration:

```
user@R1# commit and-quit
commit complete
Exiting configuration mode
```

Sample Output

```
user@R1> show configuration snmp
view all {
    oid .1 include;
}
view system {
    oid system;
}
community public {
    view all;
    authorization read-only;
}
```

```
community private {
    view system;
    authorization read-write;
}
traceoptions {
    flag pdu;
}
```

What It Means The sample output shows a configuration for SNMP that includes traceoptions. The pdu flag is configured, which results in the generation of SNMP request and response packets. The output for the tracing operation is placed into various log files in the /var/log directory.

Protocol-specific tracing operations override any equivalent operations that you specify in the global traceoptions statement. If there are no equivalent operations, they supplement the global tracing options. If you do not specify any protocol-specific tracing, the routing protocol inherits all the global tracing operations.

Step 2: Query a MIB With SNMPGet

Purpose Send an SNMP request to check that the SNMP configuration is correct.

Action To query a MIB with SNMPGet, enter the following command:

```
user@nms % snmpget hostname community oid
```

Sample Output

```
user-nms % snmpget tp1 public .1.3.6.1.2.1.1.1.0
system.sysDescr.0 = m7i internet router, kernel 6.0R1.5

user-nms % snmpget tp1 public sysDescr.0
system.sysDescr.0 = m7i internet router, kernel 6.0R1.5
```

What It Means The sample output shows a query from a network management station (nms) for the description of the system running on the router tp1. The OID is entered in numerical form in the command line, and a description (sysDescr.0) is obtained in the output. You can also use sysDescr.0 in the command line to obtain the same output.

Monitor Memory Usage on a Router

Purpose From a management station that has access to the router, you can monitor memory usage of components, applications, and associated elements that have run or are currently running on a router.

Steps To Take From a management station that has access to the router and using a tool, such as Snmpwalk, follow these steps:

1. Check Memory Utilization on Chassis Components on page 212
2. Check Memory Utilization per Process on page 215

Step 1: Check Memory Utilization on Chassis Components

Purpose The enterprise-specific chassis MIB provides information about the router and its components. Within the chassis MIB, the `jnxMIBs` branch contains one main subbranch, `jnxBoxAnatomy`, which in turn contains a section, `jnxOperatingTable`. Within `jnxOperatingTable`, you can use the `jnxOperatingBuffer` object to monitor memory usage on your router. (See Figure 18.)

Figure 18: Chassis MIB Tree

```

+--juniperMIB(2636)¶
|   +--jnxProducts(1)¶
|   +--jnxServices(2)¶
|   +--jnxMibs(3)¶
|   |   +--jnxBoxAnatomy(1)¶
|   |   |   +--jnxContainersTable(6)¶
|   |   |   +--jnxContentsTable(8)¶
|   |   |   +--jnxLEDTable(10)¶
|   |   |   +--jnxFilledTable(12)¶
|   |   |   +--jnxOperatingTable(13)¶
|   |   |   |   +--jnxOperatingEntry(1)¶
|   |   |   |   |   +-- jnxOperatingContentsIndex(1)¶
|   |   |   |   |   +-- jnxOperatingL1Index(2)¶
|   |   |   |   |   +-- jnxOperatingL2Index(3)¶
|   |   |   |   |   +-- jnxOperatingL3Index(4)¶
|   |   |   |   |   +-- jnxOperatingDescr(5)¶
|   |   |   |   |   +-- jnxOperatingState(6)¶
|   |   |   |   |   +-- jnxOperatingTemp(7)¶
|   |   |   |   |   +-- jnxOperatingCPU(8)¶
|   |   |   |   |   +-- jnxOperatingISR(9)¶
|   |   |   |   |   +-- jnxOperatingDRAMSize(10)¶
|   |   |   |   |   +-- jnxOperatingBuffer(11)¶
|   |   |   |   |   +-- jnxOperatingHeap(12)¶
|   |   |   |   |   +-- jnxOperatingUpTime(13)¶
|   |   |   |   |   +-- jnxOperatingLastRestart(14)¶
|   |   |   |   |   +-- jnxOperatingMemory(15)¶
|   |   |   |   |   +-- jnxOperatingStateOrdered(16)¶
|   |   |   +--jnxRedundancyTable(14)¶
|   |   +--jnxFruTable(15)¶

```

After each object description is a value in parenthesis, such as (1). This value can be used to enter an OID for the specific object. For example, to gather information on memory utilization, you can type the object description (`jnxOperatingBuffer`) or the OID (`.1.3.6.1.4.1.2636.3.1.13.1.11`).

Action To check memory utilization using the Juniper enterprise chassis MIB, from a management station that has access to the router, and using a tool such as `Snmpwalk`, enter the following commands:

```
user-bsd# snmpwalk [common arguments] hostname community object-id
user@host> show chassis routing-engine
```

Sample Output

```
user-nms % snmpwalk -Os -M /volume/~mibs -m all tp1 public jnxOperatingBuffer
jnxOperatingBuffer.1.1.1.0 = Gauge32: 0
jnxOperatingBuffer.1.1.2.0 = Gauge32: 0
jnxOperatingBuffer.1.1.3.0 = Gauge32: 0
jnxOperatingBuffer.2.1.0.0 = Gauge32: 0
jnxOperatingBuffer.4.1.1.0 = Gauge32: 0
jnxOperatingBuffer.4.1.2.0 = Gauge32: 0
jnxOperatingBuffer.4.1.3.0 = Gauge32: 0
jnxOperatingBuffer.4.1.4.0 = Gauge32: 0
jnxOperatingBuffer.6.1.1.0 = Gauge32: 6
jnxOperatingBuffer.6.1.2.0 = Gauge32: 6
jnxOperatingBuffer.7.1.0.0 = Gauge32: 8
jnxOperatingBuffer.7.2.0.0 = Gauge32: 8
jnxOperatingBuffer.8.1.1.0 = Gauge32: 0
jnxOperatingBuffer.8.2.3.0 = Gauge32: 0
jnxOperatingBuffer.8.2.4.0 = Gauge32: 0
jnxOperatingBuffer.9.1.0.0 = Gauge32: 28
jnxOperatingBuffer.9.1.1.0 = Gauge32: 0

user-nms % snmpwalk -Os -M /volume/~mibs -m all tp1 public jnxOperatingDescr
jnxOperatingDescr.1.1.1.0 = midplane
jnxOperatingDescr.1.1.2.0 = midplane
jnxOperatingDescr.1.1.3.0 = midplane
jnxOperatingDescr.2.1.0.0 = Power Supply A
jnxOperatingDescr.4.1.1.0 = Left Tray front fan
jnxOperatingDescr.4.1.2.0 = Left Tray second fan
jnxOperatingDescr.4.1.3.0 = Left Tray third fan
jnxOperatingDescr.4.1.4.0 = Left Tray fourth fan
jnxOperatingDescr.6.1.1.0 = CFEB Internet Processor IIv1
jnxOperatingDescr.6.1.2.0 = CFEB Internet Processor IIv1
jnxOperatingDescr.7.1.0.0 = FPC @ 0/*/*
jnxOperatingDescr.7.2.0.0 = FPC @ 1/*/*
jnxOperatingDescr.8.1.1.0 = PIC: 4x OC-3 SONET, MM @ 0/0/*
jnxOperatingDescr.8.2.3.0 = PIC: 1x Tunnel @ 1/2/*
jnxOperatingDescr.8.2.4.0 = PIC: 1x G/E, 1000 BASE-SX @ 1/3/*
jnxOperatingDescr.9.1.0.0 = Routing Engine
jnxOperatingDescr.9.1.1.0 = Routing Engine PCMCIA Card
```

```

user@R1> show chassis routing-engine
Routing Engine status:
  Temperature                28 degrees C / 82 degrees F
  DRAM                       256 MB
  Memory utilization          28 percent
  CPU utilization:
    User                      0 percent
    Background                0 percent
    Kernel                    6 percent
    Interrupt                  0 percent
    Idle                       94 percent
  Model                       RE-5.0
  Serial ID                   1000431687
  Start time                   2003-11-20 11:42:04 PST
  Uptime                       63 days, 2 hours, 34 minutes, 4 seconds
  Load averages:              1 minute   5 minute   15 minute
                               0.01        0.02        0.01

```

What It Means The sample output shows the percentage of utilization for the FPC and Routing Engine. The first object, `jnxOperatingBuffer`, shows that the Routing Engine (9.1.0.0) has 28 percent memory utilization, the two CFEB processors are using 6 percent, and the FPCs have 8 percent memory utilization.

The second object, `jnxOperatingDescr`, provides a human readable description of the separate instances in the `jnxOperatingBuffer` object. For example, 1.1.0.0 represents the midplane, and 7.1.0.0 represents FPC @ 0/*/*.

The output for the `show chassis routing-engine` command shows similar information to that displayed in the output of the `jnxOperatingBuffer` object, with 28 percent memory utilization for the Routing Engine.

Step 2: Check Memory Utilization per Process

Purpose The standard System Application MIB (RFC 2287, *Definitions of System-Level Managed Objects for Applications*), describes a set of managed objects that are restricted to information that can be determined from the system itself. The object `sysAppElmtRunMemory` provides information about applications and associated elements that have run or are currently running on the host system. (See Figure 19.)

Figure 19: System Application MIB Tree

```

+--System: Application MIB
|
|   +--sysAppLOBJ
|   +--sysAppInstalled
|   +--sysAppRun
|   +--sysAppMap
|   +--sysAppNotifications
|   +--sysAppConformance
|       +--sysAppMIBCompliances
|       +--sysAppMIBGroups
|           +--sysAppRunGroup
|               +--sysAppRunStarted
|               +--sysAppRunCurrentState
|               +--sysAppRunPastRunStarted
|               +--sysAppRunPastRunExitState
|               +--sysAppRunPastRunTimeEnded
|               +--sysAppElmtRunInstallID
|               +--sysAppElmtRunTimeStarted
|               +--sysAppElmtRunState
|               +--sysAppElmtRunName
|               +--sysAppElmtRunParameters
|               +--sysAppElmtRunCPU
|               +--sysAppElmtRunMemory
|               +--sysAppElmtRunNumFiles
|               +--sysAppElmtRunUser
|
|   [...Output Truncated...]

```

Action To check memory utilization per process, from a management station that has access to the router, and using a tool such as Snmpwalk, enter the following command:

```
user-bsd# snmpwalk [common arguments] hostname community object-id
```

```

Sample Output use-nms % snmpwalk -Os -M /volume/~mibs -m all tp1 public sysAppElmtRunMemory
sysAppElmtRunMemory.0.0.0 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.2 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.3 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.4 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.5 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.6 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.7 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.8 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.9 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.10 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.11 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.12 = Gauge32: 0 Kbytes
sysAppElmtRunMemory.0.0.116 = Gauge32: 526164 Kbytes
sysAppElmtRunMemory.0.0.2023 = Gauge32: 416 Kbytes
sysAppElmtRunMemory.0.0.2131 = Gauge32: 1100 Kbytes
sysAppElmtRunMemory.0.0.2160 = Gauge32: 984 Kbytes
sysAppElmtRunMemory.0.0.2161 = Gauge32: 1100 Kbytes

```

```

sysApp1ElmtRunMemory.0.0.2174 = Gauge32: 996 Kbytes
sysApp1ElmtRunMemory.0.0.2324 = Gauge32: 0 Kbytes
sysApp1ElmtRunMemory.0.0.16781 = Gauge32: 1072 Kbytes
sysApp1ElmtRunMemory.0.0.18311 = Gauge32: 1284 Kbytes
sysApp1ElmtRunMemory.0.0.26827 = Gauge32: 1368 Kbytes
sysApp1ElmtRunMemory.3.1.1 = Gauge32: 4028 Kbytes
sysApp1ElmtRunMemory.3.2.2163 = Gauge32: 3196 Kbytes
sysApp1ElmtRunMemory.3.3.2185 = Gauge32: 1624 Kbytes
sysApp1ElmtRunMemory.3.4.2194 = Gauge32: 9768 Kbytes
sysApp1ElmtRunMemory.3.7.2168 = Gauge32: 2484 Kbytes
sysApp1ElmtRunMemory.3.9.2169 = Gauge32: 3004 Kbytes
sysApp1ElmtRunMemory.3.12.2172 = Gauge32: 2108 Kbytes
sysApp1ElmtRunMemory.3.13.2173 = Gauge32: 1888 Kbytes
sysApp1ElmtRunMemory.3.14.2164 = Gauge32: 1672 Kbytes
sysApp1ElmtRunMemory.3.15.2175 = Gauge32: 1644 Kbytes
sysApp1ElmtRunMemory.3.16.2165 = Gauge32: 1632 Kbytes
sysApp1ElmtRunMemory.3.17.2176 = Gauge32: 2716 Kbytes
sysApp1ElmtRunMemory.3.19.2177 = Gauge32: 1668 Kbytes
sysApp1ElmtRunMemory.3.20.2178 = Gauge32: 2160 Kbytes
sysApp1ElmtRunMemory.3.21.2179 = Gauge32: 2164 Kbytes
sysApp1ElmtRunMemory.3.23.2188 = Gauge32: 1688 Kbytes
sysApp1ElmtRunMemory.3.25.2186 = Gauge32: 1292 Kbytes
sysApp1ElmtRunMemory.3.26.2180 = Gauge32: 1676 Kbytes
sysApp1ElmtRunMemory.3.27.2181 = Gauge32: 2052 Kbytes
sysApp1ElmtRunMemory.3.30.2187 = Gauge32: 1236 Kbytes
sysApp1ElmtRunMemory.3.31.2184 = Gauge32: 1032 Kbytes
sysApp1ElmtRunMemory.3.34.2171 = Gauge32: 1156 Kbytes
sysApp1ElmtRunMemory.3.35.2047 = Gauge32: 1132 Kbytes
sysApp1ElmtRunMemory.3.36.2189 = Gauge32: 1836 Kbytes
sysApp1ElmtRunMemory.3.37.2191 = Gauge32: 1052 Kbytes
sysApp1ElmtRunMemory.5.5.7495 = Gauge32: 7628 Kbytes
sysApp1ElmtRunMemory.5.6.2167 = Gauge32: 11824 Kbytes
sysApp1ElmtRunMemory.5.6.26829 = Gauge32: 11880 Kbytes
sysApp1ElmtRunMemory.5.8.26828 = Gauge32: 7984 Kbytes
sysApp1ElmtRunMemory.5.28.2182 = Gauge32: 1468 Kbytes
sysApp1ElmtRunMemory.5.29.2183 = Gauge32: 1828 Kbytes

```

```

user-nms % snmpwalk -Os -M /volume/~mibs -m all tp1 public sysApp1ElmtRunName
sysApp1ElmtRunName.0.0.0 = (swapper)
sysApp1ElmtRunName.0.0.2 = (pagedaemon)
sysApp1ElmtRunName.0.0.3 = (vmdaemon)
sysApp1ElmtRunName.0.0.4 = (bufdaemon)
sysApp1ElmtRunName.0.0.5 = (syncer)
sysApp1ElmtRunName.0.0.6 = (netdaemon)
sysApp1ElmtRunName.0.0.7 = (if_pfe)
sysApp1ElmtRunName.0.0.8 = (if_pfe_listen)
sysApp1ElmtRunName.0.0.9 = (cb_poll)
sysApp1ElmtRunName.0.0.10 = (vmuncachedaemon)
sysApp1ElmtRunName.0.0.11 = (scs_housekeeping)
sysApp1ElmtRunName.0.0.12 = (if_pic_listen)
sysApp1ElmtRunName.0.0.116 = mfs
sysApp1ElmtRunName.0.0.2023 = pccardd
sysApp1ElmtRunName.0.0.2131 = cron
sysApp1ElmtRunName.0.0.2160 = /sbin/watchdog
sysApp1ElmtRunName.0.0.2161 = /usr/sbin/tetd
sysApp1ElmtRunName.0.0.2174 = /usr/sbin/tnp.snmpd
sysApp1ElmtRunName.0.0.2324 = (peer proxy)
sysApp1ElmtRunName.0.0.16781 = /usr/libexec/getty
sysApp1ElmtRunName.0.0.18311 = /usr/sbin/xntpd
sysApp1ElmtRunName.0.0.26827 = telnetd
sysApp1ElmtRunName.3.1.1 = /sbin/preinit
sysApp1ElmtRunName.3.2.2163 = /usr/sbin/chassisd
sysApp1ElmtRunName.3.3.2185 = /usr/sbin/dfwd

```



```

sysAppElmtRunName.3.4.2194 = /sbin/dcd
sysAppElmtRunName.3.7.2168 = /usr/sbin/snmpd
sysAppElmtRunName.3.9.2169 = /usr/sbin/mib2d
sysAppElmtRunName.3.12.2172 = /usr/sbin/apsd
sysAppElmtRunName.3.13.2173 = /usr/sbin/vrrpd
sysAppElmtRunName.3.14.2164 = /usr/sbin/alarmd
sysAppElmtRunName.3.15.2175 = /usr/sbin/pfed
sysAppElmtRunName.3.16.2165 = /usr/sbin/craftd
sysAppElmtRunName.3.17.2176 = /usr/sbin/sampled
sysAppElmtRunName.3.19.2177 = /usr/sbin/ilmid
sysAppElmtRunName.3.20.2178 = /usr/sbin/rmopd
sysAppElmtRunName.3.21.2179 = /usr/sbin/cosd
sysAppElmtRunName.3.23.2188 = /usr/sbin/fsad
sysAppElmtRunName.3.25.2186 = /usr/sbin/irsd
sysAppElmtRunName.3.26.2180 = /usr/sbin/nasd
sysAppElmtRunName.3.27.2181 = /usr/sbin/fud
sysAppElmtRunName.3.30.2187 = /usr/sbin/rtsdpd
sysAppElmtRunName.3.31.2184 = /usr/sbin/smartd
sysAppElmtRunName.3.34.2171 = /usr/sbin/inetd
sysAppElmtRunName.3.35.2047 = syslogd
sysAppElmtRunName.3.36.2189 = /usr/sbin/spd
sysAppElmtRunName.3.37.2191 = /usr/sbin/eccd
sysAppElmtRunName.5.5.7495 = /usr/sbin/rpd
sysAppElmtRunName.5.6.2167 = /usr/sbin/mgd
sysAppElmtRunName.5.6.26829 = mgd: (mgd) (user)/dev/tty0
sysAppElmtRunName.5.8.26828 = -cli
sysAppElmtRunName.5.28.2182 = /usr/sbin/ppmd
sysAppElmtRunName.5.29.2183 = /usr/sbin/lmpd

```

What It Means The sample output shows the total amount of real system memory, measured in kilobytes, currently allocated to the processes retrieved by the `sysAppElmtRunMemory` object.

The `sysAppElmtRunMemory` object shows granular, per-process information about memory usage. For example, the **sampled** process (3.17.2176) is using 2716 kilobytes of memory.

The `sysAppElmtRunName` object provides a description of the separate instances displayed in the `sysAppElmtRunMemory` object. For example, the **sampled** process is represented by the OID 3.17.2176.

Monitor CPU Utilization

Purpose You can monitor CPU utilization using the Juniper specific enterprise chassis MIB and the standard system application MIB (RFC 2287, *Definitions of System-Level Managed Objects for Applications*).

Steps To Take From a management station that has access to the router, and using a tool such as Snmpwalk, follow these steps:

1. Check CPU Utilization on page 218
2. Check CPU Utilization per Process on page 220

Step 1: Check CPU Utilization

Purpose The enterprise-specific chassis MIB provides information about the router and its components. Within the chassis MIB, the `jnxMIBs` branch contains one main subbranch, `jnxBoxAnatomy`, which in turn contains a section, `jnxOperatingTable`. Within `jnxOperatingTable`, and under the `jnxOperatingEntry`, you can use the `jnxOperatingCPU` object to monitor the CPU on your router. (See Figure 20.)

Figure 20: Chassis MIB Tree

```

+--juniperMIB(2636)¶
|   +--jnxProducts(1)¶
|   +--jnxServices(2)¶
|   +--jnxMibs(3)¶
|   |   +--jnxBoxAnatomy(1)¶
|   |   |   +--jnxContainersTable(6) ¶
|   |   |   +--jnxContentsTable(8)¶
|   |   |   +--jnxLEDTTable(10)¶
|   |   |   +--jnxFilledTable(12)¶
|   |   |   +--jnxOperatingTable(13)¶
|   |   |   |   +--jnxOperatingEntry(1)¶
|   |   |   |   |   +-- jnxOperatingContentsIndex(1)¶
|   |   |   |   |   +-- jnxOperatingL1Index(2)¶
|   |   |   |   |   +-- jnxOperatingL2Index(3)¶
|   |   |   |   |   +-- jnxOperatingL3Index(4)¶
|   |   |   |   |   +-- jnxOperatingDescr(5)¶
|   |   |   |   |   +-- jnxOperatingState(6)¶
|   |   |   |   |   +-- jnxOperatingTemp(7)¶
|   |   |   |   |   +-- jnxOperatingCPU(8)¶
|   |   |   |   |   +-- jnxOperatingISR(9)¶
|   |   |   |   |   +-- jnxOperatingDRAMSize(10)¶
|   |   |   |   |   +-- jnxOperatingBuffer(11)¶
|   |   |   |   |   +-- jnxOperatingHeap(12)¶
|   |   |   |   |   +-- jnxOperatingUpTime(13)¶
|   |   |   |   |   +-- jnxOperatingLastRestart(14)¶
|   |   |   |   |   +-- jnxOperatingMemory(15)¶
|   |   |   |   |   +-- jnxOperatingStateOrdered(16)¶
|   |   |   +--jnxRedundancyTable(14)¶
|   |   +--jnxFruTable(15)¶

```

After each object description is a value in parenthesis, such as (1). This value can be used to enter an OID for the specific object. For example, to gather information on the CPU, you can type the object description (`jnxOperatingCPU`) or the OID (`.1.3.6.1.4.1.2636.3.1.13.1.8`).

Action To check CPU utilization using the Juniper enterprise chassis MIB, from a management station that has access to the router, and using a tool such as Snmpwalk, enter the following command:

```
user-bsd# snmpwalk [common arguments] hostname community object-id
```

Sample Output

```
user-nms % snmpwalk -Os -M /volume/~mibs -m all tp1 public jnxOperatingCPU
jnxOperatingCPU.1.1.1.0 = Gauge32: 0
jnxOperatingCPU.1.1.2.0 = Gauge32: 0
jnxOperatingCPU.1.1.3.0 = Gauge32: 0
jnxOperatingCPU.2.1.0.0 = Gauge32: 0
jnxOperatingCPU.4.1.1.0 = Gauge32: 0
jnxOperatingCPU.4.1.2.0 = Gauge32: 0
jnxOperatingCPU.4.1.3.0 = Gauge32: 0
jnxOperatingCPU.4.1.4.0 = Gauge32: 0
jnxOperatingCPU.6.1.1.0 = Gauge32: 224
jnxOperatingCPU.6.1.2.0 = Gauge32: 224
jnxOperatingCPU.7.1.0.0 = Gauge32: 2
jnxOperatingCPU.7.2.0.0 = Gauge32: 2
jnxOperatingCPU.8.1.1.0 = Gauge32: 0
jnxOperatingCPU.8.2.3.0 = Gauge32: 0
jnxOperatingCPU.8.2.4.0 = Gauge32: 0
jnxOperatingCPU.9.1.0.0 = Gauge32: 6
jnxOperatingCPU.9.1.1.0 = Gauge32: 0

user-nms % snmpwalk -Os -M /volume/~mibs -m all tp1 public jnxOperatingDesc
jnxOperatingDescr.1.1.1.0 = midplane
jnxOperatingDescr.1.1.2.0 = midplane
jnxOperatingDescr.1.1.3.0 = midplane
jnxOperatingDescr.2.1.0.0 = Power Supply A
jnxOperatingDescr.4.1.1.0 = Left Tray front fan
jnxOperatingDescr.4.1.2.0 = Left Tray second fan
jnxOperatingDescr.4.1.3.0 = Left Tray third fan
jnxOperatingDescr.4.1.4.0 = Left Tray fourth fan
jnxOperatingDescr.6.1.1.0 = CFEB Internet Processor IIv1
jnxOperatingDescr.6.1.2.0 = CFEB Internet Processor IIv1
jnxOperatingDescr.7.1.0.0 = FPC @ 0/*/*
jnxOperatingDescr.7.2.0.0 = FPC @ 1/*/*
jnxOperatingDescr.8.1.1.0 = PIC: 4x OC-3 SONET, MM @ 0/0/*
jnxOperatingDescr.8.2.3.0 = PIC: 1x Tunnel @ 1/2/*
jnxOperatingDescr.8.2.4.0 = PIC: 1x G/E, 1000 BASE-SX @ 1/3/*
jnxOperatingDescr.9.1.0.0 = Routing Engine
jnxOperatingDescr.9.1.1.0 = Routing Engine PCMCIA Card
```

What It Means The sample output shows the percentage CPU utilization on router, tp1. The Routing Engine (9.1.0.0) has 6 percent CPU utilization, the two CFEB Internet Processors IIv1 (6.1.1.0 and 6.1.2.0) have 22 percent each, and the FPCs (7.1.0.0 and 7.2.0.0) have 2 percent each. Components with a value of zero indicate that the information is either unavailable or inapplicable.

The output for the jnxOperatingDesc object provides a description of the separate instances in the jnxOperatingCPU object. For example, 9.1.0.0 represents the Routing Engine.

Step 2: Check CPU Utilization per Process

Purpose The standard system application MIB (RFC 2287, *Definitions of System-Level Managed Objects for Applications*), describes a set of managed objects that are restricted to information that can be determined from the system itself. The object `sysAppElmtRunCPU` provides information about applications and associated elements that have run or are currently running on the host system. (See Figure 21.)

Figure 21: System Application MIB Tree

```
+--System: Application MIB
|
| +--sysAppObj
| +--sysAppInstalled
| +--sysAppRun
| +--sysAppMap
| +--sysAppNotifications
| +--sysAppConformance
| | +--sysAppMIBCompliances
| | +--sysAppMIBGroups
| | | +--sysAppRunGroup
| | | | +--sysAppRunStarted
| | | | +--sysAppRunCurrentState
| | | | +--sysAppPastRunStarted
| | | | +--sysAppPastRunExitState
| | | | +--sysAppPastRunTimeEnded
| | | | +--sysAppElmtRunInstallID
| | | | +--sysAppElmtRunTimeStarted
| | | | +--sysAppElmtRunState
| | | | +--sysAppElmtRunName
| | | | +--sysAppElmtRunParameters
| | | | +--sysAppElmtRunCPU
| | | | +--sysAppElmtRunMemory
| | | | +--sysAppElmtRunNumFiles
| | | | +--sysAppElmtRunUser
|
| [ ...Output Truncated... ]
```

Action To check CPU utilization per process, from a management station that has access to the router, and using a tool such as `Snmpwalk`, enter the following command:

```
user-bsd# snmpwalk [common arguments] hostname community object-id
```

Sample Output

```
use-nms % snmpwalk -Os -M /volume/~mibs -m all tp1 public sysAppElmtRunCPU
sysAppElmtRunCPU.0.0.0 = Timeticks: (278) 0:00:02.78
sysAppElmtRunCPU.0.0.2 = Timeticks: (487) 0:00:04.87
sysAppElmtRunCPU.0.0.3 = Timeticks: (0) 0:00:00.00
sysAppElmtRunCPU.0.0.4 = Timeticks: (1742) 0:00:17.42
sysAppElmtRunCPU.0.0.5 = Timeticks: (13899) 0:02:18.99
sysAppElmtRunCPU.0.0.6 = Timeticks: (79) 0:00:00.79
sysAppElmtRunCPU.0.0.7 = Timeticks: (0) 0:00:00.00
sysAppElmtRunCPU.0.0.8 = Timeticks: (0) 0:00:00.00
sysAppElmtRunCPU.0.0.9 = Timeticks: (0) 0:00:00.00
sysAppElmtRunCPU.0.0.10 = Timeticks: (2229) 0:00:22.29
sysAppElmtRunCPU.0.0.11 = Timeticks: (0) 0:00:00.00
sysAppElmtRunCPU.0.0.12 = Timeticks: (0) 0:00:00.00
sysAppElmtRunCPU.0.0.116 = Timeticks: (25) 0:00:00.25
sysAppElmtRunCPU.0.0.2023 = Timeticks: (0) 0:00:00.00
sysAppElmtRunCPU.0.0.2131 = Timeticks: (1103) 0:00:11.03
sysAppElmtRunCPU.0.0.2160 = Timeticks: (1599) 0:00:15.99
sysAppElmtRunCPU.0.0.2161 = Timeticks: (4) 0:00:00.04
sysAppElmtRunCPU.0.0.2174 = Timeticks: (1168) 0:00:11.68
```

```

sysAppLElmtRunCPU.0.0.2324 = Timeticks: (1738) 0:00:17.38
sysAppLElmtRunCPU.0.0.16781 = Timeticks: (0) 0:00:00.00
sysAppLElmtRunCPU.0.0.18311 = Timeticks: (0) 0:00:00.00
sysAppLElmtRunCPU.0.0.26827 = Timeticks: (2) 0:00:00.02
sysAppLElmtRunCPU.3.1.1 = Timeticks: (483) 0:00:04.83
sysAppLElmtRunCPU.3.2.2163 = Timeticks: (33548776) 3 days, 21:11:27.76
sysAppLElmtRunCPU.3.3.2185 = Timeticks: (1314) 0:00:13.14
sysAppLElmtRunCPU.3.4.2194 = Timeticks: (5282) 0:00:52.82
sysAppLElmtRunCPU.3.7.2168 = Timeticks: (20380) 0:03:23.80
sysAppLElmtRunCPU.3.9.2169 = Timeticks: (6703) 0:01:07.03
sysAppLElmtRunCPU.3.12.2172 = Timeticks: (337) 0:00:03.37
sysAppLElmtRunCPU.3.13.2173 = Timeticks: (36) 0:00:00.36
sysAppLElmtRunCPU.3.14.2164 = Timeticks: (39783) 0:06:37.83
sysAppLElmtRunCPU.3.15.2175 = Timeticks: (4206) 0:00:42.06
sysAppLElmtRunCPU.3.16.2165 = Timeticks: (18) 0:00:00.18
sysAppLElmtRunCPU.3.17.2176 = Timeticks: (61) 0:00:00.61
sysAppLElmtRunCPU.3.19.2177 = Timeticks: (25) 0:00:00.25
sysAppLElmtRunCPU.3.20.2178 = Timeticks: (200) 0:00:02.00
sysAppLElmtRunCPU.3.21.2179 = Timeticks: (38) 0:00:00.38
sysAppLElmtRunCPU.3.23.2188 = Timeticks: (3175) 0:00:31.75
sysAppLElmtRunCPU.3.25.2186 = Timeticks: (44774) 0:07:27.74
sysAppLElmtRunCPU.3.26.2180 = Timeticks: (17) 0:00:00.17
sysAppLElmtRunCPU.3.27.2181 = Timeticks: (48950) 0:08:09.50
sysAppLElmtRunCPU.3.30.2187 = Timeticks: (11) 0:00:00.11
sysAppLElmtRunCPU.3.31.2184 = Timeticks: (93) 0:00:00.93
sysAppLElmtRunCPU.3.34.2171 = Timeticks: (80) 0:00:00.80
sysAppLElmtRunCPU.3.35.2047 = Timeticks: (1585) 0:00:15.85
sysAppLElmtRunCPU.3.36.2189 = Timeticks: (30) 0:00:00.30
sysAppLElmtRunCPU.3.37.2191 = Timeticks: (326) 0:00:03.26
sysAppLElmtRunCPU.5.5.7495 = Timeticks: (24721) 0:04:07.21
sysAppLElmtRunCPU.5.6.2167 = Timeticks: (936) 0:00:09.36
sysAppLElmtRunCPU.5.6.26829 = Timeticks: (1) 0:00:00.01
sysAppLElmtRunCPU.5.8.26828 = Timeticks: (25) 0:00:00.25
sysAppLElmtRunCPU.5.28.2182 = Timeticks: (29234) 0:04:52.34
sysAppLElmtRunCPU.5.29.2183 = Timeticks: (21) 0:00:00.21

user-nms % snmpwalk -Os -M ~/mibs -m all tpls public sysAppLElmtRunName
sysAppLElmtRunName.0.0.0 = (swapper)
sysAppLElmtRunName.0.0.2 = (pagedaemon)
sysAppLElmtRunName.0.0.3 = (vmdaemon)
sysAppLElmtRunName.0.0.4 = (bufdaemon)
sysAppLElmtRunName.0.0.5 = (syncer)
sysAppLElmtRunName.0.0.6 = (netdaemon)
sysAppLElmtRunName.0.0.7 = (if_pfe)
sysAppLElmtRunName.0.0.8 = (if_pfe_listen)
sysAppLElmtRunName.0.0.9 = (cb_poll)
sysAppLElmtRunName.0.0.10 = (vmuncachedaemon)
sysAppLElmtRunName.0.0.11 = (scs_housekeeping)
sysAppLElmtRunName.0.0.12 = (if_pic_listen)
sysAppLElmtRunName.0.0.116 = mfs
sysAppLElmtRunName.0.0.2023 = pccardd
sysAppLElmtRunName.0.0.2131 = cron
sysAppLElmtRunName.0.0.2160 = /sbin/watchdog
sysAppLElmtRunName.0.0.2161 = /usr/sbin/tetd
sysAppLElmtRunName.0.0.2174 = /usr/sbin/tnp.snmpd
sysAppLElmtRunName.0.0.2324 = (peer proxy)
sysAppLElmtRunName.0.0.16781 = /usr/libexec/getty
sysAppLElmtRunName.0.0.18311 = /usr/sbin/xntpd
sysAppLElmtRunName.0.0.26827 = telnetd
sysAppLElmtRunName.3.1.1 = /sbin/preinit
sysAppLElmtRunName.3.2.2163 = /usr/sbin/chassisd
sysAppLElmtRunName.3.3.2185 = /usr/sbin/dfwd
sysAppLElmtRunName.3.4.2194 = /sbin/dcd

```

```

sysAppElmtRunName.3.7.2168 = /usr/sbin/snmpd
sysAppElmtRunName.3.9.2169 = /usr/sbin/mib2d
sysAppElmtRunName.3.12.2172 = /usr/sbin/apsd
sysAppElmtRunName.3.13.2173 = /usr/sbin/vrrpd
sysAppElmtRunName.3.14.2164 = /usr/sbin/alarmd
sysAppElmtRunName.3.15.2175 = /usr/sbin/pfed
sysAppElmtRunName.3.16.2165 = /usr/sbin/craftd
sysAppElmtRunName.3.17.2176 = /usr/sbin/sampled
sysAppElmtRunName.3.19.2177 = /usr/sbin/ilmid
sysAppElmtRunName.3.20.2178 = /usr/sbin/rmopd
sysAppElmtRunName.3.21.2179 = /usr/sbin/cosd
sysAppElmtRunName.3.23.2188 = /usr/sbin/fsad
sysAppElmtRunName.3.25.2186 = /usr/sbin/irsd
sysAppElmtRunName.3.26.2180 = /usr/sbin/nasd
sysAppElmtRunName.3.27.2181 = /usr/sbin/fud
sysAppElmtRunName.3.30.2187 = /usr/sbin/rtspd
sysAppElmtRunName.3.31.2184 = /usr/sbin/smartd
sysAppElmtRunName.3.34.2171 = /usr/sbin/inetd
sysAppElmtRunName.3.35.2047 = syslogd
sysAppElmtRunName.3.36.2189 = /usr/sbin/spd
sysAppElmtRunName.3.37.2191 = /usr/sbin/eccd
sysAppElmtRunName.5.5.7495 = /usr/sbin/rpd
sysAppElmtRunName.5.6.2167 = /usr/sbin/mgd
sysAppElmtRunName.5.6.26829 = mgd: (mgd) (user)/dev/tty0
sysAppElmtRunName.5.8.26828 = -cli
sysAppElmtRunName.5.28.2182 = /usr/sbin/ppmd
sysAppElmtRunName.5.29.2183 = /usr/sbin/lmpd

```

What It Means The sample output shows the number of centi-seconds of total system CPU resources consumed by a particular process. For example, the chassis process (chassisd, 3.2.2163) has consumed 3 days, or 33,548,776 centi-seconds of total system CPU resources.

The `sysAppElmtRunName` object retrieves the name of the OID. For example, `sysAppElmtRunCPU.3.2.2163` represents the chassis process.

Retrieve Version Information about Router Software Components

Purpose RFC 2790, *Host Resources MIB*, describes a set of managed objects that are useful for managing host systems, including routers.

Action To retrieve version information about software components on the router, from a management station that has access to the router and using a tool, such as Snmpwalk, enter the following command:

```
user-bsd# snmpwalk [common arguments] hostname community object-id
```

Sample Output

```
user-nms % snmpwalk -Os -M /volume/~mibs -m all tp1 public .1.3.6.1.2.1.25.6.3
hrSWInstalledIndex.2 = 2
hrSWInstalledIndex.3 = 3
hrSWInstalledIndex.4 = 4
hrSWInstalledIndex.5 = 5
hrSWInstalledIndex.6 = 6
hrSWInstalledIndex.9 = 9
hrSWInstalledName.2 = "JUNOS Base OS Software Suite [6.0R1.5]"
hrSWInstalledName.3 = "JUNOS Kernel Software Suite [6.0R1.5]"
hrSWInstalledName.4 = "JUNOS Packet Forwarding Engine Support (M7i/M10i)
[6.0R1.5]"
hrSWInstalledName.5 = "JUNOS Routing Software Suite [6.0R1.5]"
hrSWInstalledName.6 = "JUNOS Online Documentation [6.0R1.5]"
hrSWInstalledName.9 = "JUNOS Support Tools Package [6.0-20031122-unocM2]"
hrSWInstalledID.2 = OID: zeroDotZero
hrSWInstalledID.3 = OID: zeroDotZero
hrSWInstalledID.4 = OID: zeroDotZero
hrSWInstalledID.5 = OID: zeroDotZero
hrSWInstalledID.6 = OID: zeroDotZero
hrSWInstalledID.9 = OID: zeroDotZero
hrSWInstalledType.2 = operatingSystem(2)
hrSWInstalledType.3 = operatingSystem(2)
hrSWInstalledType.4 = operatingSystem(2)
hrSWInstalledType.5 = operatingSystem(2)
hrSWInstalledType.6 = application(4)
hrSWInstalledType.9 = operatingSystem(2)
hrSWInstalledDate.2 = 2003-8-10,20:34:45.0,-7:0
hrSWInstalledDate.3 = 2003-8-10,20:35:21.0,-7:0
hrSWInstalledDate.4 = 2003-8-10,20:36:30.0,-7:0
hrSWInstalledDate.5 = 2003-8-10,20:36:47.0,-7:0
hrSWInstalledDate.6 = 2003-8-10,20:36:51.0,-7:0
hrSWInstalledDate.9 = 2003-11-22,4:8:47.0,-8:0a1
```

What It Means The sample output shows the version information for various software components on the router.

Part 4

Gather System Management Information

- Display Basic Chassis Information on page 227
- Display and Locate Files and Directories on page 231
- Check Time on a Router on page 239
- Check User Accounts and Permissions on page 247

Chapter 18

Display Basic Chassis Information

This chapter describes how to obtain basic system information, including a list of all Flexible PIC Concentrators (FPCs) and Physical Interface Cards (PICs) installed in the router chassis, the hardware version level, and the serial number. (See Table 40.)

Table 40: Checklist for Displaying Basic Chassis Information

Display Basic Chassis Information Task	Command or Action
Display Basic Chassis Information on page 228	show chassis hardware <detail>

Display Basic Chassis Information

Purpose Before you return a router component to Juniper Networks, you must contact the Juniper Networks Technical Assistance Center (JTAC) with the serial number of the failed component and failure information. JTAC will then grant a Return Materials Authorization (RMA).

Action To display a list of the serial numbers of components installed in the router chassis, use the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> show chassis hardware <detail>
```

Sample Output user@host> show chassis hardware

Hardware inventory:

Item	Version	Part number	Serial number	Description
Chassis			25708	M20
Backplane	REV 03	710-002334	BB9738	
Power Supply A	REV 06	740-001465	005234	AC
Power Supply B	REV 06	740-001465	005237	AC
Display	REV 04	710-001519	BA4681	
Routing Engine 0	REV 06	740-003239	1000224893	RE-2.0
Routing Engine 1	REV 06	740-003239	9000022146	RE-2.0
SSB slot 0	REV 02	710-001951	AZ8112	Internet Processor II
SSB slot 1	N/A	N/A	N/A	backup
FPC 0	REV 03	710-003308	BD8455	E-FPC
PIC 0	REV 08	750-002303	AZ5310	4x F/E, 100 BASE-TX
PIC 1	REV 07	750-004745	BC9368	2x CT3-NxDSO
FPC 1	REV 03	710-003308	BB9032	E-FPC
PIC 0	REV 03	750-002914	BC0131	2x OC-3 ATM, MM

```
user@host> show chassis hardware
```

Hardware inventory:

Item	Version	Part number	Serial number	Description
Chassis			00159	M40
Backplane	REV 08	710-000073	AA2125	
Power Supply B	Rev A1	740-000235	000289	DC
Maxicab	REV 08	710-000229	CA4516	
Minicab	REV 04	710-001739	CA4610	
Display	REV 07	710-000150	AA5145	
Routing Engine	REV 07	740-005022	P10865702236	RE-3.0
SCB	REV 03	710-007684	CA3900	Internet Processor II
FPC 1	REV 01	710-001292	AL7435	
PIC 0	REV 03	750-000617	AA3530	1x OC-48 SONET, SMIR
FPC 2	REV 09	710-000175	AA4740	
PIC 0	REV 03	750-000617	AA4557	1x OC-48 SONET, SMIR
FPC 3	REV 01	710-001292	AB4775	
PIC 0	REV 03	750-000612	AA1771	2x OC-3 ATM, MM
PIC 1	REV 03	750-002977	AV3457	2x OC-3 ATM, MM
FPC 5	REV 01	710-001292	AC5118	
PIC 1	REV 03	750-003628	AS8882	1x G/E, 1000 BASE-LH

```

user@host> show chassis hardware detail
Hardware inventory:

```

Item	Version	Part number	Serial number	Description
Chassis			25708	M20
Backplane	REV 03	710-002334	BB9738	
Power Supply A	REV 06	740-001465	005234	AC
Power Supply B	REV 06	740-001465	005237	AC
Display	REV 04	710-001519	BA4681	
Routing Engine 0	REV 06	740-003239	1000224893	RE-2.0
Routing Engine 0			58000007348d9a01	RE-2.0
Routing Engine 1	REV 06	740-003239	9000022146	RE-2.0
Routing Engine 1			d800000734745701	RE-2.0
SSB slot 0	REV 02	710-001951	AZ8112	Internet Processor II
SSRAM bank 0	REV 02	710-001385	242525	2 Mbytes
SSRAM bank 1	REV 02	710-001385	242741	2 Mbytes
SSRAM bank 2	REV 02	710-001385	242886	2 Mbytes
SSRAM bank 3	REV 02	710-001385	242482	2 Mbytes
SSB slot 1	N/A	N/A	N/A	backup
FPC 0	REV 03	710-003308	BD8455	E-FPC
SSRAM	REV 02	710-001385	241669	2 Mbytes
SDRAM bank 0	REV 01	710-000099	0003409	64 Mbytes
SDRAM bank 1	REV 01	710-000099	0003408	64 Mbytes
PIC 0	REV 08	750-002303	AZ5310	4x F/E, 100 BASE-TX
PIC 1	REV 07	750-004745	BC9368	2x CT3-NxDS0
FPC 1	REV 03	710-003308	BB9032	E-FPC
SSRAM	REV 01	710-001385	V00818	2 Mbytes
SDRAM bank 0	REV 01	710-000099	0003803	64 Mbytes
SDRAM bank 1	REV 01	710-000099	0003847	64 Mbytes
PIC 0	REV 03	750-002914	BC0131	2x OC-3 ATM, MM

What It Means The sample output is for an M20 and an M40 router. It shows a list of all FPCs and PICs installed in the router chassis, including the hardware version level and serial number.

The **detail** option displays detailed information about hardware, including memory, hardware version level, serial number, and additional information about memory.

If the Routing Engine is identified by a 10- and 16-digit serial number, both numbers are displayed in the output for the **detail** option, and are especially important when processing an RMA for such a Routing Engine. In addition, when you request an RMA for the M40 router, include the **maxicab** serial number.

Table 41 provides a description of all the output fields for the **show chassis hardware** command.

Table 41: Output fields for the show chassis hardware command

Output field	Description
Item	<p>(For M-series routers) Chassis component. Information is displayed about the backplane; power supplies; Routing Engine; maxicab (the connection between the Routing Engine and the backplane, for the M40 router only); System Control Board (SCB), System and Switch Board (SSB), Switching and Forwarding Module (SFM), or Forwarding Engine Board (FEB); Miscellaneous Control Subsystem (MCS) and PFE clock generator (PCG) (for the M160 router only); and each FPC and PIC.</p> <p>(For T-series platforms) Chassis component. Information is displayed about the backplane, power supplies, midplane, Control Board (CB), Connector Interface Panel (CIP), FPC, Front Panel Module (FPM) (craft interface), Power Entry Module (PEM), PIC, SONET Clock Generator (SCG), Small Form-factor Pluggable (SFP) modules, Switch Interface Board (SIB), and Switch Processor Mezzanine Board (SPMB).</p>
Version	Revision level of the chassis component.
Part number	Part number of the chassis component.
Serial number	Serial number of the chassis component. For all RMAs, the chassis serial number must be provided to JTAC. If the RMA is for the chassis itself, you must obtain the backplane or midplane serial number as well.
Description	Brief description of the hardware item.



NOTE: When you request an RMA, you must also include output from the **show chassis environment** command, the **show version** command, and the troubleshooting output used to identify the failure.

Chapter 19

Display and Locate Files and Directories

This chapter describes how to display and locate files and directories on a router.
(See Table 42.)

Table 42: Checklist for Displaying and Locating Files and Directories on a Router

Display and Locate Files and Directories Tasks	Command or Action
Copy a File on a Routing Engine on page 232	
1. Copy a File from One Routing Engine to Another on page 232	<code>file copy source destination</code>
2. Copy Files between the Local Router and a Remote System on page 232	<code>file copy filename ftp://hostname/path/filename</code> <code>file copy filename ftp://user:password@hostname/filename</code> <code>file copy filename ftp://user@hostname/filename</code> <code>file copy filename scp://user@hostname/path/filename</code>
Maintain a Single Configuration File for Both Routing Engines on page 234	
1. Configure the New Group on page 234	<code>[edit groups]</code> <code>set group-name</code> <code>[edit groups re0]</code> <code>set interfaces interface name unit unit family inet address address</code> <code>[edit groups re0 system]</code> <code>set host-name hostname</code> <code>show</code> <code>commit</code>
2. Apply the New Group on page 236	<code>[edit]</code> <code>set apply-groups group-name</code> <code>show</code> <code>commit</code>
List Files and Directories on a Router on page 237	<code>file list filename or directory</code>
Display File Contents on page 237	<code>file show filename</code>
Rename a File on a Router on page 238	<code>file rename source destination</code>
Delete a File on a Router on page 238	<code>file delete filename</code>

Copy a File on a Routing Engine

Purpose When you configure one Routing Engine and another Routing Engine needs to have a similar configuration, or when you upgrade the JUNOS software version on one Routing Engine, you can simplify the process by copying files from one Routing Engine to another.

Steps To Take To copy a file, follow these steps:

1. Copy a File from One Routing Engine to Another on page 232
2. Copy Files between the Local Router and a Remote System on page 232

Step 1: Copy a File from One Routing Engine to Another

Purpose When you have a dual Routing Engine configuration, you can copy a configuration file from Routing Engine 0 to Routing Engine 1 or vice versa.

Action To copy a configuration file from Routing Engine 0 to Routing Engine 1, use the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> file copy source destination
```

Sample Output user@host> file copy /config/juniper.conf re1:/var/tmp/copied-juniper.conf

What It Means In this case, *source* is the name of the configuration file on Routing Engine 0. Configuration files are stored in the directory /config. The active configuration is /config/juniper.conf, and older configurations are in /config/juniper.conf {1...9}. *destination* is a file on Routing Engine 1.



NOTE: Refer to “Maintain a Single Configuration File for Both Routing Engines” on page 234 for details about naming the Routing Engines correctly.

Step 2: Copy Files between the Local Router and a Remote System

Action You can copy a configuration file from a Routing Engine to a remote system in the network using the File Transfer Protocol (FTP) or secure copy protocol (scp) in any one of the following ways:

- To use anonymous FTP to copy a local file to a remote system, enter the following command:

```
root@host> file copy filename ftp://hostname/filename
```

In the following example, /config/juniper.conf is the local file and hostname is the FTP server:

```
root@host> file copy /config/juniper.conf ftp://hostname/juniper.conf
Receiving ftp: //hostname/juniper.conf (2198 bytes): 100%
2198 bytes transferred in 0.0 seconds (2.69 MBps)
```


- To use FTP where a valid username and password are required, enter the following command:

```
root@host> file copy filename ftp://user:password@hostname/filename
```

In the following example, /config/juniper.conf is the local file, user is the username, testing123 is the password, and hostname is the FTP server:

```
root@host> file copy /config/juniper.conf
ftp://user:testing123@hostname/juniper.conf
Receiving ftp: //user:testing123@hostname/juniper.conf (2198 bytes): 100%
2198 bytes transferred in 0.0 seconds (2.69 MBps)
```

- To use FTP where you require more privacy and are prompted for a password, enter the following command:

```
root@host> file copy filename ftp://user@hostname/filename
```

In the following example, /config/juniper.conf is the local file, user is the username, and hostname is the FTP server:

```
root@host> file copy /config/juniper.conf
ftp://user@hostname/juniper.conf
Password for user@hostname: *****
Receiving ftp: //user@hostname/juniper.conf (2198 bytes): 100%
2198 bytes transferred in 0.0 seconds (2.69 MBps)
```

- To use scp to copy a local file to a remote system, enter the following command:

```
root@host> file copy filename scp://user@hostname/path/filename
```

In the following example, /config/juniper.conf is the local file, user is the username, and ssh-host is the scp server:

```
root@host> file copy /config/juniper.conf
scp://user@ssh-host/tmp/juniper.conf
user@ssh-host's password: *****
juniper.conf      100%
|*****|
*****| 2198      00:00
```



NOTE: You cannot use scp or ssh to copy a file in the worldwide version of the JUNOS software.

Maintain a Single Configuration File for Both Routing Engines

Purpose For routers that support multiple Routing Engines, you can specify **re0** and **re1** as group names to ensure that the correct IP addresses are used for each Routing Engine and to maintain a single configuration file for both Routing Engines. It is important that the names of the Routing Engines correspond to a slot position because the names **re0** and **re1** are special group names that you must use for the Routing Engines to recognize which configuration statement to use. Routing Engine 0 must be in slot position 0 and must be named **re0**, and Routing Engine 1 must be in slot position 1 and must be named **re1**.

Steps to Take To maintain a single configuration file for both Routing Engines, follow these steps:

1. Configure the New Group on page 234
2. Apply the New Group on page 236

Step 1: Configure the New Group

Purpose Each **re0** or **re1** group typically contains, at a minimum, the configuration for the hostname and the management interface (**fxp0**). If each Routing Engine uses a different management interface, the group must also contain the configuration for the backup router and static routes.

Action To configure the **re0** and **re1** groups, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit groups
```

2. Configure the group:

```
[edit groups]
user@host# set group-name
```

For example:

```
[edit groups]
user@host# set re0
```

3. To configure the management interface, go to the following hierarchy level:

```
[edit groups]
user@host# edit groups re0
```

4. Include the following statement:

```
[edit groups re0]
user@host# set interfaces interface-name unit unit family inet address
address
```

For example:

```
[edit groups re0]
user@host# set interfaces fxp0 unit 0 family inet address 1.1.1.1/24
```

5. To configure the hostname, go to the following hierarchy level:

```
[edit groups re0]
user@host# edit groups re0 system
```

6. Include the following statement:

```
[edit groups re0 system]
user@host# set host-name hostname
```

For example:

```
[edit groups re0 system]
user@host# set host-name foo-re0
```

7. Verify the configuration:

```
[edit groups re0]
user@host# show
re0 {
  system {
    host-name foo-re0;
  }
  interfaces {
    fxp0 {
      unit 0 {
        family inet {
          address 1.1.1.1/24;
        }
      }
    }
  }
}
```

8. Commit the configuration:

```
user@host# commit
```

9. Repeat Step 1 through Step 8 for the **re1** group.

What It Means The sample output in Step 7 shows that the **re0** group contains the minimum configuration for a group, the hostname, and the management interface (**fxp0**). If each Routing Engine uses a different management interface, the group must also contain the configuration for the backup router and static routes.

Step 2: Apply the New Group

Action To apply the re0 group to maintain a single configuration file for both Routing Engines, follow these steps:

1. In configuration mode, go to the top hierarchy level and include the following statement:

```
user@host# [edit]
user@host# set apply-groups group-name
```

For example:

```
user@host# [edit]
user@host# set apply-groups re0
```

2. Verify the configuration:

```
user@host# show
groups {
  re0 {
    system {
      host-name foo-re0;
    }
    interfaces {
      fxp0 {
        unit 0 {
          family inet {
            address 1.1.1.1/24;
          }
        }
      }
    }
  }
  re1 {
    system {
      host-name foo-re1;
    }
    interfaces {
      fxp0 {
        unit 0 {
          family inet {
            address 1.1.1.2/24;
          }
        }
      }
    }
  }
}
apply-groups [ re0 re1 ];
```

3. Commit the configuration:

```
user@host# commit
```

What It Means The sample output shows that each group, **re0** and **re1**, has its own IP address that is used for each Routing Engine to maintain a single configuration file.

List Files and Directories on a Router

Purpose If a system board crashes, you must check that certain files are in specific directories.

Action To display files in the `/var/tmp` and `var/crash` directories, use the following CLI operational mode command:

```
user@host> file list filename or directory
```

Sample Output

```
user@host> file list /var/tmp/
sampled.pkts
vi.recover/

user@host> file list /var/crash/
bounds
minfree
vmcore.0
```

What It Means The sample output shows the files in the `/var/tmp/` and `/var/crash/` directories. The Juniper Networks Technical Assistance Center (JTAC) can ask you to verify the existence of similar files.

Display File Contents

Action To display the contents of a file on the local router, use the following CLI operational mode command:

```
user@host> file show filename
```

Sample Output

```
user@host> file show /var/log/messages
Apr 13 21:00:08 romney /kernel: so-1/1/2: loopback suspected; going to standby.
Apr 13 21:00:40 romney /kernel: so-1/1/2: loopback suspected; going to standby.
Apr 13 21:02:48 romney last message repeated 4 times
Apr 13 21:07:04 romney last message repeated 8 times
Apr 13 21:07:13 romney /kernel: so-1/1/0: Clearing SONET alarm(s) RDI-P
Apr 13 21:07:29 romney /kernel: so-1/1/0: Asserting SONET alarm(s) RDI-P
Apr 13 21:07:36 romney /kernel: so-1/1/2: loopback suspected; going to standby.
Apr 13 21:08:08 romney /kernel: so-1/1/2: loopback suspected; going to standby.
...Output truncated...
```

What It Means The sample output shows the contents of the `/var/log/messages` file.

Rename a File on a Router

Action To rename a file on the local router, use the following CLI operational mode command:

```
user@host> file rename source destination
```

Sample Output

```
user@host> file list /var/tmp
dcd.core
rpd.core
snmpd.core
user@host> file rename /var/tmp/dcd.core /var/tmp/dcd.core.990413
user@host> file list /var/tmp
dcd.core.990413
rpd.core
snmpd.core
```

What It Means The sample output shows that the dcd.core file was renamed to dcd.core.990413. The original name of the file is the *source* and the new name for the file is the *destination*.

Delete a File on a Router

Action To delete a file on the local router, use the following CLI operational mode command:

```
user@host> file delete filename
```

Sample Output

```
user@host> file list /var/tmp
dcd.core
rpd.core
snmpd.core
user@host> file delete /var/tmp/snmpd.core
user@host> file list /var/tmp
dcd.core
rpd.core
```

What It Means The sample output shows that the snmpd.core file was deleted.

Chapter 20

Check Time on a Router

This chapter describes how to display the current time on the router, determine whether router components failed during a problem, and check that the local clock time on the router is synchronized with the time on the Network Time Protocol (NTP) server. (See Table 43.)

Table 43: Checklist for Checking Time on a Router

Check Time on a Router	Command or Action
Check the Time on a Router on page 240	show system uptime
Check How Long Router Components Have Been Up on page 240	show chassis fpc detail show chassis routing-engine show chassis feb show chassis scb show chassis sfm show chassis ssb
Check the NTP Peers on page 243	show ntp associations
Check the NTP Status on page 244	show ntp status

Check the Time on a Router

Purpose Display the current time on a router and display information about how long the router, router software, and routing protocols have been running.

Action To check time on a router, use the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> show system uptime
```

Sample Output

```
user@host> show system uptime
Current time:      1998-10-13 19:45:47 UTC
System booted:     1998-10-12 20:51:41 UTC (22:54:06 ago)
Protocols started: 1998-10-13 19:33:45 UTC (00:12:02 ago)
Last configured:   1998-10-13 19:33:45 UTC (00:12:02 ago) by abc
12:45PM up 22:54, 2 users, load averages: 0.07, 0.02, 0.01
```

What It Means The sample output shows the current system time in UTC, the date and time when the router was last booted and how long it has been running, when the routing protocols were last started and how long they have been running, when a configuration was last committed, and the name of the user who issued the last commit command. If a different time zone is configured, the output shows that time zone. For information on configuring the time zone, see the *JUNOS System Basics Configuration Guide*.

The sample output shows that the current time is 12:45 PM, the router has been operational for 22:54 hours, and two users are logged in to the router. The output also shows that the load average is 0.07 seconds for the last minute, 0.02 seconds for the last 5 minutes, and 0.01 seconds for the last 15 minutes.

Check How Long Router Components Have Been Up

Purpose When a problem occurs and you check the system to see how long it has been up, you may find that the `show system uptime` command displays the current time and information about how long the router, router software, and routing protocols have been running, but does not tell you if a component failed. Determining whether a component failed when a problem occurred with the router is an important step in the diagnosis of a problem.

Action To check how long router components have been up, issue the `show chassis` command for the components on your router:

```
user@host> show chassis fpc detail
user@host> show chassis routing-engine
user@host> show chassis feb
user@host> show chassis scb
user@host> show chassis sfm
user@host> show chassis ssb
```


Sample Output The following sample output is for an M20 router:

```

user@host> show chassis fpc detail
Slot 0 information:
  State                               Empty
Slot 1 information:
  State                               Online
  Logical slot                        0
  Temperature                         32 degrees C / 89 degrees F
  Total CPU DRAM                      8 MB
  Total SRAM                          1 MB
  Total SDRAM                         128 MB
  Total notification SDRAM            24 MB
  I/O Manager ASIC information        Version 2.0, Foundry IBM, Part number 0
  Start time:                        2003-09-23 17:20:42 UTC
  Uptime:                             1 day, 4 hours, 45 minutes, 14 seconds
Slot 2 information:
  State                               Empty
Slot 3 information:
  State                               Online
  Logical slot                        1
  Temperature                         32 degrees C / 89 degrees F
  Total CPU DRAM                      8 MB
  Total SRAM                          1 MB
  Total SDRAM                         128 MB
  Total notification SDRAM            24 MB
  I/O Manager ASIC information        Version 1.1, Foundry IBM, Part number 0
  Start time:                        2003-09-12 01:28:16 UTC
  Uptime:                             12 days, 20 hours, 37 minutes, 40 seconds

user@host> show chassis routing-engine
Routing Engine status:
Slot 0:
  Current state                       Master
  Election priority                   Master (default)
  Temperature                         30 degrees C / 86 degrees F
  DRAM                                768 MB
  Memory utilization                   17 percent
  CPU utilization:
    User                             0 percent
    Background                       0 percent
    Kernel                           1 percent
    Interrupt                         0 percent
    Idle                             99 percent
  Model                              RE-2.0
  Serial ID                           58000007348d9a01
  Start time                          2003-09-19 07:05:20 PDT
  Uptime                              6 hours, 42 minutes, 26 seconds
  Load averages:                     1 minute   5 minute   15 minute
                                       0.00       0.00       0.00
Routing Engine status:
Slot 1:
  Current state                       Backup
  Election priority                   Backup (default)
  Temperature                         30 degrees C / 86 degrees F
  DRAM                                768 MB
  Memory utilization                   0 percent
  CPU utilization:
    User                             0 percent
    Background                       0 percent
    Kernel                           0 percent
    Interrupt                         0 percent
    Idle                             100 percent

```

```
Model RE-2.0
Serial ID d800000734745701
Start time 2003-06-17 16:37:33 PDT
Uptime 93 days, 20 hours, 58 minutes, 14 seconds

user@host> show chassis ssb
SSB status:
Slot 0 information:
  State Master
  Temperature 33 degrees C / 91 degrees F
  CPU utilization 2 percent
  Interrupt utilization 0 percent
  Heap utilization 17 percent
  Buffer utilization 43 percent
  Total CPU DRAM 64 MB
  Internet Processor II Version 1, Foundry IBM, Part number 9
  Start time: 2003-09-19 07:06:52 PDT
  Uptime: 6 hours, 43 minutes, 52 seconds
Slot 1 information:
  State Backup
```

What It Means The sample output shows the time when the component started running and how long the component has been running. A short uptime can indicate a problem with the component.

Check the NTP Peers

Purpose Ensure that the clock time on the router is synchronized with the time on the NTP server.

Action To check NTP peers, enter the following JUNOS CLI operational mode command:

```
user@host> show ntp associations
```

Sample Output 1

```
user@host> show ntp associations
      remote      refid      st t when poll reach  delay  offset  jitter
=====
*coetanian.junip .GPS.          1 u  22   64  377   6.861  -1.297   0.811
```

Sample Output 2

```
user@jhost> show ntp associations
>      remote      refid      st t when poll reach  delay  offset  jitter
> =====
> ntp1.usno.navy. PSC.          1 -  44   64   77   86.829  -1830.3  915.177
> Tick.UH.EDU     USNO.         1 -  36   64   77   42.560  -1835.3  917.667
```

What It Means Sample output 1 is synchronized with the NTP server because there is an asterisk (*) at the beginning of the output. Also, the router with the asterisk (*) is the master router and the system is synchronizing with this NTP server.

Sample output 2 shows that the time on the server and router is so far apart that NTP will not attempt to synchronize. The **offset** value of **1830** is too large a difference and the **jitter** value of **917.667** is also too large to provide reliability to the **offset** value.

As a general rule, if the time difference between the server and the router is less than 100 seconds, NTP adjusts the router's clock speed so that it drifts towards the server's time. For instance, if the router clock is running fast, NTP slows the clock down so that the time of day only advances 900 milliseconds every time the server's clock advances a full second. The time on the two clocks gradually becomes identical. When the clock time is the same, NTP adjusts the clock on the router to keep in step with the server's time.

For more detailed information on configuring the NTP server, see the *JUNOS System Basics Configuration Guide*.

Check the NTP Status

Purpose View the configuration of the NTP server and the status of the system.

Action To check NTP status, enter the following JUNOS CLI operational mode command:

```
user@host> show ntp status
```

Sample Output

```
user@host> show ntp status
status=0644 leap_none, sync_ntp, 4 events, event_peer/strat_chg,
processor="i386", system="JUNOS5.7-20030919-IMAYzc", leap=00, stratum=2,
precision=-28, rootdelay=6.861, rootdispersion=10.465, peer=11004,
refid=coetanian.company.net,
reftime=c315b20a.a5c768df Fri, Sep 19 2003 9:49:14.647, poll=6,
clock=c315b22a.1b31a08b Fri, Sep 19 2003 9:49:46.106, state=4,
phase=-1.297, frequency=74.659, jitter=0.725, stability=0.005
```

What It Means The sample output shows when the clock was last adjusted (reftime), together with its status and most recent exception event. Table 44 lists and describes the fields in the output of the `show ntp status` command.

Table 44: Sample Output Fields for the show ntp status Command

Output Field	Description
status=0644	Internal status flags.
leap_none	The router is not doing a leap second.
sync_ntp	The server and the router are synchronized.
4 events	The accumulated number of events since NTP was started.
event_peer/strat_chg	Last event code.
processor="i386", system="JUNOS5.7-20030919-IMAYzc"	Both fields identify the current system information.
leap=00	An internal value related to leap seconds.
stratum=2	The router stratum, which is always one higher than the stratum of the server to which the router is synchronized. If the router is not synchronized, the value is 16 instead of 2.
precision=-28	Order of magnitude of how small an interval the local system's clock can measure. In this example, -28 means that the system can measure a period of -28 seconds, approximately 1/64 of a microsecond, or 16 nanoseconds.
rootdelay=6.861	One-way delay between the local system and the stratum 0 clock source. Essentially, this is the sum of the delays between this router and its synchronized source, the source and its source, and so on, all the way up to the atomic clock (which is stratum 0).
rootdispersion=10.465	The confidence level of the clock, in microseconds, that encompasses delay, jitter, and so on.
peer=11004	No information is available for this field yet.
refid=coetanian.company.net, reftime=c315b20a.a5c768df Fri, Sep 19 2003 9:49:14.647	These two fields identify the selected and synchronized source, and the last reference time received from it.
poll=6	The delay interval at which the synchronized server polls. In this example, 6 indicates that the server polls every 2 ⁶ seconds, or every 64 seconds.

Output Field	Description
clock=c315b22a.1b31a08b Fri, Sep 19 2003 9:49:46.106	The current time.
state=4	No information is available for this field yet.
phase=-1.297	The calculated offset based on the local time, the server's last reported time, and the sense of the delay between the router and the server. This is measured in milliseconds.
frequency=74.659	The clock frequency, in MHz. Note that this is the same order of magnitude as precision .
jitter=0.725	Variation in the time delay between the router and the server.
stability=0.005	A measure of how often the speed on the router must be changed to keep synchronized with the server. If the local clock is not perfectly stable, it will speed up or slow down, and NTP will have to counteract that tendency.

Chapter 21

Check User Accounts and Permissions

This chapter describes how to check user accounts and permissions. (See Table 45.)

Table 45: Checklist for Checking User Accounts and Permissions

Check User Accounts and Permissions Tasks	Command or Action
Understand User Accounts and Permissions on page 248	
Check Users Logged In To a Router on page 248	show system users
Check for Users in Configuration Mode on page 249	[edit] status
Check the Commands That Users Are Entering on page 250	
1. Configure the Log File for Tracking CLI Commands on page 250	[edit] edit system syslog edit file <i>filename</i> set interactive-commands info show commit
2. Display the Configured Log File on page 251	[edit system syslog] run show log <i>filename</i>
Log a User Out of the Router on page 252	request system logout <i>username</i>
Check When the Last Configuration Change Occurred on page 253	
1. Configure Configuration Change Tracking on page 253	[edit] edit system syslog edit file <i>filename</i> set change-log info show commit
2. Display the Configured Log File on page 254	[edit system syslog] run show log <i>filename</i>
Force a Message to Logged-In User Terminals on page 255	request message all message " <i>text</i> " request message message " <i>text</i> " user <i>username</i>
Check RADIUS Server Connectivity on page 256	[edit system] show run ping <i>IP-address</i>

Understand User Accounts and Permissions

JUNOS software can be configured to support any number of user accounts. Each user account has an access level for which you can define the login name and, optionally, information that identifies the user. After you create an account, the software creates a home directory in the file system for the user.

In this chapter, it is assumed that user accounts and permissions are configured on the router. For more detailed information about creating a user account and configuring permissions, see the *JUNOS Network Management Configuration Guide*.

Check Users Logged In To a Router

Purpose You may need to take note of the users currently logged in to a router.

Action To list all users who are currently logged in to a router, enter the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> show system users
```

Sample Output

```
user@host> show system users
1:49PM PDT up 6:44, 3 users, load averages: 0.00, 0.00, 0.00
USER      TTY      FROM                LOGIN@  IDLE WHAT
jgchan    p0       big.company.net     1:36PM  12  -csh (csh)
user      p1       pink.company.net    1:49PM   -  -cli (cli)
blue      p2       level5.company.net  2:30PM   -  -cli
```

What It Means The sample output lists information about the users who are currently logged in to a router. There are three users, one of whom has not recently accessed the router. Two of the users are running the CLI, and one is working from the UNIX-level shell (csh). Figure 45 lists and describes the fields in the output of the `show system users` command.

Table 46: Description of Output Fields for the `show system users` Command

Field	Description
<i>time</i> and up	Current time, in the local time zone, and how long the router has been operational.
users	Number of users logged in to the router.
load averages	Load averages for the last 1 minute, 5 minutes, and 15 minutes.
USER	Username.
TTY	Terminal through which the user is logged in.
FROM	System from which the user is logged in. A hyphen indicates that the user is logged in through the console.
LOGIN@	Time when the user logged in.
IDLE	How long the user has been idle.
WHAT	Processes that the user is running.

Check for Users in Configuration Mode

Purpose Before you change the configuration or commit a candidate configuration, it is a good idea to check for users in configuration mode.

Action To display users currently editing the configuration, follow these steps:

1. To enter configuration mode, type the following command:

```
user@host> edit
```

For example:

```
user@host> edit
Entering configuration mode
```

2. Enter the following configuration mode command:

```
[edit]
user@host# status
```

For example:

```
user@host> show system users
4:58PM PST up 5 days, 9:52, 5 users, load averages: 0.01, 0.01, 0.00
USER      TTY      FROM          LOGIN@  IDLE WHAT
mwazna    p0       bigpunk.juniper.net  4:58PM   - -cli (cli)
jgchan    p1       bigpunk.juniper.net  2:25PM  2:32 -csh (csh)
jgchan    p2       bigpunk.juniper.net  2:35PM  2:18 cli
taffy     p3       bigpunk.juniper.net  3:28PM   5 -cli (cli)
tmauro    p4       bigpunk.juniper.net  4:16PM  37 cli
```

What It Means The sample output lists the users who are currently logged in to the router. Five users are logged in to the router, with one user logged in twice, **jgchan**. Each user is logged in through a different terminal (TTY—**p0**, **p1**, **p2**, **p3**, and **p4**) from the system **bigpunk.juniper.net**. A hyphen in the **FROM** field indicates that the user logged in through the console.

Additional information includes the time when the user logged in (**LOGIN**), the amount of time the user is not active on the router (**IDLE**), and the processes that the user is running (**WHAT**). In this example, the users are running the command-line interface (**cli**) and the UNIX-level shell (**csh**).

Check the Commands That Users Are Entering

Purpose A common set of operations you can check is when users log in to the router and the CLI commands they issue.

Steps To Take To check the commands that users are entering, follow these steps:

1. Configure the Log File for Tracking CLI Commands on page 250
2. Display the Configured Log File on page 251

Step 1: Configure the Log File for Tracking CLI Commands

Action To configure the log file for tracking CLI commands, follow these steps:

1. In configuration mode, go to the following hierarchy level:

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

2. Configure the log file:

```
[edit system syslog]
user@host# edit file filename
```

For example:

```
[edit system syslog]
user@host# edit file cli-commands
```

3. Configure the interactive-commands facility and severity level:

```
[edit system syslog filename]
user@host# set interactive-commands info
```

4. Verify the configuration:

```
[edit system syslog]
user@host# show
file cli-commands {
    interactive-commands info;
}
```

5. Commit the configuration:

```
user@host# commit
```

What It Means The configuration example shows that the log file `cli-commands` is configured with the interactive-commands facility at the `info` severity level. Table 47 lists and describes the severity levels.

Table 47: Severity Levels

Severity Level	Description
info	Log all top-level CLI commands, including the <code>configure</code> command, and all configuration mode commands.
notice	Log the configuration mode commands <code>rollback</code> and <code>commit</code> .
warning	Log when any software process restarts.

Step 2: Display the Configured Log File

Action To display the log file in configuration mode, enter the following command:

```
[edit system syslog]
user@host# run show log filename
```

For example:

```
[edit system syslog]
user@host# run show log cli-commands
```

Sample Output

```
[edit system syslog]
user@host# run show log cli-commands
Sep 16 11:24:25 nut mgd[3442]: UI_COMMIT_PROGRESS: commit: signaling 'Syslog
daemon', pid 2457, signal 1, status 0
Sep 16 11:24:25 nut mgd[3442]: UI_COMMIT_PROGRESS: commit: signaling 'SNMP
daemon', pid 2592, signal 31, status 0
Sep 16 11:28:36 nut mgd[3442]: UI_CMDLINE_READ_LINE: User 'user', command 'run
show log cli-commands '
Sep 16 11:30:39 nut mgd[3442]: UI_CMDLINE_READ_LINE: User 'user', command 'run
show log security '
Sep 16 11:31:26 nut mgd[3442]: UI_CMDLINE_READ_LINE: User 'user', command 'run
show log messages '
Sep 16 11:41:21 nut mgd[3442]: UI_CMDLINE_READ_LINE: User 'user', command 'edit
file cli-commands '
Sep 16 11:41:25 nut mgd[3442]: UI_CMDLINE_READ_LINE: User 'user', command 'show
'
Sep 16 11:44:57 nut mgd[3442]: UI_CMDLINE_READ_LINE: User 'user', command 'set
interactive-commands info '
Sep 16 14:32:15 nut mgd[3442]: UI_CMDLINE_READ_LINE: User 'user', command 'run
show log cli-commands '
```

What It Means The sample output shows the CLI commands that were entered since the log file was configured.

Log a User Out of the Router

Purpose Disconnect a user session when that session does not terminate after the user logs out.

Action To log a user out of all terminal sessions on a router, enter the following JUNOS CLI operational mode command:

```
user@host> request system logout username
```

Sample Output

```
user@host> show system users
10:07PM up 13 days, 1:25, 2 users, load averages: 0.17, 0.05, 0.02
USER    TTY    FROM                                LOGIN@  IDLE WHAT
harry   p0     hpot-1t.cmpy.net                  10:07PM   - -cli (c1
wizard  p1     hpot-1t.cmpy.net                  10:06PM   - -cli (c1
```

```
user@host> request system logout user harry
```

```
user@host> show system users
10:07PM up 13 days, 1:25, 1 user, load averages: 0.24, 0.06, 0.02
USER    TTY    FROM                                LOGIN@  IDLE WHAT
wizard  p1     hpot-1t.cmpy.net                  10:06PM   - -cli (c1
```

What It Means The sample output for the first `show system users` command shows there were two users on the router, `harry` and `wizard`. The `request system logout user` command was issued to log out user `harry`. Because there is no output to indicate that `harry` was logged out, the `show system users` command was issued again to verify that user `harry` was actually logged out of the router.

Check When the Last Configuration Change Occurred

Purpose When a problem occurs on a router, it is a good idea to check when the last configuration change was made because it may have had some influence on the problem.

Action To check when the last configuration change occurred, follow these steps:

1. Configure Configuration Change Tracking on page 253
2. Display the Configured Log File on page 254

Step 1: Configure Configuration Change Tracking

Action To configure this type of logging, follow these steps:

1. In configuration mode, go to the following hierarchy level:

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

2. Configure the log file:

```
[edit system syslog]
user@host# edit file filename
```

For example:

```
[edit system syslog]
user@host# edit file mw-configuration-changes
```

3. Configure the change-log facility and severity level:

```
[edit system syslog filename]
user@host# set change-log info
```

4. Verify the configuration:

```
[edit system syslog]
user@host# show
file mw-configuration-changes {
  change-log info;
}
```

5. Commit the configuration:

```
user@host# commit
```

Step 2: Display the Configured Log File

Action To display the log file in configuration mode, enter the following command:

```
[edit system syslog]
user@host# run show log filename
```

For example:

```
[edit system syslog]
user@host# run show log mw-configuration-changes
```

Sample Output

```
[edit system syslog]
user@host# run show log mw-configuration-changes
Sep 17 07:03:22 nut mgd[7793]: UI_CFG_AUDIT_OTHER: User 'root' override:
/config/juniper.conf
Sep 17 07:07:21 nut mgd[2751]: UI_CFG_AUDIT_OTHER: User 'root' set: [interfaces
lo0 unit 0 family inet address 127.0.0.1/32]
Sep 17 07:07:21 nut mgd[2751]: UI_CFG_AUDIT_SET: User 'root' set: [system
domain-name] "englab.company.net" -> "englab.company.net"
Sep 17 07:07:21 nut mgd[2751]: UI_CFG_AUDIT_OTHER: User 'root' set: [system
name-server 172.17.28.101]
Sep 17 07:07:22 nut mgd[2751]: UI_CFG_AUDIT_OTHER: User 'root' set: [system
domain-search] "englab.company.net"
Sep 17 07:07:22 nut mgd[2751]: UI_CFG_AUDIT_OTHER: User 'root' set: [system
domain-search] "company.net"
Sep 17 07:07:22 nut mgd[2751]: UI_CFG_AUDIT_OTHER: User 'root' set: [system
domain-search] "jnpr.net"
```

What It Means The sample output shows the contents of the log file and that the last configuration change was on September 17 at 07:07:22.

Force a Message to Logged-In User Terminals

Purpose You have a scheduled maintenance window or have other important information to convey to users logged in to the router.

Action To force a message to the terminals of logged-in users, enter the following JUNOS CLI operational mode command:

```
user@host> request message all message "text"
```

Sample Output user@host> request message all message "This is an experiment, please be patient"

```
Broadcast Message from user@host
(/dev/tty0) at 10:50 PDT...
```

```
This is an experiment, please be patient
```

```
user@host> request message message "Maintenance window in 10 minutes" user maria
Message from user@host on tty0 at 20:27 ...
Maintenance window in 10 minutes
EOF
```

What It Means The sample output shows that the message “This is an experiment, please be patient” was sent to the consoles of all logged-in users, and the message “Maintenance window in 10 minutes” was sent to the console of the logged-in user, maria. For more detailed information about this command, see the *JUNOS Network Management Configuration Guide*.

Syntax request message all message "text"
request message message "text" (terminal terminal-name | user user-name)

Check RADIUS Server Connectivity

Purpose It is important to check connectivity to the RADIUS server when attempting to diagnose an authentication problem.

Action To ensure that you can ping the RADIUS server, follow these steps:

1. In configuration mode, go to the following hierarchy level:

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

2. Determine the IP address of the RADIUS server:

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

For example:

```
[edit system]
user@host# show
host-name nut;
domain-name englab.company.net;
[...Output truncated...]
radius-server {
  10.10.10.5 {
    secret "$9$14bhIM-VYJGDx7-w2gUD"; # SECRET-DATA
    timeout 5;
    retry 3;
  }
  10.10.10.240 {
    secret "$9$hMKrMXwYoDik-VwgaJHk"; # SECRET-DATA
    timeout 5;
    retry 3;
  }
}
[...Output truncated...]
```


3. Ping the IP address of the RADIUS server:

```
user@host# run ping IP address
```

For example:

```
user@host# run ping 10.10.10.5  
PING 10.10.10.5 (10.10.10.5): 56 data bytes  
64 bytes from 10.10.10.5: icmp_seq=0 ttl=254 time=0.402 ms  
64 bytes from 10.10.10.5: icmp_seq=1 ttl=254 time=0.279 ms  
64 bytes from 10.10.10.5: icmp_seq=2 ttl=254 time=0.292 ms  
64 bytes from 10.10.10.5: icmp_seq=3 ttl=254 time=0.283 ms  
64 bytes from 10.10.10.5: icmp_seq=4 ttl=254 time=0.283 ms  
^C  
— 10.10.10.5 ping statistics —  
5 packets transmitted, 5 packets received, 0% packet loss  
round-trip min/avg/max/stddev = 0.271/0.295/0.402/0.036 ms
```

What It Means The sample output shows that the RADIUS server is connected and that the connection is running at a reasonable speed.

Part 5

Search Log Messages

- Track Normal Operations on page 261
- Track Error Conditions on page 273
- Collect Crash Data on page 299

Chapter 22

Track Normal Operations

This chapter describes how to configure system logging to monitor system-wide, high-level operations. (See Table 48.)

Table 48: Checklist for Tracking Normal Operations

Track Normal Operations Tasks	Command or Action
Configure System Logging on page 263	
1. Log Messages to a Local Log File on page 263	[edit] [edit system syslog] set file <i>filename facility level</i> show commit
2. Log Information to a Remote Host on page 265	[edit] [edit system syslog] set host <i>hostname facility level</i> show commit
3. Log Information to a User Terminal on page 266	[edit] [edit system syslog] set user <i>username facility level</i> show commit
4. Log Information to a Router Console on page 267	[edit] [edit system syslog] set console <i>facility level</i> show commit
5. Configure the Number and Size of Log Files on page 267	[edit] [edit system syslog] set archive files <i>number size size</i> show commit or [edit] [edit system syslog file <i>filename</i>] set archive files <i>number size size</i> show commit

Track Normal Operations Tasks	Command or Action
6. Log BGP State Transition Events on page 268	[edit] [edit protocol bgp] set log-updown show commit
7. Display a Log File on page 270	show log <i>filename</i>
8. Monitor Messages in Near-Real Time on page 271	monitor start <i>filename</i>
9. Stop Monitoring Log Files on page 271	monitor stop <i>filename</i> or monitor stop

Configure System Logging

Purpose System logging operations use a system logging mechanism to record system-wide, high-level operations, such as interfaces going up or down and users logging in to or out of a router.

Steps To Take To configure system logging, follow these steps:

1. Log Messages to a Local Log File on page 263
2. Log Information to a Remote Host on page 265
3. Log Information to a User Terminal on page 266
4. Log Information to a Router Console on page 267
5. Configure the Number and Size of Log Files on page 267
6. Log BGP State Transition Events on page 268
7. Display a Log File on page 270
8. Monitor Messages in Near-Real Time on page 271
9. Stop Monitoring Log Files on page 271

Step 1: Log Messages to a Local Log File

Action To log messages to a local log file on the router, follow these steps:

1. In configuration mode, go to the following hierarchy level:

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

2. Configure the file, facility, and level:

```
user@host# set file filename facility level
```

For example:

```
[edit system syslog]
user@host# set file security authorization info
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit system syslog]
user@host# show
file security
authorization info
```

4. Commit the configuration:

```
user@host# commit
```

Table 49 lists the JUNOS system logging facilities. Each message is assigned to a facility, which is a group of messages that are either generated by the same software process or concern a similar condition or activity (such as authentication attempts).

Table 49: JUNOS System Logging Facilities

Facility	Type of Event or Error
any	Any (includes messages from all facilities).
authorization	Authentication and authorization attempts.
change-log	Change to the JUNOS configuration.
conflict-log	Configuration that is inconsistent with router hardware.
cron	Actions performed or errors encountered by the <code>cron</code> process.
daemon	Actions performed or errors encountered by various system processes.
firewall	Packet filtering actions performed by a firewall filter.
interactive-commands	Commands issued at the JUNOS command-line interface (CLI) operational mode prompt.
kernel	Actions performed or errors encountered by the JUNOS kernel.
pfe	Actions performed or errors encountered by the Packet Forwarding Engine.
user	Actions performed or errors encountered by various user-space processes.

Table 50 lists the system log message severity levels supported by the JUNOS software. Each message is assigned a severity level, which indicates how seriously it affects router functioning.

Table 50: System Log Message Severity Levels

Severity Level	Description
emergency	System panic or other condition that causes the router to stop functioning.
alert	Conditions that require immediate correction, such as a corrupted system database.
critical	Critical conditions, such as hard drive errors.
error	Error conditions that generally have less serious consequences than errors at the emergency, alert, and critical levels.
warning	Conditions that warrant monitoring.
notice	Conditions that are not errors but might warrant special handling.
info	Events or nonerror conditions of interest.
debug	Software debugging messages. Specify this level only when directed by a technical support representative.

Step 2: Log Information to a Remote Host

Action To log messages to a remote host, follow these steps:

1. In configuration mode, go to the following hierarchy level:

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

2. Configure the host, facility, and level:

```
user@host# set host hostname facility level
```

For example:

```
[edit system syslog]
user@host# set host junipero.berry.net daemon info
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit system syslog]
user@host# show
host junipero.berry.net
daemon info;
```

4. Commit the configuration:

```
user@host# commit
```

See Also For information on logging facilities and severity levels supported by the JUNOS software, see Table 49 on page 264 and Table 50 on page 265.

Step 3: Log Information to a User Terminal

Action To log messages to a user terminal, follow these steps:

1. In configuration mode, go to the following hierarchy level:

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

2. Configure the user, facility, and level:

```
user@host# set user username facility level
```

For example:

```
[edit system syslog]
user@host# set user alex any critical
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit system syslog]
user@host# show
user alex
any critical
```

4. Commit the configuration:

```
user@host# commit
```

See Also For information on logging facilities and security levels supported by the JUNOS software, see Table 49 on page 264 and Table 50 on page 265.

Step 4: Log Information to a Router Console

Action To log messages to a router console, follow these steps:

1. In configuration mode, go to the following hierarchy level:

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

2. Configure the router console, facility, and level:

```
user@host# set console facility level
```

For example:

```
[edit system syslog]
user@host# set console any error
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit system syslog]
user@host# show
console
any error
```

4. Commit the configuration:

```
user@host# commit
```

See Also For information on logging facilities and security levels supported by the JUNOS software, see Table 49 on page 264 and Table 50 on page 265.

Step 5: Configure the Number and Size of Log Files

Purpose By default, the JUNOS logging facility stops writing messages to a log file when the file reaches 128 KB in size. It closes the file and adds a numerical suffix, then opens and directs messages to a new file with the original name. By default, the JUNOS logging facility creates up to 10 files before it begins overwriting the contents of the oldest file.

Action To configure the number and size of the log files, follow these steps:

1. In configuration mode, go to one of the following hierarchy levels:

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

or

```
[edit]
user@host# edit system syslog filename
```

2. Configure the number and size of the archive files:

```
user@host# set archive files number size size
```

For example:

```
[edit system syslog]
user@host# set archive files 10 size 65536
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit system syslog]
user@host# show
archive size 64k files 10
```

4. Commit the configuration:

```
user@host# commit
```

See Also See the *JUNOS System Basics Configuration Guide* for more detailed explanations and examples of log file configurations.

Step 6: Log BGP State Transition Events

Purpose Border Gateway Protocol (BGP) state transitions indicate a network problem and need to be logged and investigated.

Action To log BGP state transition events to the system log, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocol bgp
```

2. Configure the system log:

```
user@host# set log-updown
```

3. Verify the configuration:

```
user@host# show
```

4. Commit the configuration:

```
user@host# commit
```

What It Means Log messages from BGP state transition events are sufficient to diagnose most BGP session problems. Table 51 lists and describes the six states of a BGP session.

Table 51: Six States of a BGP Session

BGP State	Description
Idle	<p>This is the first state of a connection. BGP waits for a start event initiated by an administrator. The start event might be the establishment of a BGP session through router configuration or the resetting of an existing session. After the start event, BGP initializes its resources, resets a connect-retry timer, initiates a TCP transport connection, and starts listening for connections initiated by remote peers. BGP then transitions to a Connect state.</p> <p>If there are errors, BGP falls back to the Idle state.</p>
Connect	<p>BGP waits for the transport protocol connection to complete. If the TCP transport connection is successful, the state transitions to OpenSent.</p> <p>If the transport connection is not successful, the state transitions to Active.</p> <p>If the connect-retry timer has expired, the state remains in the Connect state, the timer is reset, and a transport connection is initiated.</p> <p>With any other event, the state goes back to Idle.</p>
Active	<p>BGP tries to acquire a peer by initiating a transport protocol connection.</p> <p>If it is successful, the state transitions to OpenSent.</p> <p>If the connect-retry timer expires, BGP restarts the connect timer and falls back to the Connect state. BGP continues to listen for a connection that may be initiated from another peer. The state may go back to Idle in case of other events, such as a stop event.</p> <p>In general, a neighbor state flip-flopping between Connect and Active is an indication that there is a problem with the TCP transport connection. Such a problem might be caused by many TCP retransmissions or the inability of a neighbor to reach the IP address of its peer.</p>
OpenSent	<p>BGP receives an open message from its peer. In the OpenSent state, BGP compares its autonomous system (AS) number with the AS number of its peer and recognizes whether the peer belongs to the same AS (internal BGP) or to a different AS (external BGP).</p> <p>The open message is checked for correctness. In case of errors, such as a bad version number of an unacceptable AS, BGP sends an error-notification message and goes back to Idle.</p> <p>For any other errors, such as expiration of the hold timer or a stop event, BGP sends a notification message with the corresponding error code and falls back to the Idle state.</p> <p>If there are no errors, BGP sends keepalive messages and resets the keepalive timer. In this state, the hold time is negotiated. If the hold time is 0, the hold and keepalive timers are not restarted.</p> <p>When a TCP transport disconnect is detected, the state falls back to Active.</p>

BGP State	Description
OpenConfirm	<p>BGP waits for a keepalive or notification message.</p> <p>If a keepalive is received, the state becomes Established, and the neighbor negotiation is complete. If the system receives an update or keepalive message, it restarts the hold timer (assuming that the negotiated hold time is not 0).</p> <p>If a notification message is received, the state falls back to Idle.</p> <p>The system sends periodic keepalive messages at the rate set by the keepalive timer. In case of a transport disconnect notification or in response to a stop event, the state falls back to Idle. In response to other events, the system sends a notification message with a finite state machine (FSM) error code and goes back to Idle.</p>
Established	<p>This is the final state in the neighbor negotiation. In this state, BGP exchanges update packets with its peers and the hold timer is restarted at the receipt of an update or keepalive message when it is not set to zero.</p> <p>If the system receives a notification message, the state falls back to Idle.</p> <p>Update messages are checked for errors, such as missing attributes, duplicate attributes, and so on. If errors are found, a notification is sent to the peer, and the state falls back to Idle.</p> <p>BGP goes back to Idle when the hold timer expires, a disconnect notification is received from the transport protocol, a stop event is received, or in response to any other event.</p>

See Also For more detailed BGP protocol packet information, configure BGP-specific tracing. See “Track Error Conditions” on page 273 for more information.

Step 7: Display a Log File

Action To look at a log or trace file, use the following JUNOS CLI operational mode command:

```
user@host> show log filename
```

Sample Output

```
user@host> show log messages
Sep 10 07:00:00 host newsyslog[7249]: logfile turned over
Sep 10 07:01:49 host rpd[6451]: bgp_listen_accept: Connection attempt from
unconfigured neighbor: 10.0.8.1+1348
Sep 10 07:04:17 host rpd[6451]: bgp_listen_accept: Connection attempt from
unconfigured neighbor: 10.0.8.1+1349
Sep 10 07:06:45 host rpd[6451]: bgp_listen_accept: Connection attempt from
unconfigured neighbor: 10.0.8.1+1350
Sep 10 07:07:53 host login: 2 LOGIN FAILURES FROM 172.24.16.21
Sep 10 07:07:53 host login: 2 LOGIN FAILURES FROM 172.24.16.21, show
configuration | no-more
```

```

Sep 10 07:08:25 host inetd[2785]: /usr/libexec/telnetd[7251]: exit status 0x100
Sep 10 07:09:13 host rpd[6451]: bgp_listen_accept: Connection attempt from
unconfigured neighbor: 10.0.8.1+1351
Sep 10 07:11:41 host rpd[6451]: bgp_listen_accept: Connection attempt from
unconfigured neighbor: 10.0.8.1+1352
Sep 10 07:14:09 host rpd[6451]: bgp_listen_accept: Connection attempt from
unconfigured neighbor: 10.0.8.1+1353
Sep 10 07:16:37 host rpd[6451]: bgp_listen_accept: Connection attempt from
unconfigured neighbor: 10.0.8.1+1354
Sep 10 07:19:05 host rpd[6451]: bgp_listen_accept: Connection attempt from
unconfigured neighbor: 10.0.8.1+1355
Sep 10 07:21:33 host rpd[6451]: bgp_listen_accept: Connection attempt from
unconfigured neighbor:

```

What It Means The sample output shows the rpd log messages in the `messages` file for September 10 from 7:00 to 7:21 AM.



NOTE: Local log files are saved in the `/var/log` directory.

Step 8: Monitor Messages in Near-Real Time

Action To monitor messages in near-real time as they are being written to the log file, use the following JUNOS CLI operational mode command:

```
user@host> monitor start filename
```

Sample Output

```

user@host> monitor start messages
*** messages ***
Sep 10 19:46:30 router rpd[6451]: bgp_listen_accept: Connection attempt from
unconfigured neighbor: 10.0.8.1+1658

```

What It Means The sample output shows the routing protocol log messages in the `messages` file for September 10.

Step 9: Stop Monitoring Log Files

Action To stop monitoring log files, use the following JUNOS CLI operational mode command:

```
user@host> monitor stop filename
```

or

```
user@host> monitor stop
```


Chapter 23

Track Error Conditions

This chapter describes how to configure routing protocol daemon tracing, Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (IS-IS) protocol, and Open Shortest Path First (OSPF) protocol tracing to diagnose error conditions. (See Table 52.)

Table 52: Checklist for Tracking Error Conditions

Track Error Conditions Tasks	Command or Action
Configure Routing Protocol Process Tracing on page 275	
1. Configure Routing Protocol Process Tracing on page 275	[edit] edit routing-options traceoptions set file <i>filename</i> size <i>size</i> files <i>number</i> show commit run show log <i>filename</i>
2. Configure Routing Protocol Tracing for a Specific Routing Protocol on page 278	[edit] edit protocol <i>protocol-name</i> traceoptions set file <i>filename</i> size <i>size</i> files <i>number</i> show commit run show log <i>filename</i>
3. Monitor Trace File Messages Written in Near-Real Time on page 279	monitor start <i>filename</i>
4. Stop Trace File Monitoring on page 280	monitor stop <i>filename</i>
Configure BGP-Specific Options on page 281	
1. Display Detailed BGP Protocol Information on page 281	[edit] edit protocol bgp traceoptions set flag update detail show commit run show log <i>filename</i>
2. Display Sent or Received BGP Packets on page 283	[edit] edit protocol bgp traceoptions set flag update (send receive) show commit run show log <i>filename</i>

Track Error Conditions Tasks	Command or Action
3. Diagnose BGP Session Establishment Problems on page 284	[edit] edit protocol bgp set traceoptions flag open detail show commit run show log <i>filename</i>
Configure IS-IS-Specific Options on page 286	
1. Display Detailed IS-IS Protocol Information on page 286	[edit] edit protocol isis traceoptions set flag hello detail show commit run show log <i>filename</i>
2. Display Sent or Received IS-IS Protocol Packets on page 289	[edit] edit protocols isis traceoptions set flag hello (send receive) show commit run show log <i>filename</i>
3. Analyze IS-IS-Link State Packets in Detail on page 291	[edit] edit protocols isis traceoptions set flag lsp detail show commit run show log <i>filename</i>
Configure OSPF-Specific Options on page 293	
1. Diagnose OSPF Session Establishment Problems on page 293	[edit] edit protocols ospf traceoptions set flag hello detail show commit run show log <i>filename</i>
2. Analyze OSPF Link-State Advertisement Packets in Detail on page 297	[edit] edit protocols ospf traceoptions set flag lsa update detail show commit run show log <i>filename</i>

Configure Routing Protocol Process Tracing

Purpose Routing protocol process (rpd) tracing tracks all general routing operations and records them in a log file.

Steps To Take To configure routing protocol process (rpd) tracing and monitor trace file messages, follow these steps:

1. Configure Routing Protocol Process Tracing on page 275
2. Configure Routing Protocol Tracing for a Specific Routing Protocol on page 278
3. Monitor Trace File Messages Written in Near-Real Time on page 279
4. Stop Trace File Monitoring on page 280

Step 1: Configure Routing Protocol Process Tracing

Action To configure routing protocol process (rpd) tracing, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit routing-options traceoptions
```

2. Configure the file, file size, number, and flags:

```
[edit routing-options traceoptions]
user@host# set file filename size size files number
[edit routing-options traceoptions]
user@host# set flag flag
```

For example:

```
[edit routing-options traceoptions]
user@host# set file daemonlog size 10240 files 10
[edit routing-options traceoptions]
user@host# set flag general
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit routing-options traceoptions]
user@host# show
file daemonlog size 10k files 10;
flag general;
```

4. Commit the configuration:

```
user@host# commit
```



NOTE: Some traceoptions flags generate an extensive amount of information. Tracing can also slow down the operation of routing protocols. Delete the traceoptions configuration if you no longer require it.

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
[edit routing-options traceoptions]
user@pro4-a# run show log daemonlog
Sep 17 14:17:31 trace_on: Tracing to "/var/log/daemonlog" started
Sep 17 14:17:31 Tracing flags enabled: general
Sep 17 14:17:31 inet_routerid_notify: Router ID: 10.255.245.44
Sep 17 14:17:31 inet_routerid_notify: No Router ID assigned
Sep 17 14:17:31 Initializing LSI globals
Sep 17 14:17:31 LSI initialization complete
Sep 17 14:17:31 Initializing OSPF instances
Sep 17 14:17:31 Reinitializing OSPFv2 instance master
Sep 17 14:17:31 OSPFv2 instance master running
[...Output truncated...]
```

What It Means Table 53 lists tracing flags and example output for JUNOS-supported routing protocol daemon tracing.

Table 53: Routing Protocol Daemon Tracing Flags

Tracing Flag	Description	Example Output
all	All operations	Not available.
general	Normal operations and routing table change	Not available.
normal	Normal operations	Not available.

Tracing Flag	Description	Example Output
policy	Policy operations and actions	Nov 29 22:19:58 export: Dest 10.0.0.0 proto Static Nov 29 22:19:58 policy_match_qual_or: Qualifier proto Sense: 0 Nov 29 22:19:58 policy_match_qual_or: Qualifier proto Sense: 0 Nov 29 22:19:58 export: Dest 10.10.10.0 proto IS-IS
route	Routing table changes	Nov 29 22:23:59 Nov 29 22:23:59 rtlist_walker_job: rt_list walk for RIB inet.0 started with 42 entries Nov 29 22:23:59 rt_flash_update_callback: flash KRT (inet.0) start Nov 29 22:23:59 rt_flash_update_callback: flash KRT (inet.0) done Nov 29 22:23:59 rtlist_walker_job: rt_list walk for inet.0 ended with 42 entries Nov 29 22:23:59 Nov 29 22:23:59 KRT Request: send len 68 v14 seq 0 CHANGE route/user af 2 addr 172.16.0.0 nhop-type unicast nhop 10.10.10.33 Nov 29 22:23:59 KRT Request: send len 68 v14 seq 0 ADD route/user af 2 addr 172.17.0.0 nhop-type unicast nhop 10.10.10.33 Nov 29 22:23:59 KRT Request: send len 68 v14 seq 0 ADD route/user af 2 addr 10.149.3.0 nhop-type unicast nhop 10.10.10.33 Nov 29 22:24:19 trace_on: Tracing to "/var/log/rpdlog" started Nov 29 22:24:19 KRT Request: send len 68 v14 seq 0 DELETE route/user af 2 addr 10.10.218.0 nhop-type unicast nhop 10.10.10.29 Nov 29 22:24:19 RELEASE 10.10.218.0 255.255.255.0 gw 10.10.10.29,10.10.10.33 BGP pref 170/-101 metric so-1/1/0.0,so-1/1/1.0 <Release Delete Int Ext> as 65401 Nov 29 22:24:19 KRT Request: send len 68 v14 seq 0 DELETE route/user af 2 addr 172.18.0.0 nhop-type unicast nhop 10.10.10.33
state	State transitions	Not available.
task	Interface transactions and processing	Nov 29 22:50:04 foreground dispatch running job task_collect for task Scheduler Nov 29 22:50:04 task_collect_job: freeing task MGMT_Listen (DELETED) Nov 29 22:50:04 foreground dispatch completed job task_collect for task Scheduler Nov 29 22:50:04 background dispatch running job rt_static_update for task RT Nov 29 22:50:04 task_job_delete: delete background job rt_static_update for task RT Nov 29 22:50:04 background dispatch completed job rt_static_update for task RT Nov 29 22:50:04 background dispatch running job Flash update for task RT Nov 29 22:50:04 background dispatch returned job Flash update for task RT Nov 29 22:50:04 background dispatch running job Flash update for task RT Nov 29 22:50:04 task_job_delete: delete background job Flash update for task RT Nov 29 22:50:04 background dispatch completed job Flash update for task RT Nov 29 22:50:04 background dispatch running job Flash update for task RT Nov 29 22:50:04 task_job_delete: delete background job Flash update for task RT
timer	Timer usage	Nov 29 22:52:07 task_timer_hiprio_dispatch: ran 1 timer Nov 29 22:52:07 main: running normal priority timer queue Nov 29 22:52:07 main: ran 1 timer Nov 29 22:52:07 task_timer_hiprio_dispatch: running high priority timer queue Nov 29 22:52:07 task_timer_hiprio_dispatch: ran 1 timer Nov 29 22:52:07 main: running normal priority timer queue Nov 29 22:52:07 main: ran 1 timer Nov 29 22:52:07 main: running normal priority timer queue Nov 29 22:52:07 main: ran 2 timers

Step 2: Configure Routing Protocol Tracing for a Specific Routing Protocol

Action To configure routing protocol tracing for a specific routing protocol, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocol protocol-name traceoptions
```

2. Configure the file, file size, number, and flags:

```
[edit protocols protocol name traceoptions]
user@host# set file filename size size files number
[edit protocols protocol name traceoptions]
user@host# set flag flag
```

For example:

```
[edit protocols ospf traceoptions]
user@host# set file ospflog size 10240 files 10
[edit protocols ospf traceoptions]
user@host# set flag general
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit protocols ospf traceoptions]
user@host# show
file ospflog size 10k files 10;
flag general;
```

4. Commit the configuration:

```
user@host# commit
```

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
[edit protocols ospf traceoptions]
user@pro4-a# run show log ospflog
Sep 17 14:23:10 trace_on: Tracing to "/var/log/ospflog" started
Sep 17 14:23:10 rt_flash_update_callback: flash OSPF (inet.0) start
Sep 17 14:23:10 OSPF: multicast address 224.0.0.5/32, route ignored
Sep 17 14:23:10 rt_flash_update_callback: flash OSPF (inet.0) done
Sep 17 14:23:10 CHANGE 10.255.245.46/32 gw 10.10.208.67 OSPF
pref 10/0 metric 1/0 fe-0/0/0.0 <Delete Int>
Sep 17 14:23:10 CHANGE 10.255.245.46/32 gw 10.10.208.67 OSPF
pref 10/0 metric 1/0 fe-0/0/0.0 <Active Int>
Sep 17 14:23:10 ADD 10.255.245.46/32 gw 10.10.208.67 OSPF
```

```

pref 10/0 metric 1/0 fe-0/0/0.0 <Active Int>
Sep 17 14:23:10 CHANGE 10.255.245.48/32 gw 10.10.208.69 OSPF
pref 10/0 metric 1/0 fe-0/0/0.0 <Delete Int>
Sep 17 14:23:10 CHANGE 10.255.245.48/32 gw 10.10.208.69 OSPF
pref 10/0 metric 1/0 fe-0/0/0.0 <Active Int>
Sep 17 14:23:10 ADD 10.255.245.48/32 gw 10.10.208.69 OSPF
pref 10/0 metric 1/0 fe-0/0/0.0 <Active Int>
Sep 17 14:23:10 rt_close: 4/4 routes proto OSPF
[...Output truncated...]

```

What It Means Table 54 lists standard tracing options that are available globally or that can be applied to specific protocols. You can also configure tracing for a specific BGP peer or peer group. For more information, see the *JUNOS System Basics Configuration Guide*.

Table 54: Standard Trace Options for Routing Protocols

Tracing Flag	Description
all	All operations
general	Normal operations and routing table changes
normal	Normal operations
policy	Policy operations and actions
route	Routing table changes
state	State transitions
task	Interface transactions and processing
timer	Timer usage

Step 3: Monitor Trace File Messages Written in Near-Real Time

Action To monitor messages in near-real time as they are being written to a trace file, use the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> monitor start filename
```

Sample Output user@host> monitor start isis

```

user@host>
*** isis ***
Sep 15 18:32:21 Updating LSP isis5.02-00 in database
Sep 15 18:32:21 Updating L2 LSP isis5.02-00 in TED
Sep 15 18:32:21 Adding a half link from isis5.02 to isis6.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis5.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis6.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis5.00
Sep 15 18:32:21 Scheduling L2 LSP isis5.02-00 sequence 0xd87 on interface fxp2.3
Sep 15 18:32:21 Updating LSP isis5.00-00 in database
Sep 15 18:32:21 Updating L1 LSP isis5.00-00 in TED
Sep 15 18:32:21 Sending L2 LSP isis5.02-00 on interface fxp2.3
Sep 15 18:32:21     sequence 0xd87, checksum 0xc1c8, lifetime 1200

```

Step 4: Stop Trace File Monitoring

Action To stop monitoring a trace file in near-real time, use the following JUNOS CLI operational mode command after you have started monitoring:

monitor stop *filename*

Sample Output user@host> monitor start isis

```
user@host>
*** isis ***
Sep 15 18:32:21 Updating LSP isis5.02-00 in database
Sep 15 18:32:21 Updating L2 LSP isis5.02-00 in TED
Sep 15 18:32:21 Adding a half link from isis5.02 to isis6.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis5.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis6.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis5.00
Sep 15 18:32:21 Scheduling L2 LSP isis5.02-00 sequence 0xd87 on interface fxp2.3
Sep 15 18:32:21 Updating LSP isis5.00-00 in database
Sep 15 18:32:21 Updating L1 LSP isis5.00-00 in TED
Sep 15 18:32:21 Sending L2 LSP isis5.02-00 on interface fxp2.3
Sep 15 18:32:21      sequence 0xd87, checksum 0xc1c8, lifetime 1200
monitor stop isis

user@host>
```


Configure BGP-Specific Options

Purpose When unexpected events or problems occur, or if you want to diagnose BGP establishment issues, you can view more detailed information by configuring options specific to BGP. You can also configure tracing for a specific BGP peer or peer group. For more information, see the *JUNOS System Basics Configuration Guide*.

- Steps To Take**
1. Display Detailed BGP Protocol Information on page 281
 2. Display Sent or Received BGP Packets on page 283
 3. Diagnose BGP Session Establishment Problems on page 284

Step 1: Display Detailed BGP Protocol Information

Action To display BGP protocol information in detail, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocol bgp traceoptions
```

2. Configure the flag to display detailed BGP protocol messages:

```
[edit protocols bgp traceoptions]
user@host# set flag update detail
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit protocols bgp traceoptions]
user@host# show
flag update detail;
```

4. Commit the configuration:

```
user@host# commit
```

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
[edit protocols bgp traceoptions]
user@pro5-a# run show log bgp
Sep 17 14:47:16 trace_on: Tracing to "/var/log/bgp" started
Sep 17 14:47:17 bgp_read_v4_update: receiving packet(s) from
10.255.245.53 (Internal AS 10458)
Sep 17 14:47:17 BGP RECV 10.255.245.53+179 -> 10.255.245.50+1141
Sep 17 14:47:17 BGP RECV message type 2 (Update) length 128
Sep 17 14:47:17 BGP RECV flags 0x40 code Origin(1): IGP
Sep 17 14:47:17 BGP RECV flags 0x40 code ASPath(2): 2
Sep 17 14:47:17 BGP RECV flags 0x80 code MultiExitDisc(4): 0
Sep 17 14:47:17 BGP RECV flags 0x40 code LocalPref(5): 100
Sep 17 14:47:17 BGP RECV flags 0xc0 code Extended Communities(16):
2:10458:1
[...Output truncated...]
```

What It Means Table 55 lists tracing flags specific to BGP and presents example output for some of the flags. You can also configure tracing for a specific BGP peer or peer group. For more information, see the *JUNOS System Basics Configuration Guide*.

Table 55: BGP Protocol Tracing Flags

Tracing Flags	Description	Example Output
aspath	AS path regular expression operations	Not available.
damping	Damping operations	Nov 28 17:01:12 bgp_damp_change: Change event Nov 28 17:01:12 bgp_dampen: Damping 10.10.1.0 Nov 28 17:01:12 bgp_damp_change: Change event Nov 28 17:01:12 bgp_dampen: Damping 10.10.2.0 Nov 28 17:01:12 bgp_damp_change: Change event Nov 28 17:01:12 bgp_dampen: Damping 10.10.3.0
keepalive	BGP keepalive messages	Nov 28 17:09:27 bgp_send: sending 19 bytes to 10.217.5.101 (External AS 65471) Nov 28 17:09:27 Nov 28 17:09:27 BGP SEND 10.217.5.1+179 -> 10.217.5.101+52162 Nov 28 17:09:27 BGP SEND message type 4 (KeepAlive) length 19 Nov 28 17:09:28 Nov 28 17:09:28 BGP RECV 10.217.5.101+52162 -> 10.217.5.1+179 Nov 28 17:09:28 BGP RECV message type 4 (KeepAlive) length 19
open	BGP open packets	Nov 28 18:37:42 bgp_send: sending 37 bytes to 10.217.5.101 (External AS 65471) Nov 28 18:37:42 Nov 28 18:37:42 BGP SEND 10.217.5.1+179 -> 10.217.5.101+38135 Nov 28 18:37:42 BGP SEND message type 1 (Open) length 37

Tracing Flags	Description	Example Output
packets	All BGP protocol packets	<pre>Sep 27 17:45:31 BGP RECV 10.0.100.108+179 -> 10.0.100.105+1033 Sep 27 17:45:31 BGP RECV message type 4 (KeepAlive) length 19 Sep 27 17:45:31 bgp_send: sending 19 bytes to 10.0.100.108 (Internal AS 100) Sep 27 17:45:31 BGP SEND 10.0.100.105+1033 -> 10.0.100.108+179 Sep 27 17:45:31 BGP SEND message type 4 (KeepAlive) length 19 Sep 27 17:45:31 bgp_read_v4_update: receiving packet(s) from 10.0.100.108 (Internal AS 100)</pre>
update	Update packets	<pre>Nov 28 19:05:24 BGP SEND 10.217.5.1+179 -> 10.217.5.101+55813 Nov 28 19:05:24 BGP SEND message type 2 (Update) length 53 Nov 28 19:05:24 bgp_send: sending 65 bytes to 10.217.5.101 (External AS 65471) Nov 28 19:05:24 Nov 28 19:05:24 BGP SEND 10.217.5.1+179 -> 10.217.5.101+55813 Nov 28 19:05:24 BGP SEND message type 2 (Update) length 65 Nov 28 19:05:24 bgp_send: sending 55 bytes to 10.217.5.101 (External AS 65471)</pre>

Step 2: Display Sent or Received BGP Packets

Action To configure the tracing for sent or received BGP protocol packets, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocol bgp traceoptions
```

2. Configure the flag to display sent, received, or both sent and received packet information:

```
[edit protocols bgp traceoptions]
user@host# set flag update send
```

or

```
[edit protocols bgp traceoptions]
user@host# set flag update receive
```

or

```
[edit protocols bgp traceoptions]
user@host# set flag update
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit protocols bgp traceoptions]
user@host# show
file bgplog size 10k files 10;
flag update send;
```

or

```
[edit protocols bgp traceoptions]
```

```
user@host# show
file bgplog size 10k files 10;
flag update receive;
```

or

```
[edit protocols bgp traceoptions]
user@host# show
file bgplog size 10k files 10;
flag update send receive;
```

4. Commit the configuration:

```
user@host# commit
```

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
[edit protocols bgp traceoptions]
user@host# run show log bgplog
Sep 13 12:58:23 trace_on: Tracing to "/var/log/bgplog" started
Sep 13 12:58:23 BGP RECV flags 0x40 code ASPath(2): <null>
Sep 13 12:58:23 BGP RECV flags 0x40 code LocalPref(5): 100
Sep 13 12:58:23 BGP RECV flags 0xc0 code Extended Communities(16):
2:10458:3
[...Output truncated...]
```

Step 3: Diagnose BGP Session Establishment Problems

Action To trace BGP session establishment problems, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocol bgp
```

2. Configure BGP open messages:

```
[edit protocols bgp]
user@host# set traceoptions flag open detail
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit protocols bgp]
user@host# show
traceoptions {
  file bgplog size 10k files 10;
  flag open detail;
```

4. Commit the configuration:

```
user@host# commit
```

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
[edit protocols bgp]
user@host# run show log bgplog
Sep 17 17:13:14 trace_on: Tracing to "/var/log/bgplog" started
Sep 17 17:13:14 bgp_read_v4_update: done with 201.0.0.2 (Internal AS
10458) received 19 octets 0 updates 0 routes
Sep 17 17:13:15 bgp_read_v4_update: receiving packet(s) from 201.0.0.3
(Internal AS 10458)
Sep 17 17:13:15 bgp_read_v4_update: done with 201.0.0.3 (Internal AS
10458) received 19 octets 0 updates 0 routes
Sep 17 17:13:44 bgp_read_v4_update: receiving packet(s) from 201.0.0.2
(Internal AS 10458)
[...Output truncated...]
```

Configure IS-IS-Specific Options

Purpose When unexpected events or problems occur, or if you want to diagnose IS-IS adjacency establishment issues, you can view more detailed information by configuring options specific to IS-IS.

Steps To Take To configure IS-IS options, follow these steps:

1. Display Detailed IS-IS Protocol Information on page 286
2. Display Sent or Received IS-IS Protocol Packets on page 289
3. Analyze IS-IS-Link State Packets in Detail on page 291

Step 1: Display Detailed IS-IS Protocol Information

Action To trace IS-IS messages in detail, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocols isis traceoptions
```

2. Configure the flag to display detailed IS-IS protocol messages:

```
[edit protocols isis traceoptions]
user@host# set flag hello detail
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit protocols isis traceoptions]
user@host# show
file isislog size 10k files 10;
flag hello detail;
```

4. Commit the configuration:

```
user@host# commit
```

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
user@host# run show log isislog
Nov 29 23:17:50 trace_on: Tracing to "/var/log/isislog" started
Nov 29 23:17:50 Sending PTP IIH on so-1/1/1.0
Nov 29 23:17:53 Sending PTP IIH on so-1/1/0.0
Nov 29 23:17:54 Received PTP IIH, source id abc-core-01 on so-1/1/0.0
Nov 29 23:17:54   from interface index 11
Nov 29 23:17:54   max area 0, circuit type I2, packet length 4469
Nov 29 23:17:54   hold time 30, circuit id 6
Nov 29 23:17:54   neighbor state up
Nov 29 23:17:54   speaks IP
Nov 29 23:17:54   area address 99.0008 (1)
Nov 29 23:17:54   IP address 10.10.10.29
Nov 29 23:17:54   4396 bytes of total padding
Nov 29 23:17:54   updating neighbor abc-core-01
Nov 29 23:17:55 Received PTP IIH, source id abc-core-02 on so-1/1/1.0
Nov 29 23:17:55   from interface index 12
Nov 29 23:17:55   max area 0, circuit type I2, packet length 4469
Nov 29 23:17:55   hold time 30, circuit id 6
Nov 29 23:17:55   neighbor state up
Nov 29 23:17:55   speaks IP
Nov 29 23:17:55   area address 99.0000 (1)
Nov 29 23:17:55   IP address 10.10.10.33
Nov 29 23:17:55   4396 bytes of total padding
Nov 29 23:17:55   updating neighbor abc-core-02
```

What It Means Table 56 lists tracing flags that can be configured specific to IS-IS and presents example output for some of the flags.

Table 56: IS-IS Protocol Tracing Flags

Tracing Flags	Description	Example Output
csn	Complete sequence number PDU (CSNP)	Nov 28 20:02:48 Sending L2 CSN on interface so-1/1/0.0 Nov 28 20:02:48 Sending L2 CSN on interface so-1/1/1.0 With the detail option. Nov 28 20:06:08 Sending L2 CSN on interface so-1/1/1.0 Nov 28 20:06:08 LSP abc-core-01.00-00 lifetime 1146 Nov 28 20:06:08 sequence 0x1c4f8 checksum 0xa1e9 Nov 28 20:06:08 LSP abc-core-02.00-00 lifetime 411 Nov 28 20:06:08 sequence 0x7435 checksum 0x5424 Nov 28 20:06:08 LSP abc-brdr-01.00-00 lifetime 465 Nov 28 20:06:08 sequence 0xf73 checksum 0xab10 Nov 28 20:06:08 LSP abc-edge-01.00-00 lifetime 1089 Nov 28 20:06:08 sequence 0x1616 checksum 0xdb29 Nov 28 20:06:08 LSP abc-edge-02.00-00 lifetime 1103 Nov 28 20:06:08 sequence 0x45cc checksum 0x6883
hello	Hello packet	Nov 28 20:13:50 Sending PTP IIH on so-1/1/1.0 Nov 28 20:13:50 Received PTP IIH, source id abc-core-01 on so-1/1/0.0 Nov 28 20:13:53 Received PTP IIH, source id abc-core-02 on so-1/1/1.0 Nov 28 20:13:57 Sending PTP IIH on so-1/1/0.0 Nov 28 20:13:58 Received PTP IIH, source id abc-core-01 on so-1/1/0.0 Nov 28 20:13:59 Sending PTP IIH on so-1/1/1.0
lsp	Link-state PDU (LSP) packets	Nov 28 20:15:46 Received L2 LSP abc-edge-01.00-00, interface so-1/1/0.0 Nov 28 20:15:46 from abc-core-01 Nov 28 20:15:46 sequence 0x1617, checksum 0xd92a, lifetime 1197 Nov 28 20:15:46 Updating L2 LSP abc-edge-01.00-00 in TED Nov 28 20:15:47 Received L2 LSP abc-edge-01.00-00, interface so-1/1/1.0 Nov 28 20:15:47 from abc-core-02 Nov 28 20:15:47 sequence 0x1617, checksum 0xd92a, lifetime 1197
lsp-generation	LSP generation packets	Nov 28 20:21:24 Regenerating L1 LSP abc-edge-03.00-00, old sequence 0x682 Nov 28 20:21:27 Rebuilding L1, fragment abc-edge-03.00-00 Nov 28 20:21:27 Rebuilt L1 fragment abc-edge-03.00-00, size 59 Nov 28 20:31:52 Regenerating L2 LSP abc-edge-03.00-00, old sequence 0x689 Nov 28 20:31:54 Rebuilding L2, fragment abc-edge-03.00-00 Nov 28 20:31:54 Rebuilt L2 fragment abc-edge-03.00-00, size 256 Nov 28 20:34:05 Regenerating L1 LSP abc-edge-03.00-00, old sequence 0x683 Nov 28 20:34:08 Rebuilding L1, fragment abc-edge-03.00-00 Nov 28 20:34:08 Rebuilt L1 fragment abc-edge-03.00-00, size 59
packets	All IS-IS protocol packets	Not available.

Tracing Flags	Description	Example Output
psn	Partial sequence number PDU (PSNP) packets	Nov 28 20:40:39 Received L2 PSN, source abc-core-01, interface so-1/1/0.0 Nov 28 20:40:39 Received L2 PSN, source abc-core-02, interface so-1/1/1.0 Nov 28 20:41:36 Sending L2 PSN on interface so-1/1/1.0 Nov 28 20:41:36 Sending L2 PSN on interface so-1/1/0.0 Nov 28 20:42:35 Received L2 PSN, source abc-core-02, interface so-1/1/1.0 Nov 28 20:42:35 LSP abc-edge-03.00-00 lifetime 1196 Nov 28 20:42:35 sequence 0x68c checksum 0x746d Nov 28 20:42:35 Received L2 PSN, source abc-core-01, interface so-1/1/0.0 Nov 28 20:42:35 LSP abc-edge-03.00-00 lifetime 1196 Nov 28 20:42:35 sequence 0x68c checksum 0x746d Nov 28 20:42:49 Sending L2 PSN on interface so-1/1/1.0 Nov 28 20:42:49 LSP abc-core-01.00-00 lifetime 1197 Nov 28 20:42:49 sequence 0x1c4fb checksum 0x9bec Nov 28 20:42:49 Sending L2 PSN on interface so-1/1/0.0 Nov 28 20:42:49 LSP abc-core-01.00-00 lifetime 1197 Nov 28 20:42:49 sequence 0x1c4fb checksum 0x9bec
spf	Shortest path first (SPF) calculations	Nov 28 20:44:01 Scheduling SPF for L1: Reconfig Nov 28 20:44:01 Scheduling multicast SPF for L1: Reconfig Nov 28 20:44:01 Scheduling SPF for L2: Reconfig Nov 28 20:44:01 Scheduling multicast SPF for L2: Reconfig Nov 28 20:44:02 Running L1 SPF Nov 28 20:44:02 L1 SPF initialization complete: 0.000099s cumulative time Nov 28 20:44:02 L1 SPF primary processing complete: 0.000303s cumulative time Nov 28 20:44:02 L1 SPF result postprocessing complete: 0.000497s cumulative time Nov 28 20:44:02 L1 SPF RIB postprocessing complete: 0.000626s cumulative time Nov 28 20:44:02 L1 SPF routing table postprocessing complete: 0.000736s cumulative time

Step 2: Display Sent or Received IS-IS Protocol Packets

Action To configure the tracing for only sent or received IS-IS protocol packets, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocol isis traceoptions
```

2. Configure the flag to display sent, received, or both sent and received packets:

```
[edit protocols isis traceoptions]
user@host# set flag hello send
```

or

```
[edit protocols isis traceoptions]
user@host# set flag hello receive
```

or

```
[edit protocols isis traceoptions]
user@host# set flag hello
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit protocols isis traceoptions]  
user@host# show  
file isislog size 10k files 10;  
flag hello send;
```

or

```
[edit protocols isis traceoptions]  
user@host# show  
file isislog size 10k files 10;  
flag hello receive;
```

or

```
[edit protocols isis traceoptions]  
user@host# show  
file isislog size 10k files 10;  
flag hello send receive;
```

4. Commit the configuration:

```
user@host# commit
```

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
user@host# run show log isislog  
Sep 27 18:17:01 ISIS periodic xmit to 01:80:c2:00:00:15 (IFL 2)  
Sep 27 18:17:01 ISIS periodic xmit to 01:80:c2:00:00:14 (IFL 2)  
Sep 27 18:17:03 ISIS periodic xmit to 01:80:c2:00:00:15 (IFL 2)  
Sep 27 18:17:04 ISIS periodic xmit to 01:80:c2:00:00:14 (IFL 2)  
Sep 27 18:17:06 ISIS L2 hello from 0000.0000.0008 (IFL 2) absorbed  
Sep 27 18:17:06 ISIS periodic xmit to 01:80:c2:00:00:15 (IFL 2)  
Sep 27 18:17:06 ISIS L1 hello from 0000.0000.0008 (IFL 2) absorbed
```

Step 3: Analyze IS-IS Link State Packets in Detail

Action To analyze IS-IS link-state packets (LSPs) in detail, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocols isis traceoptions
```

2. Configure IS-IS open messages:

```
[edit protocols isis traceoptions]
user@host# set flag lsp detail
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit protocols isis traceoptions]
user@host# show
file isislog size 5m world-readable;
flag error;
flag lsp detail;
```

4. Commit the configuration:

```
user@host# commit
```

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
user@host# run show log isislog
Nov 28 20:17:24 Received L2 LSP abc-core-01.00-00, interface so-1/1/0.0
Nov 28 20:17:24   from abc-core-01
Nov 28 20:17:24   sequence 0x1c4f9, checksum 0x9fea, lifetime 1199
Nov 28 20:17:24   max area 0, length 426
Nov 28 20:17:24   no partition repair, no database overload
Nov 28 20:17:24   IS type 3, metric type 0
Nov 28 20:17:24   area address 99.0908 (1)
Nov 28 20:17:24   speaks CLNP
Nov 28 20:17:24   speaks IP
Nov 28 20:17:24   dyn hostname abc-core-01
Nov 28 20:17:24   IP address 10.10.134.11
Nov 28 20:17:24   IP prefix: 10.10.10.0/30 metric 1 up
Nov 28 20:17:24   IP prefix: 10.10.10.4/30 metric 5 up
Nov 28 20:17:24   IP prefix: 10.10.10.56/30 metric 5 up
Nov 28 20:17:24   IP prefix: 10.10.10.52/30 metric 1 up
Nov 28 20:17:24   IP prefix: 10.10.10.64/30 metric 5 up
Nov 28 20:17:24   IP prefix: 10.10.10.20/30 metric 5 up
Nov 28 20:17:24   IP prefix: 10.10.10.28/30 metric 5 up
Nov 28 20:17:24   IP prefix: 10.10.10.44/30 metric 5 up
```

```

Nov 28 20:17:24 IP prefix 10.10.10.0 255.255.255.252
Nov 28 20:17:24 internal, metrics: default 1
Nov 28 20:17:24 IP prefix 10.10.10.4 255.255.255.252
Nov 28 20:17:24 internal, metrics: default 5
Nov 28 20:17:24 IP prefix 10.10.10.56 255.255.255.252
Nov 28 20:17:24 internal, metrics: default 5
Nov 28 20:17:24 IP prefix 10.10.10.52 255.255.255.252
Nov 28 20:17:24 internal, metrics: default 1
Nov 28 20:17:24 IP prefix 10.10.10.64 255.255.255.252
Nov 28 20:17:24 internal, metrics: default 5
Nov 28 20:17:24 IP prefix 10.10.10.20 255.255.255.252
Nov 28 20:17:24 internal, metrics: default 5
Nov 28 20:17:24 IP prefix 10.10.10.28 255.255.255.252
Nov 28 20:17:24 internal, metrics: default 5
Nov 28 20:17:24 IP prefix 10.10.10.44 255.255.255.252
Nov 28 20:17:24 internal, metrics: default 5
Nov 28 20:17:24 IS neighbors:
Nov 28 20:17:24 IS neighbor abc-core-02.00
Nov 28 20:17:24 internal, metrics: default 1
[...Output truncated...]
Nov 28 20:17:24 internal, metrics: default 5
Nov 28 20:17:24 IS neighbor abc-brdr-01.00
Nov 28 20:17:24 internal, metrics: default 5
Nov 28 20:17:24 IS neighbor abc-core-02.00, metric: 1
Nov 28 20:17:24 IS neighbor abc-esr-02.00, metric: 5
Nov 28 20:17:24 IS neighbor abc-edge-03.00, metric: 5
Nov 28 20:17:24 IS neighbor abc-edge-01.00, metric: 5
Nov 28 20:17:24 IS neighbor abc-edge-02.00, metric: 5
Nov 28 20:17:24 IS neighbor abc-brdr-01.00, metric: 5
Nov 28 20:17:24 IP prefix: 10.10.134.11/32 metric 0 up
Nov 28 20:17:24 IP prefix: 10.11.0.0/16 metric 5 up
Nov 28 20:17:24 IP prefix: 10.211.0.0/16 metric 0 up
Nov 28 20:17:24 IP prefix 10.10.134.11 255.255.255.255
Nov 28 20:17:24 internal, metrics: default 0
Nov 28 20:17:24 IP prefix 10.11.0.0 255.255.0.0
Nov 28 20:17:24 internal, metrics: default 5
Nov 28 20:17:24 IP prefix 10.211.0.0 255.255.0.0
Nov 28 20:17:24 internal, metrics: default 0
Nov 28 20:17:24 Updating LSP
Nov 28 20:17:24 Updating L2 LSP abc-core-01.00-00 in TED
Nov 28 20:17:24 Analyzing subtlv's for abc-core-02.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Analyzing subtlv's for abc-esr-02.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Analyzing subtlv's for abc-edge-03.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Analyzing subtlv's for abc-edge-01.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Analyzing subtlv's for abc-edge-02.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Analyzing subtlv's for abc-brdr-01.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Scheduling L2 LSP abc-core-01.00-00 sequence 0x1c4f9
on interface so-1/1/1.0

```

Configure OSPF-Specific Options

Purpose When unexpected events or problems occur, or if you want to diagnose OSPF neighbor establishment issues, you can view more detailed information by configuring options specific to OSPF.

Steps To Take To configure OSPF options, follow these steps:

1. Diagnose OSPF Session Establishment Problems on page 293
2. Analyze OSPF Link-State Advertisement Packets in Detail on page 297

Step 1: Diagnose OSPF Session Establishment Problems

Action To trace OSPF messages in detail, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocols ospf traceoptions
```

2. Configure OSPF hello messages:

```
[edit protocols ospf traceoptions]
user@host# set flag hello detail
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit protocols ospf traceoptions]
user@host# show
file ospf size 5m world-readable;
flag hello detail;
```

4. Commit the configuration:

```
user@host# commit
```

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
user@host# run show log ospf
Dec 2 16:14:24 OSPF sent Hello (1) -> 224.0.0.5 (so-1/1/3.0)
Dec 2 16:14:24 Version 2, length 44, ID 10.0.0.6, area 1.0.0.0
Dec 2 16:14:24 checksum 0xf01a, authtype 0
Dec 2 16:14:24 mask 0.0.0.0, hello_ivl 10, opts 0x2, prio 128
Dec 2 16:14:24 dead_ivl 40, DR 0.0.0.0, BDR 0.0.0.0
Dec 2 16:14:24 OSPF sent Hello (1) -> 224.0.0.5 (so-1/1/2.0)
Dec 2 16:14:24 Version 2, length 44, ID 10.0.0.6, area 1.0.0.0
Dec 2 16:14:24 checksum 0xf01a, authtype 0
Dec 2 16:14:24 mask 0.0.0.0, hello_ivl 10, opts 0x2, prio 128
Dec 2 16:14:24 dead_ivl 40, DR 0.0.0.0, BDR 0.0.0.0
Dec 2 16:14:26 OSPF rcvd Hello 10.10.10.33 -> 224.0.0.5 (so-1/1/1.0)
Dec 2 16:14:26 Version 2, length 48, ID 10.10.134.12, area 0.0.0.0
Dec 2 16:14:26 checksum 0x99b8, authtype 0
Dec 2 16:14:26 mask 255.255.255.252, hello_ivl 10, opts 0x2, prio 1
Dec 2 16:14:26 dead_ivl 40, DR 0.0.0.0, BDR 0.0.0.0
Dec 2 16:14:29 OSPF rcvd Hello 10.10.10.29 -> 224.0.0.5 (so-1/1/0.0)
Dec 2 16:14:29 Version 2, length 48, ID 10.108.134.11, area 0.0.0.0
Dec 2 16:14:29 checksum 0x99b9, authtype 0
Dec 2 16:14:29 mask 255.255.255.252, hello_ivl 10, opts 0x2, prio 1
Dec 2 16:14:29 dead_ivl 40, DR 0.0.0.0, BDR 0.0.0.0
```

What It Means Table 57 lists OSPF tracing flags and presents example output for some of the flags.

Table 57: OSPF Protocol Tracing Flags

Tracing Flags	Description	Example Output
database-description	All database description packets	<pre>Dec 2 15:44:51 RPD_OSPF_NBRDOWN: OSPF neighbor 10.10.10.29 (so-1/1/0.0) state changed from Full to Down Dec 2 15:44:51 RPD_OSPF_NBRDOWN: OSPF neighbor 10.10.10.33 (so-1/1/1.0) state changed from Full to Down Dec 2 15:44:55 RPD_OSPF_NBRUP: OSPF neighbor 10.10.10.33 (so-1/1/1.0) state changed from Init to ExStart Dec 2 15:44:55 OSPF sent DbD (2) -> 224.0.0.5 (so-1/1/1.0) Dec 2 15:44:55 Version 2, length 32, ID 10.0.0.6, area 0.0.0.0 Dec 2 15:44:55 checksum 0xf76b, authtype 0 Dec 2 15:44:55 options 0x42, i 1, m 1, ms 1, seq 0xa009eee, mtu 4470 Dec 2 15:44:55 OSPF rcvd DbD 10.10.10.33 -> 224.0.0.5 (so-1/1/1.0) Dec 2 15:44:55 Version 2, length 32, ID 10.10.134.12, area 0.0.0.0 Dec 2 15:44:55 checksum 0x312c, authtype 0 Dec 2 15:44:55 options 0x42, i 1, m 1, ms 1, seq 0x2154, mtu 4470</pre>
error	OSPF errored packets	<pre>Dec 2 15:49:34 OSPF packet ignored: no matching interface from 172.16.120.29 Dec 2 15:49:44 OSPF packet ignored: no matching interface from 172.16.120.29 Dec 2 15:49:54 OSPF packet ignored: no matching interface from 172.16.120.29 Dec 2 15:50:04 OSPF packet ignored: no matching interface from 172.16.120.29 Dec 2 15:50:14 OSPF packet ignored: no matching interface from 172.16.120.29</pre>

Tracing Flags	Description	Example Output
event	OSPF state transitions	Dec 2 15:52:35 OSPF interface ge-2/2/0.0 state changed from DR to DR Dec 2 15:52:35 OSPF interface ge-3/1/0.0 state changed from DR to DR Dec 2 15:52:35 OSPF interface ge-3/2/0.0 state changed from DR to DR Dec 2 15:52:35 OSPF interface ge-4/2/0.0 state changed from DR to DR Dec 2 15:53:21 OSPF neighbor 10.10.10.29 (so-1/1/0.0) state changed from Full to Down Dec 2 15:53:21 RPD_OSPF_NBRDOWN: OSPF neighbor 10.10.10.29 (so-1/1/0.0) state changed from Full to Down Dec 2 15:53:21 OSPF neighbor 10.10.10.33 (so-1/1/1.0) state changed from Full to Down Dec 2 15:53:21 RPD_OSPF_NBRDOWN: OSPF neighbor 10.10.10.33 (so-1/1/1.0) state changed from Full to Down Dec 2 15:53:25 OSPF neighbor 10.10.10.33 (so-1/1/1.0) state changed from Down to Init Dec 2 15:53:25 OSPF neighbor 10.10.10.33 (so-1/1/1.0) state changed from Init to ExStart Dec 2 15:53:25 RPD_OSPF_NBRUP: OSPF neighbor 10.10.10.33 (so-1/1/1.0) state changed from Init to ExStart Dec 2 15:53:25 OSPF neighbor 10.10.10.33 (so-1/1/1.0) state changed from ExStart to Exchange Dec 2 15:53:25 OSPF neighbor 10.10.10.33 (so-1/1/1.0) state changed from Exchange to Full Dec 2 15:53:25 RPD_OSPF_NBRUP: OSPF neighbor 10.10.10.33 (so-1/1/1.0) state changed from Exchange to Full
flooding	Link-state flooding packets	Dec 2 15:55:21 OSPF LSA Summary 10.218.0.0 10.0.0.6 flooding on so-1/1/0.0 Dec 2 15:55:21 OSPF LSA Summary 10.218.0.0 10.0.0.6 flooding on so-1/1/1.0 Dec 2 15:55:21 OSPF LSA Summary 10.218.0.0 10.0.0.6 on no so-1/1/2.0 retransmit lists, no flood Dec 2 15:55:21 OSPF LSA Summary 10.218.0.0 10.0.0.6 on no so-1/1/3.0 retransmit lists, no flood Dec 2 15:55:21 OSPF LSA Summary 10.245.0.1 10.0.0.6 on no so-1/1/2.0 retransmit lists, no flood Dec 2 15:55:21 OSPF LSA Summary 10.245.0.1 10.0.0.6 on no so-1/1/3.0 retransmit lists, no flood
hello	Hello packets	Dec 2 15:57:25 OSPF sent Hello (1) -> 224.0.0.5 (ge-3/1/0.0) Dec 2 15:57:25 Version 2, length 44, ID 10.0.0.6, area 2.0.0.0 Dec 2 15:57:25 checksum 0xe43f, authtype 0 Dec 2 15:57:25 mask 255.255.0.0, hello_intvl 10, opts 0x2, prio 128 Dec 2 15:57:25 dead_intvl 40, DR 10.218.0.1, BDR 0.0.0.0 Dec 2 15:57:25 OSPF rcvd Hello 10.10.10.33 -> 224.0.0.5 (so-1/1/1.0) Dec 2 15:57:25 Version 2, length 48, ID 10.10.134.12, area 0.0.0.0 Dec 2 15:57:25 checksum 0x99b8, authtype 0 Dec 2 15:57:25 mask 255.255.255.252, hello_intvl 10, opts 0x2, prio 1 Dec 2 15:57:25 dead_intvl 40, DR 0.0.0.0, BDR 0.0.0.0 Dec 2 15:57:27 OSPF sent Hello (1) -> 224.0.0.5 (ge-3/2/0.0) Dec 2 15:57:27 Version 2, length 44, ID 10.0.0.6, area 2.0.0.0 Dec 2 15:57:27 checksum 0xe4a5, authtype 0 Dec 2 15:57:27 mask 255.255.0.0, hello_intvl 10, opts 0x2, prio 128 Dec 2 15:57:27 dead_intvl 40, DR 10.116.0.1, BDR 0.0.0.0 Dec 2 15:57:28 OSPF rcvd Hello 10.10.10.29 -> 224.0.0.5 (so-1/1/0.0) Dec 2 15:57:28 Version 2, length 48, ID 10.10.134.11, area 0.0.0.0 Dec 2 15:57:28 checksum 0x99b9, authtype 0 Dec 2 15:57:28 mask 255.255.255.252, hello_intvl 10, opts 0x2, prio 1 Dec 2 15:57:28 dead_intvl 40, DR 0.0.0.0, BDR 0.0.0.0

Tracing Flags	Description	Example Output
lsa-ack	Link-state acknowledgment packets	Dec 2 16:00:11 OSPF rcvd LSAck 10.10.10.29 -> 224.0.0.5 (so-1/1/0.0) Dec 2 16:00:11 Version 2, length 44, ID 10.10.134.11, area 0.0.0.0 Dec 2 16:00:11 checksum 0xcdbf, authtype 0 Dec 2 16:00:11 OSPF rcvd LSAck 10.10.10.33 -> 224.0.0.5 (so-1/1/1.0) Dec 2 16:00:11 Version 2, length 144, ID 10.10.134.12, area 0.0.0.0 Dec 2 16:00:11 checksum 0x73bc, authtype 0 Dec 2 16:00:16 OSPF rcvd LSAck 10.10.10.33 -> 224.0.0.5 (so-1/1/1.0) Dec 2 16:00:16 Version 2, length 44, ID 10.10.134.12, area 0.0.0.0 Dec 2 16:00:16 checksum 0x8180, authtype 0
lsa-request	Link-state request packets	Dec 2 16:01:38 OSPF rcvd LSReq 10.10.10.29 -> 224.0.0.5 (so-1/1/0.0) Dec 2 16:01:38 Version 2, length 108, ID 10.10.134.11, area 0.0.0.0 Dec 2 16:01:38 checksum 0xe86, authtype 0
lsa-update	Link-state update packets	Dec 2 16:09:12 OSPF built router LSA, area 0.0.0.0 Dec 2 16:09:12 OSPF built router LSA, area 1.0.0.0 Dec 2 16:09:12 OSPF built router LSA, area 2.0.0.0 Dec 2 16:09:13 OSPF sent LSUpdate (4) -> 224.0.0.5 (so-1/1/0.0) Dec 2 16:09:13 Version 2, length 268, ID 10.0.0.6, area 0.0.0.0 Dec 2 16:09:13 checksum 0x8047, authtype 0 Dec 2 16:09:13 adv count 7 Dec 2 16:09:13 OSPF sent LSUpdate (4) -> 224.0.0.5 (so-1/1/1.0) Dec 2 16:09:13 Version 2, length 268, ID 10.0.0.6, area 0.0.0.0 Dec 2 16:09:13 checksum 0x8047, authtype 0 Dec 2 16:09:13 adv count 7
packets	All OSPF packets	Not available.
packet-dump	Dump the contents of selected packet types	Not available.
spf	SPF calculations	Dec 2 16:08:03 OSPF full SPF refresh scheduled Dec 2 16:08:04 OSPF SPF start, area 1.0.0.0 Dec 2 16:08:04 OSPF add LSA Router 10.0.0.6 distance 0 to SPF list Dec 2 16:08:04 SPF elapsed time 0.000525s Dec 2 16:08:04 Stub elapsed time 0.000263s Dec 2 16:08:04 OSPF SPF start, area 2.0.0.0 Dec 2 16:08:04 OSPF add LSA Router 10.0.0.6 distance 0 to SPF list Dec 2 16:08:04 SPF elapsed time 0.000253s Dec 2 16:08:04 Stub elapsed time 0.000249s Dec 2 16:08:04 OSPF SPF start, area 0.0.0.0 Dec 2 16:08:04 OSPF add LSA Router 10.0.0.6 distance 0 to SPF list Dec 2 16:08:04 OSPF add LSA Router 10.10.134.11 distance 1 to SPF list Dec 2 16:08:04 IP nexthop so-1/1/0.0 0.0.0.0 Dec 2 16:08:04 OSPF add LSA Router 10.10.134.12 distance 1 to SPF list Dec 2 16:08:04 IP nexthop so-1/1/1.0 0.0.0.0

Step 2: Analyze OSPF Link-State Advertisement Packets in Detail

Action To analyze OSPF link-state advertisement packets in detail, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocols ospf traceoptions
```

2. Configure OSPF link-state packages:

```
[edit protocols ospf traceoptions]
user@host# set flag lsa-update detail
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit protocols ospf traceoptions]
user@host# show
file ospf size 5m world-readable;
flag hello detail;
flag lsa-update detail;
```

4. Commit the configuration:

```
user@host# commit
```

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
user@host# run show log ospf
Dec 2 16:23:47 OSPF sent LSUpdate (4) -> 224.0.0.5 (so-1/1/0.0)
Dec 2 16:23:47 Version 2, length 196, ID 10.0.0.6, area 0.0.0.0
Dec 2 16:23:47 checksum 0xcc46, authtype 0
Dec 2 16:23:47 adv count 6
Dec 2 16:23:47 OSPF sent LSUpdate (4) -> 224.0.0.5 (so-1/1/1.0)
Dec 2 16:23:47 Version 2, length 196, ID 10.0.0.6, area 0.0.0.0
Dec 2 16:23:47 checksum 0xcc46, authtype 0
Dec 2 16:23:47 adv count 6
```


Chapter 24

Collect Crash Data

This chapter explains the crashes that can occur in different areas of the JUNOS software, and provides procedures you use to collect the crash data necessary for troubleshooting by the Juniper Networks Technical Assistance Center (JTAC). (See Table 58.)

Table 58: Checklist for Collecting Crash Data

Collecting Crash Data Tasks	Command or Action
Understand Crash Data Collection on page 301	
Collect Crash Data for a Routing Engine Kernel on page 302	
1. Check the Routing Engine Core Files on page 302	file list detail /var/crash
2. Collect and Send Crash Data to JTAC on page 303	
a. List the Core Files on page 303	start shell su root password cd /var/crash ls -l
b. Compress the vmcore File on page 304	gzip vmcore.number To unzip the vmcore file: gzip -d vmcore.number.gz
c. Log Software Version Information on page 304	show version
d. Open a Case with JTAC on page 305	support@juniper.net ftp ftp.juniper.net
Collect Crash Data for Routing Engine Daemons on page 306	
1. Check for Daemon Core Files on page 306	file list detail /var/tmp
2. Collect and Send Crash Data to JTAC on page 307	
a. List the Daemon Core Files on page 307	start shell su root password cd /var/tmp ls -l
b. Compress the Daemon Core Files on page 308	gzip daemon-executable-name.core.number
c. Log Software Version Information on page 308	show version
d. Open a Case with JTAC on page 309	support@juniper.net ftp ftp.juniper.net

Collecting Crash Data Tasks	Command or Action
Collect Crash Data for the Packet Forwarding Engine Microkernel on page 310	
1. Display the Crash Stack Traceback and Registration Information on page 311	start shell su <i>root password</i> vty <i>component-executable-name</i> show nvram show syslog messages
2. Clear the NVRAM Contents on page 314	start shell su root password vty <i>component-executable-name</i> clear nvram
3. Check Packet Forwarding Engine Microkernel Core Files on page 315	file list detail /var/crash
4. Collect and Send Crash Data to JTAC on page 315	
a. List the Core Files Generated by the Crash on page 315	start shell su <i>root password</i> cd /var/crash ls -l
b. Compress the Core Files on page 316	gzip <i>filename</i>
c. Log Software Version Information on page 316	show version
d. Open a Case with JTAC on page 317	support@juniper.net ftp ftp.juniper.net

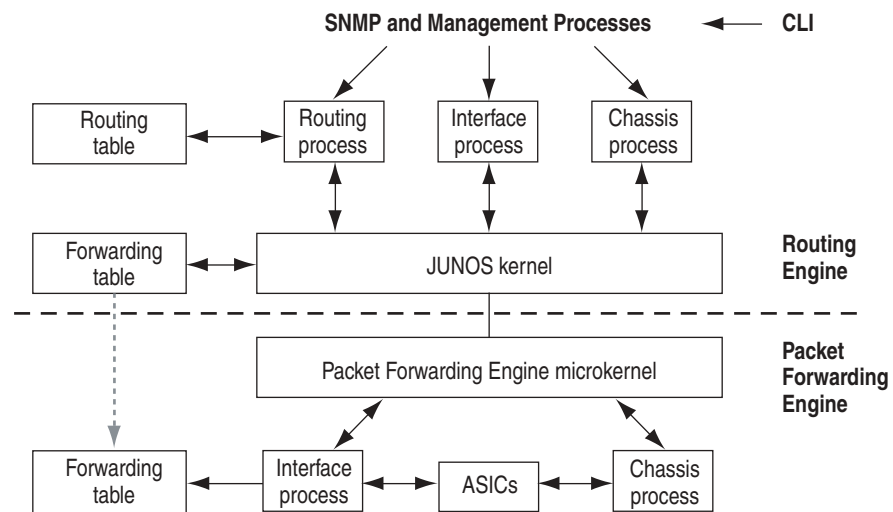
Understand Crash Data Collection

A crash can occur in one of three areas in the JUNOS software architecture (see Figure 22):

- Routing Engine kernel
- Routing Engine daemons (processes)
- Packet Forwarding Engine microkernel

Figure 22: Three Areas Where a Software Crash Can Occur

Software Architecture



1992

Collect Crash Data for a Routing Engine Kernel

Purpose When a Routing Engine kernel crashes, the Routing Engine automatically reboots. By default, the Juniper Networks router does not attempt to dump a core if the Routing Engine kernel crashes. As a result, there is no crash data on the router to help investigate the crash. In addition, the system log messages are similar to those generated when the router is powered down and restarted, so you cannot tell if the Routing Engine restart was caused by a kernel crash or a normal power restart.

Steps To Take To collect crash data for a Routing Engine kernel crash, follow these steps:

1. Check the Routing Engine Core Files on page 302
2. Collect and Send Crash Data to JTAC on page 303

Step 1: Check the Routing Engine Core Files

Purpose If you observe disruption to the Routing Engine kernel, check the `/var/crash` directory for any core files created around the time of the crash.

Action To check the `/var/crash` directory, use the following JUNOS command-line interface (CLI) operational mode command:

```
user@host> file list detail /var/crash
```

Sample Output

```
user@host> file list detail /var/crash
total 1577912
drwxr-x---  2 root  wheel           512 Sep  9 11:59 ./
drwxr-xr-x 22 root  wheel           512 Oct 29 2001 ../
-rw-r--r--  1 root  wheel             2 Jul 20 01:11 bounds
-rw-r--r--  1 root  wheel    2166913 Jul 20 01:11 kernel.0
-rw-r--r--  1 root  wheel             5 Feb 15 2002 minfree
-rw-----  1 root  wheel    805306368 Jul 20 01:11 vmcore.0
```

What It Means The sample output lists the contents of the `/var/crash/` directory. Check the date and timestamp for any kernel core files created around the time of the crash. In the example above, two core files are listed: `kernel.0` and `vmcore.0`.

Step 2: Collect and Send Crash Data to JTAC

Purpose If a Routing Engine kernel crash occurs on your router, collect the following data for JTAC evaluation and instruction.

Steps To Take To collect and send crash data to JTAC, follow these steps:

1. List the Core Files on page 303
2. Compress the vmcore File on page 304
3. Log Software Version Information on page 304
4. Open a Case with JTAC on page 305

List the Core Files

Action To list the core files, follow these steps:

1. Exit from the CLI environment and create a UNIX-level shell by entering the `start shell` command:

```
user@host> start shell
```

2. Type `su` and the root password when prompted. You are now in the shell and the prompt is `%` instead of `>`, for example:

```
% su
Password: ****
```

3. Change the directory to `/var/crash` and type `ls -l`, for example:

```
root@host% cd /var/crash
root@host% ls -l
```

4. Look for any core files created around the time of the crash.

Sample Output

```
user@host> start shell
% su
Password: ****
root@host% cd /var/crash
root@host% ls -l
total 1577908
-rw-r--r-- 1 root wheel          2 Jul 20 01:11 bounds
-rw-r--r-- 1 root wheel    2166913 Jul 20 01:11 kernel.0
-rw-r--r-- 1 root wheel         5 Feb 15 2002 minfree
-rw----- 1 root wheel 805306368 Jul 20 01:11 vmcore.0
```

What It Means The sample output lists the contents of the `/var/crash` directory and shows the current core files `kernel.0` and `vm.core.0`.

Compress the vmcore File

Purpose gzip is a compression utility used to compress files. Compress the **vmcore** file if it is larger than 50 MB. Files created using the **gzip** command end with the file extension **.gz**.



NOTE: Use lowercase for the **gzip** command when you are in the shell.

Action To compress the **vmcore** file with **gzip**, use the following command from the shell:

```
root@host% gzip vmcore.number
```

To unzip the **vmcore** file with **gzip**, use the following command from the shell:

```
root@host% gzip -d vmcore.number.gz
```

What It Means The contents of the **vmcore** file are compressed into a single compressed file named **vmcore.number.gz**. The **gzip** command preserves the mode, ownership, and timestamps of files when compressing or decompressing them.

Log Software Version Information

Action To log the JUNOS software version information, use the following JUNOS CLI operational mode command:

```
user@host> show version
```

Sample Output user@host> show version

```
Hostname: host
Model: m10
JUNOS Base OS boot [5.0R5]
JUNOS Base OS Software Suite [5.0R5]
JUNOS Kernel Software Suite [5.0R5]
JUNOS Routing Software Suite [5.0R5]
JUNOS Packet Forwarding Engine Support [5.0R5]
JUNOS Crypto Software Suite [5.0R5]
JUNOS Online Documentation [5.0R5]
KERNEL 5.0R5 #0 built by builder on 2002-03-02 05:10:28 UTC
MGD release 5.0R5 built by builder on 2002-03-02 04:45:32 UTC
CLI release 5.0R5 built by builder on 2002-03-02 04:44:22 UTC
CHASSISD release 5.0R5 built by builder on 2002-03-02 04:43:37 UTC
DCD release 5.0R5 built by builder on 2002-03-02 04:42:47 UTC
RPD release 5.0R5 built by builder on 2002-03-02 04:46:17 UTC
SNMPD release 5.0R5 built by builder on 2002-03-02 04:52:26 UTC
MIB2D release 5.0R5 built by builder on 2002-03-02 04:45:37 UTC
APSD release 5.0R5 built by builder on 2002-03-02 04:43:31 UTC
VRRPD release 5.0R5 built by builder on 2002-03-02 04:52:34 UTC
ALARM release 5.0R5 built by builder on 2002-03-02 04:43:24 UTC
PFED release 5.0R5 built by builder on 2002-03-02 04:46:06 UTC
CRAFTD release 5.0R5 built by builder on 2002-03-02 04:44:30 UTC
SAMPLED release 5.0R5 built by builder on 2002-03-02 04:52:20 UTC
ILMID release 5.0R5 built by builder on 2002-03-02 04:45:21 UTC
BPRELAYD release 5.0R5 built by builder on 2002-03-02 04:42:41 UTC
RMOPD release 5.0R5 built by builder on 2002-03-02 04:46:11 UTC
jkernel-dd release 5.0R5 built by builder on 2002-03-02 04:41:07 UTC
jroute-dd release 5.0R5 built by builder on 2002-03-02 04:41:21 UTC
```



```
jdocs-dd release 5.0R5 built by builder on 2002-03-02 04:39:11 UTC
```

What It Means The sample output shows the hostname, router model, and the different JUNOS software packages, processes, and documents.

Open a Case with JTAC

Action To open a case with JTAC, follow these steps:

1. Send an e-mail to support@juniper.net, and include the information from the `show version` command.
2. At the support engineer's request, ftp the `vmcore.number.gz` file to a case-number directory at [ftp.juniper.net](ftp://ftp.juniper.net). To ftp the core file to a directory, follow these steps:
 - a. At the shell prompt, enter `ftp ftp.juniper.net`.
 - b. At the name prompt, enter `anonymous`.
 - c. At the password prompt, enter your e-mail address as the password.
 - d. At the ftp prompt, enter the `cd pub/ incoming` string.
 - e. Enter the `mkdir case-number` command, where the *case-number* is the value of the case you opened with JTAC, for example, `1999-1231-9999`. If a directory has already been created, continue with the next step.
 - f. Enter the `cd case-number` command.
 - g. Enter the `binary` command so that the file transfer is in binary and not ASCII.
 - h. Enter the `put vmcore.0.gz` command.

Sample Output The following output is an example of copying a core file from the shell to an ftp directory at [ftp.juniper.net](ftp://ftp.juniper.net):

```
root@host% ftp ftp.juniper.net
Connected to colo-ftp.juniper.net.
220 colo-ftp.juniper.net FTP server (Version 6.00LS) ready.
Name (ftp.juniper.net: root): anonymous
331 Guest login ok, send your email address as password.
Password: ****
230 Guest login ok, access restrictions apply.
ftp> cd pub/ incoming
250 CWD command successful
ftp> mkdir 1999-1231-9999
257 MKD command successful.
ftp> cd 1999-1231-9999
250 CWD command successful.
ftp> bin
200 Type set to I.
ftp> put vmcore.0.gz
```

What It Means The sample output shows that there is a connection to `ftp.juniper.net`, that the login name and password were entered, and that the core file was successfully copied from the shell to an ftp directory at `ftp.juniper.net`.

Collect Crash Data for Routing Engine Daemons

Steps To Take To collect crash data for Routing Engine daemons, follow these steps:

1. Check for Daemon Core Files on page 306
2. Collect and Send Crash Data to JTAC on page 307

Step 1: Check for Daemon Core Files

Purpose If you observe disruption to routing protocol operation, system log operation, Simple Network Management Protocol (SNMP) operation, or other operations handled by Routing Engine daemons, check the `/var/tmp` directory for any daemon core files created around the time of the crash.

Action To check the `/var/tmp` directory, use the following JUNOS CLI operational mode command:

```
user@host> file list detail /var/tmp
```

Sample Output

```
user@host> file list detail /var/tmp
total 1292622
drwxrwxrwt  3 root  field      512 Dec 31 06:48 ./
drwxr-xr-x 21 root  field      512 Mar  5 1999 ../
-rw-rw----  1 root  field 119713792 Nov 17 21:58 rpd.core.0
-rw-rw----  1 root  field 120782848 Nov 17 22:12 rpd.core.1
```

What It Means The sample output lists the contents of the `/var/tmp/` directory. Look for any daemon core files created around the time of the crash. In the example above, two core files are listed: `rpdc.core.0` and `rpdc.core.1`.

Table 59 lists the major Routing Engine daemons supported by the JUNOS software.

Table 59: Major Routing Engine Daemons

Executable Name	Definition	Description
rpdc	Routing protocol daemon	Provides routing protocol intelligence (Border Gateway Protocol [BGP], Intermediate System-to-Intermediate System [ISIS], Open Shortest Path First [OSPF], and so on).
dcd	Device control daemon	Manages all interface devices.
mgd	Management daemon	Provides user configuration access to the system. The CLI is a client of <code>mgd</code> .
snmpd	Simple Network Management Protocol daemon	Provides remote network management information to the network management system.
chassisd	Chassis daemon	Monitors and manages Flexible PIC Concentrator (FPC) slots and other environmental components.
alarmd	Alarm daemon	Manages system alarm notifications.

Executable Name	Definition	Description
apsd	Automatic protection switching daemon	Provides SONET Automatic Protection Switching (APS) functionality.
sampled	Traffic sampling daemon	Gathers traffic sampling information.
vrrpd	Virtual Router Redundancy Protocol daemon	Provides Virtual Router Redundancy Protocol (VRRP) functionality.
syslogd	System log daemon	Manages the router system logging operation.
mib2d	MIB2 daemon	Management Information Base (MIB) subagent for MIB2.

Step 2: Collect and Send Crash Data to JTAC

Purpose If a Routing Engine daemon crash occurs on your router and a core file is created in the `/var/tmp` directory, collect the following data for JTAC evaluation and instruction.

Steps To Take To collect and send crash data to JTAC, follow these steps:

1. List the Daemon Core Files on page 307
2. Compress the Daemon Core Files on page 308
3. Log Software Version Information on page 308
4. Open a Case with JTAC on page 309

List the Daemon Core Files

Action To list the daemon core files, follow these steps:

1. Exit from the CLI environment and create a UNIX-level shell by entering the `start shell` command:

```
user@host> start shell
```

2. Type `su` and the root password when prompted. You are now in the shell and the prompt is `%` instead of `>`, for example:

```
% su
Password: ****
```

3. Change the directory to `/var/tmp` and type `ls -l`, for example:

```
root@host% cd /var/tmp
root@host% ls -l
```

4. Look for any daemon core files created around the time of the crash.

Sample Output

```
user@host> start shell
% su
Password: ****
root@host% cd /var/tmp
root@host% ls -l
total 1292618
```

```
-rw-rw---- 1 root field 119713792 Nov 17 21:58 rpd.core.0
-rw-rw---- 1 root field 120782848 Nov 17 22:12 rpd.core.1
```

What It Means The sample output lists the contents of the `/var/tmp` directory and shows the current core file (`rpd.core.1`) and one previous core file (`rpd.core.0`) for the routing protocol daemon (`rpd`). For each daemon, you can have a total of five core files in the `/var/tmp` directory: the current core file and the four previous core files numbered 0 through 4 (from oldest to newest).

Compress the Daemon Core Files

Purpose `gzip` is a compression utility used to compress the files if they are large. Files created using the `gzip` command end with the file extension `.gz`. Compress the core file if it is over 50 MB.



NOTE: Use lowercase for the `gzip` command when you are in the shell.

You only need to compress the daemon core files when the tarball file is not created.

Action To compress the daemon core file with `gzip`, use the following command from the shell:

```
root@host% gzip daemon-executable-name.core.number
```

Sample Output `root@host% gzip rpd.core.0`

What It Means The contents of the daemon core file are compressed into a single compressed file named `daemon.number.gz`. The `gzip` command preserves the mode, ownership, and timestamps of files when compressing or decompressing them.

Log Software Version Information

Action To log JUNOS software version information, use the following JUNOS CLI operational mode command:

```
user@host> show version
```

Sample Output

```
user@host> show version
Hostname: host
Model: m10
JUNOS Base OS boot [5.0R5]
JUNOS Base OS Software Suite [5.0R5]
JUNOS Kernel Software Suite [5.0R5]
JUNOS Routing Software Suite [5.0R5]
JUNOS Packet Forwarding Engine Support [5.0R5]
JUNOS Crypto Software Suite [5.0R5]
JUNOS Online Documentation [5.0R5]
KERNEL 5.0R5 #0 built by builder on 2002-03-02 05:10:28 UTC
MGD release 5.0R5 built by builder on 2002-03-02 04:45:32 UTC
CLI release 5.0R5 built by builder on 2002-03-02 04:44:22 UTC
CHASSISD release 5.0R5 built by builder on 2002-03-02 04:43:37 UTC
DCD release 5.0R5 built by builder on 2002-03-02 04:42:47 UTC
RPD release 5.0R5 built by builder on 2002-03-02 04:46:17 UTC
SNMPD release 5.0R5 built by builder on 2002-03-02 04:52:26 UTC
```

```

MIB2D release 5.0R5 built by builder on 2002-03-02 04:45:37 UTC
APSD release 5.0R5 built by builder on 2002-03-02 04:43:31 UTC
VRRPD release 5.0R5 built by builder on 2002-03-02 04:52:34 UTC
ALARM release 5.0R5 built by builder on 2002-03-02 04:43:24 UTC
PFED release 5.0R5 built by builder on 2002-03-02 04:46:06 UTC
CRAFTD release 5.0R5 built by builder on 2002-03-02 04:44:30 UTC
SAMPLED release 5.0R5 built by builder on 2002-03-02 04:52:20 UTC
ILMID release 5.0R5 built by builder on 2002-03-02 04:45:21 UTC
BPRELAYD release 5.0R5 built by builder on 2002-03-02 04:42:41 UTC
RMOPD release 5.0R5 built by builder on 2002-03-02 04:46:11 UTC
jkernel-dd release 5.0R5 built by builder on 2002-03-02 04:41:07 UTC
jroute-dd release 5.0R5 built by builder on 2002-03-02 04:41:21 UTC
jdocs-dd release 5.0R5 built by builder on 2002-03-02 04:39:11 UTC

```

What It Means The output shows the hostname, router model, and the different JUNOS software packages, processes, and documents.

Open a Case with JTAC

Action To open a case with JTAC, follow these steps:

1. Send an e-mail to support@juniper.net, and include the information from the `show version` command.
2. At the support engineer's request, ftp the `daemon.number.gz` file to a directory at `ftp.juniper.net`. To ftp the core file to a directory, follow these steps:
 - a. At the shell prompt, enter `ftp ftp.juniper.net`.
 - b. At the name prompt, enter `anonymous`.
 - c. At the password prompt, enter your e-mail address as the password.
 - d. At the ftp prompt, enter the `cd pub/ incoming` string.
 - e. Enter the `mkdir case-number` command, where the `case-number` is the value of the case you opened with JTAC, for example, `1999-1231-9999`. If a directory has already been created, continue with the next step.
 - f. Enter the `cd case-number` command.
 - g. Enter the `binary` command so that the file transfer is in binary and not ASCII.
 - h. Enter the `put daemon.core.number.gz` command.

Sample Output The following output is an example of copying a core file from the shell to an ftp directory at `ftp.juniper.net`:

```

root@host% ftp ftp.juniper.net
Connected to colo-ftp.juniper.net.
220 colo-ftp.juniper.net FTP server (Version 6.00LS) ready.
Name (ftp.juniper.net: root): anonymous
331 Guest login ok, send your email address as password.
Password: ****
230 Guest login ok, access restrictions apply.
ftp> cd pub/ incoming

```

```

250 CWD command successful
ftp> mkdir 1999-1231-9999
257 MKD command successful.
ftp> cd 1999-1231-9999
250 CWD command successful.
ftp> bin
200 Type set to I.
ftp> put rpd.4.gz

```

What It Means The sample output shows that there is a connection to `ftp.juniper.net`, that the login name and password were entered, and that the core file was successfully copied from the shell to an ftp directory at `ftp.juniper.net`.

Collect Crash Data for the Packet Forwarding Engine Microkernel

Purpose Each of the following Packet Forwarding Engine components of a Juniper Networks router runs a microkernel:

- Flexible PIC Concentrator (FPC) on M-series platforms except for the M5 and M10 Internet routers
- Gibson Flexible PIC Concentrator (GFPC) on T640 and T320 Internet routing nodes
- Switched Printed Mezzanine Board (SPMB) on T640 and T320 Internet routing nodes
- Forwarding Engine Board (FEB) on M5 and M10 Internet routers
- System Switching Board (SSB) on an M20 Internet router
- System Control Board (SCB) on an M40 Internet router
- Switching and Forwarding Module (SFM) on M160 and M40e Internet routers

When a crash occurs, crash stack traceback and registration information is placed into nonvolatile random access memory (NVRAM) on the different components. Table 60 shows where the NVRAM is located for the components for each router.

Table 60: NVRAM Location on the Microkernel of the Packet Forwarding Engine Components

Router Type	NVRAM Location
M5 and M10	FEB
M20	SSB and crash stack traceback and register information for the FPC
M40	SCB and crash stack traceback and register information for the FPC
M40e	FPC SFM
M160	FPC SFM

Router Type	NVRAM Location
T320	GFPC SPMB
T640	GFPC SPMB

Steps To Take To collect crash data for the Packet Forwarding Engine microkernel, follow these steps:

1. Display the Crash Stack Traceback and Registration Information on page 311
2. Clear the NVRAM Contents on page 314
3. Check Packet Forwarding Engine Microkernel Core Files on page 315
4. Collect and Send Crash Data to JTAC on page 315

Step 1: Display the Crash Stack Traceback and Registration Information

Action To display the crash stack traceback and registration information, follow these steps:

1. Exit from the CLI environment and create a UNIX-level shell by entering the **start shell** command:

```
user@host> start shell
```

2. Type **su** and the root password when prompted. You are now in the shell and prompt is **%** instead of **>**, for example:

```
% su
Password: ****
```

3. Establish a vty session to the appropriate component. Use the **vtty** command followed by the executable name for the component; for example, **scb**, **ssb0**, **ssb1**, **fpc0**, or **fpc1**:

```
root@host% vty sfm0
```



NOTE: For the M40e and M160 routers, you can also create a **cty** session to the components if the components are not online.

4. Type the **show nvram** command to view the NVRAM information.
5. Type the **show syslog messages** command to view the system log messages.

Sample Output 1

```
user@host> start shell
% su
Password: ****
root@host% vty sfm0
```

```
SFM platform (266Mhz PPC 603e processor, 64Mb memory, 512Kb flash)
```

```

SFM3(host vty)# show nvram
System NVRAM :
  4080 available bytes, 4080 used, 0 free
  Contents:

mpc106 machine check caused by error on the PCI Bus
mpc106 error detect register 1: 0x08, 2: 0x00
mpc106 error ack count = 0
mpc106 error address: 0x0a000000
mpc106 PCI bus error status register: 0x02
  mpc106 was the PCI master
  C/BE bits: I/O read [0b0010]
mpc106 error detection reg1: PCI cycle
mpc106 PCI status reg: parity error

System Exception: Vector/Code 0x00700, Signal 4
Event occurred at: Oct 26 13:32:40.952

Juniper Embedded Microkernel Version 4.2R1
Built by tlim on 2000-09-23 06:11:28 UTC
Copyright (C) 1998-2000, Juniper Networks, Inc.
All rights reserved.
Reason string: "Program Check"
Context: Thread (PFE Manager)

Registers:
R00: 0x06f5f81c R01: 0x06f5f9cc R02: 0x00003344 R03: 0x00000000
R04: 0x00008000 R05: 0x00000000 R06: 0x0010052c R07: 0x06f637e4
R08: 0x06f5f81c R09: 0x00169810 R10: 0x000000e8 R11: 0x00000001
R12: 0x00046cdf R13: 0xffffffff R14: 0xffffffff R15: 0xffffffff
R16: 0xffffffff R17: 0xffffffff R18: 0xffffffff R19: 0xffffffff
R20: 0xffffffff R21: 0xffffffff R22: 0xffffffff R23: 0xffffffff
R24: 0x00000003 R25: 0x00000000 R26: 0x00000001 R27: 0x0000fc78
R28: 0x00150000 R29: 0x0016c4b0 R30: 0x06f5eb7c R31: 0x97cb1d36
MSR: 0x0008b030 CTR: 0x000ac008 Link:0x06f5f81c SP: 0x06f5f9cc
CCR: 0x22200024 XER: 0x20000000 PC: 0x06f5f81c
DSISR: 0x00000000 DAR: 0xffffffff K_MSR: 0x00001030

Stack Traceback:
Frame 01: sp = 0x06f5f9cc, pc = 0x06f5f81c
Frame 02: sp = 0x06f5f9e4, pc = 0x000c7e28
Frame 03: sp = 0x06f5fa04, pc = 0x00026620

ROM NVRAM:
  0 available bytes, 0 used, 0 free

SFM3(host vty)# show syslog messages
Oct 26 12:02:05 router tnp_sfm_2 PFEMAN: sent Resync request to Master
Oct 26 12:02:07 router tnp_sfm_3 CM(3): Slot 1: On-line
Oct 26 12:02:07 router tnp_sfm_3 CM(3): Slot 2: On-line
Oct 26 12:02:07 router tnp_sfm_3 CM(3): Slot 6: On-line
Oct 26 12:02:07 router tnp_sfm_3 PFEMAN: sent Resync request to Master
Oct 26 12:05:58 router tnp_sfm_3 mpc106 machine check caused by error on the
PCI Bu
s
Oct 26 12:05:58 router tnp_sfm_3 mpc106 error detect register 1: 0x08,
2: 0x00
Oct 26 12:05:58 router tnp_sfm_3 mpc106 error ack count = 0
Oct 26 12:05:58 router tnp_sfm_3 mpc106 error address: 0x0a000000
Oct 26 12:05:58 router tnp_sfm_3 mpc106 PCI bus error status register: 0x02
Oct 26 12:05:58 router tnp_sfm_3 mpc106 was the PCI master
Oct 26 12:05:58 router tnp_sfm_3 C/BE bits: I/O read [0b0010]
Oct 26 12:05:58 router tnp_sfm_3 mpc106 error detection reg1: PCI cycle

```



```

Oct 26 12:05:58 router tnp_sfm_3 mpc106 PCI status reg: parity error
Oct 26 12:05:58 router tnp_sfm_3 AB
Oct 26 12:05:58 router tnp_sfm_3 last message repeated 7 times
Oct 26 12:05:58 router tnp_sfm_3 Registers:
Oct 26 12:05:58 router tnp_sfm_3 R00: 0x06f5f81c R01: 0x06f5f9cc
R02: 0x00003344 R0
3: 0x00000000
Oct 26 12:05:58 router tnp_sfm_3 R04: 0x00008000 R05: 0x00000000
R06: 0x0010052c R0
7: 0x06f637e4
Oct 26 12:05:58 router tnp_sfm_3 R08: 0x06f5f81c R09: 0x00169810
R10: 0x000003b4 R1
1: 0x00000001
Oct 26 12:05:58 router tnp_sfm_3 R12: 0x00017b97 R13: 0xffffffff
R14: 0xffffffff R1
5: 0xffffffff
Oct 26 12:05:58 router tnp_sfm_3 R16: 0xffffffff R17: 0xffffffff
R18: 0xffffffff R1
9: 0xffffffff
Oct 26 12:05:58 router tnp_sfm_3 R20: 0xffffffff R21: 0xffffffff
R22: 0xffffffff R2
3: 0xffffffff
Oct 26 12:05:58 router tnp_sfm_3 R24: 0x00000003 R25: 0x00000000
R26: 0x00000001 R2
7: 0x0000fc78
Oct 26 12:05:58 router tnp_sfm_3 R28: 0x00150000 R29: 0x0016c4b0
R30: 0x06f5eb7c R3
1: 0x97c9c35e
Oct 26 12:05:58 router tnp_sfm_3 MSR: 0x0008b030 CTR: 0x000ac008
Link:0x06f5f81c SP
: 0x06f5f9cc
Oct 26 12:05:58 router tnp_sfm_3 CCR: 0x22200024 XER: 0x20000000
PC: 0x06f5f81c
Oct 26 12:05:58 router tnp_sfm_3 DSISR: 0x00000000 DAR: 0xffffffff
K_MSR: 0x0000103
0

```

Sample Output 2 The following sample output is another example of displaying the crash stack traceback and registration information:

```

root@host% vty fpc1

FPC160 platform (PPC 603e processor, 32Mb memory, 512Kb flash)

FPC1(host vty)# show nvram
System NVRAM :
  4080 available bytes, 4080 used, 0 free
  Contents:
00000000 R06: 0x0000005c R07: 0x850400d0
R08: 0x00000000 R09: 0x00000020 R10: 0x00000000 R11: 0x00000129
R12: 0x00000000 R13: 0x00000000 R14: 0x4005009a R15: 0x20000260
R16: 0xc8828784 R17: 0x84212800 R18: 0xc0004c61 R19: 0x80005900
R20: 0x80206000 R21: 0x84000304 R22: 0xd0410180 R23: 0x8c2005ac
R24: 0x00000003 R25: 0x00000000 R26: 0x00000001 R27: 0x0000fc48
R28: 0x001d0000 R29: 0x00000001 R30: 0x00136bb8 R31: 0x00000000
MSR: 0x0000b030 CTR: 0x001331e0 Link:0x000308c8 SP: 0x01baba34
CCR: 0x42200020 XER: 0x00000000 PC: 0x000308cc
DSISR: 0x00000000 DAR: 0xffffffff K_MSR: 0x00001030

```

Stack Traceback:

```

Frame 01: sp = 0x01baba34, pc = 0x000308c8
Frame 02: sp = 0x01babac4, pc = 0x0002647c
Frame 03: sp = 0x01babad4, pc = 0x00026590
Frame 04: sp = 0x01babadc, pc = 0x00106fcc
Frame 05: sp = 0x01babafc, pc = 0x00026620

```

ROM NVRAM:

```
0 available bytes, 0 used, 0 free
```

FPC1(host vty)# show syslog messages

```

[0+00:00:00.780 LOG: Info] Version 4.0R5 by tlim on 2000-08-10 04:45:54 UTC
[0+00:00:00.780 LOG: Info] On-board NVRAM contains diagnostic information.
[0+00:00:03.175 LOG: Info] PFEMAN: Established connection to Master
[Jan 30 21:53:05.804 LOG: Info] SNTDP: Initial time of day set.

```

What It Means Sample output 1 and 2 show the stack trace from the microkernel crash. Save the output from the `show nvram` and `show syslog` commands so that you can send them to JTAC when you open a case.

Step 2: Clear the NVRAM Contents

Purpose Currently the storage area for the logs on the NVRAM is limited to 4 KB. You need to delete old NVRAM logs to make room for new ones.

Action To clear the content of the NVRAM after you have captured the necessary information, follow these steps:

1. Exit from the CLI environment and create a UNIX-level shell by entering the `start shell` command:

```
user@host> start shell
```

2. Type `su` and the root password when prompted. You are now in the shell and the prompt is `%` instead of `>`, for example:

```
% su
Password: ****
```

3. Establish a vty session to the appropriate component. Use the `vty` command followed by the abbreviation for the component, for example:

```
root@host% vty sfm0
```

The vty prompt will vary depending on the component, for example:

```
SFM3(host vty)#
FPC1(host vty)#
```

4. Type the `clear nvram` command, for example:

```
SFM3(host vty)# clear nvram
FPC1(host vty)# clear nvram
```

Step 3: Check Packet Forwarding Engine Microkernel Core Files

Purpose If you observe disruption to the Packet Forwarding Engine microkernel, check the `/var/crash` directory for any core files created around the time of the crash.

Action To check the `/var/crash` directory, use the following JUNOS CLI operational mode command:

```
user@host> file list detail /var/crash
```

Sample Output

```
user@host> file list detail /var/crash
var/crash:
total 456630
-rw-r--r--  1 root  wheel   6814720 Dec 18 08:03 core-FPC4.100111808032
-rw-r--r--  1 root  wheel   65613824 Dec 10 04:58 core-SCB.100111004570
-rw-r--r--  1 root  wheel   65613824 Dec 19 00:23 core-SCB.100111900221
-rw-r--r--  1 root  wheel   65545216 Feb  9 20:46 core-SCB.101010920452
```

What It Means The sample output lists the contents of the `/var/crash/` directory. Check the date and timestamp for any core files created around the time of the crash. In the example above, four core files are listed.

Step 4: Collect and Send Crash Data to JTAC

Purpose If a Packet Forwarding Engine crash occurs on your router, collect the following data for JTAC evaluation and instruction.

Steps To Take To collect and send crash data to JTAC, follow these steps:

1. List the Core Files Generated by the Crash on page 315
2. Compress the Core Files on page 316
3. Log Software Version Information on page 316
4. Open a Case with JTAC on page 317

List the Core Files Generated by the Crash

Action To list the core files, follow these steps:

1. Exit from the CLI environment and create a UNIX-level shell by entering the `start shell` command:

```
user@host> start shell
```

2. Type `su` and the root password when prompted. You are now in the shell and the prompt is `%` instead of `>`, for example:

```
% su
Password: ****
```

3. Change the directory to `/var/crash` and type `ls -l`, for example:

```
root@host% cd /var/crash
root@host% ls -l
```

4. Look for any core files created around the time of the crash.

Sample Output

```
user@host> start shell
% su
Password: ****
root@host% cd /var/crash
root@host% ls -l
total 456630
-rw-r--r--  1 root  wheel   6814720 Dec 18 08:03 core-FPC4.100111808032
-rw-r--r--  1 root  wheel   65613824 Dec 10 04:58 core-SCB.100111004570
-rw-r--r--  1 root  wheel   65613824 Dec 19 00:23 core-SCB.100111900221
-rw-r--r--  1 root  wheel   65545216 Feb  9 20:46 core-SCB.101010920452
```

What It Means The sample output shows the current core files for the different components on the router; for example, `core-FPC4.100111808032` and `core-SCB.100111004570`.

Compress the Core Files

Purpose `gzip` is a compression utility used to compress the core files. Files created using the `gzip` command end with the file extension `.gz`. Compress the core files if they are larger than 50 MB.

Action To compress the core files with `gzip`, use the following command from the shell:

```
root@host% gzip filename
```

Sample Output

```
root@host% gzip core-SCB.101010920452
```

What It Means The contents of the core file are compressed into a single compressed file named `core-SCB.10101092045.gz`. The `gzip` command preserves the mode, ownership, and timestamps of files when compressing or decompressing them.

Log Software Version Information

Action To log JUNOS software version information, use the following JUNOS CLI operational mode command:

```
user@host> show version
```

Sample Output

```
user@host> show version
Hostname: host
Model: m10
JUNOS Base OS boot [5.0R5]
JUNOS Base OS Software Suite [5.0R5]
JUNOS Kernel Software Suite [5.0R5]
JUNOS Routing Software Suite [5.0R5]
JUNOS Packet Forwarding Engine Support [5.0R5]
JUNOS Crypto Software Suite [5.0R5]
JUNOS Online Documentation [5.0R5]
KERNEL 5.0R5 #0 built by builder on 2002-03-02 05:10:28 UTC
MGD release 5.0R5 built by builder on 2002-03-02 04:45:32 UTC
CLI release 5.0R5 built by builder on 2002-03-02 04:44:22 UTC
```

```

CHASSISD release 5.0R5 built by builder on 2002-03-02 04:43:37 UTC
DCD release 5.0R5 built by builder on 2002-03-02 04:42:47 UTC
RPD release 5.0R5 built by builder on 2002-03-02 04:46:17 UTC
SNMPD release 5.0R5 built by builder on 2002-03-02 04:52:26 UTC
MIB2D release 5.0R5 built by builder on 2002-03-02 04:45:37 UTC
APSD release 5.0R5 built by builder on 2002-03-02 04:43:31 UTC
VRRPD release 5.0R5 built by builder on 2002-03-02 04:52:34 UTC
ALARMD release 5.0R5 built by builder on 2002-03-02 04:43:24 UTC
PFED release 5.0R5 built by builder on 2002-03-02 04:46:06 UTC
CRAFTD release 5.0R5 built by builder on 2002-03-02 04:44:30 UTC
SAMPLED release 5.0R5 built by builder on 2002-03-02 04:52:20 UTC
ILMID release 5.0R5 built by builder on 2002-03-02 04:45:21 UTC
BPRELAYD release 5.0R5 built by builder on 2002-03-02 04:42:41 UTC
RMOPD release 5.0R5 built by builder on 2002-03-02 04:46:11 UTC
jkernel-dd release 5.0R5 built by builder on 2002-03-02 04:41:07 UTC
jroute-dd release 5.0R5 built by builder on 2002-03-02 04:41:21 UTC
jdocs-dd release 5.0R5 built by builder on 2002-03-02 04:39:11 UTC

```

What It Means The sample output shows the hostname, router model, and the different JUNOS software packages, processes, and documents.

Open a Case with JTAC

Action To open a case with JTAC, follow these steps:

1. Send an e-mail to support@juniper.net, and include the following information:
 - Output from the `show nvram` command
 - Output from the `show syslog messages` command
 - Output from the `show version` command
2. At the support engineer's request, ftp the `core-component.number.gz` file to a directory at [ftp.juniper.net](ftp://ftp.juniper.net). To ftp the core file to a directory, follow these steps:
 - a. At the shell prompt, enter `ftp ftp.juniper.net`.
 - b. At the name prompt, enter `anonymous`.
 - c. At the password prompt, enter your e-mail address as the password.
 - d. At the ftp prompt, enter the `cd pub/ incoming` string.
 - e. Enter the `mkdir case-number` command, where the `case-number` is the value of the case you opened with JTAC, for example, `1999-1231-9999`. If a directory has already been created, continue with the next step.
 - f. Enter the `cd case-number` command.
 - g. Enter the `binary` command so that the file transfer is in binary and not ASCII.
 - h. Enter the `put core-component.number.gz` command.

Sample Output The following output is an example of copying a core file from the shell to an ftp directory at ftp.juniper.net:

```
root@host% ftp ftp.juniper.net
Connected to colo-ftp.juniper.net.
220 colo-ftp.juniper.net FTP server (Version 6.00LS) ready.
Name (ftp.juniper.net: root): anonymous
331 Guest login ok, send your email address as password.
Password: ****
230 Guest login ok, access restrictions apply.
ftp> cd pub/ incoming
250 CWD command successful
ftp> mkdir 1999-1231-9999
257 MKD command successful.
ftp> cd 1999-1231-9999
250 CWD command successful.
ftp> bin
200 Type set to I.
ftp> put core-fpc4.100111808032.gz
```

What It Means The sample output shows that there is a connection to ftp.juniper.net, that the login name and password were entered, and that the core file was successfully copied from the shell to an ftp directory at ftp.juniper.net.

Part 6

Appendix

- [Command-Line Interface Overview on page 321](#)

Appendix 1

Command-Line Interface Overview

This chapter provides an overview of the JUNOS software command-line interface (CLI). For more detailed information about using the JUNOS software CLI, see the *JUNOS System Basics Configuration Guide* and the *JUNOS System Basics and Services Command Reference*.

The CLI is the interface to the software that you use whenever you access the router—whether from the console or through a remote network connection. The CLI, which automatically starts after the router finishes booting, provides commands that you use to perform various tasks, including configuring the JUNOS software, and monitoring and troubleshooting the software, network connectivity, and the router hardware.

The CLI has two modes:

- CLI Operational Mode on page 322
- CLI Configuration Mode on page 327

CLI Operational Mode

In operational mode you enter commands to monitor and troubleshoot the software, network connectivity, and the router by entering commands. When you log in to the router and the CLI starts, you are at the top level of the CLI operational mode. At this level, there are several broad groups of CLI commands (see Table 61).

Table 61: CLI Operational Mode Top-Level Commands

Command	Description
clear	Clear statistics and protocol database information. Syntax: clear (arp bgp chassis firewall igmp interfaces isis ldp log mpls msdp multicast ospf pim rip route rsvp snmp system vrrp)
configure	Enter CLI configuration mode. Alternative commands: configure exclusive configure private
file	Perform file manipulation operations, such as copy, delete, list, rename, and show. Syntax: file (compare copy delete list rename show)
help	Provide help information. Syntax: help (reference topic)
monitor	Monitor a log file or interface traffic in real time. Syntax: monitor (start stop interface list traffic)
mtrace	Display trace information about a multicast path from a source to a receiver. Syntax: mtrace (from-source to-gateway monitor)
ping	Try to connect to a remote target.
pipe	Filter the output of an operational mode or configuration mode command. Syntax: (compare count display <detail inheritance xml> except pattern find pattern hold match pattern no-more resolve <full-names> save <i>filename</i> trim columns)
quit	Exit from the CLI to a UNIX shell.
request	Make system-level requests, such as stop or reboot the router, load software packages, and back up the router's file systems. Syntax: request system (reboot halt software snapshot)
restart	Restart the router software processes. Syntax: restart (fpc interface-control mib-process routing sampling sfm snmp soft)
set	Set CLI properties, the router's date and time, and the craft interface display text. Syntax: set (chassis cli date)
show	Show information about all aspects of the software, including interfaces and routing protocols. Syntax: (aps arp as-path bgp chassis cli configuration connections dvmrp firewall host igmp interfaces isis ldp log mpls msdpl multicast ntp ospf pfe pim policy rpl route rsvp sap snmp system task ted version vrrp)
ssh	Open a secure shell to another host.
start	Start a software process. Syntax: start shell
telnet	Start a telnet session to another host.
test	Run various diagnostic debugging commands. Syntax: test (configuration interface msdp policy)
traceroute	Trace the route to a remote host.

Use the CLI Operational Mode

This section describes how to use the CLI operational mode. You can do the following:

- Enter the CLI Operational Mode on page 323
- Get Help on Commands at a Hierarchy Level on page 323
- Get Help about Commands on page 324
- Have the CLI Complete Commands on page 325
- Use CLI Command Completion on page 326
- Display CLI Command History on page 326

Enter the CLI Operational Mode

To enter the JUNOS software CLI, use the following command:

```
% cli
```

You are in the CLI when you see the > prompt, which is preceded by a string that defaults to the name of the user and the name of the router. For example:

```
user@host>
```

Get Help on Commands at a Hierarchy Level

The CLI provides context-sensitive help at every level of the command hierarchy. The help information tells you which commands are available at the current level in the hierarchy and provides a brief description of each.

To get help while in the CLI, type ?. You do not need to press **Enter** after typing the question mark. You have the following options:

- If you type the question mark at the command-line prompt, the CLI lists the available commands and options.
- If you type the question mark after entering the complete name of a command or command option, the CLI lists the available commands and options, then redisplay the command names and options that you typed.
- If you type the question mark in the middle of a command name, the CLI lists possible command completions that match the letters you have entered so far, then redisplay the letters that you typed.

Get Help about Commands

To get help about operational mode CLI commands, you can do the following:

- List Top-Level Operational Mode CLI Commands on page 324
- List CLI Commands that Start with a Particular Letter on page 324
- List All Available Commands of a Particular Type on page 325

List Top-Level Operational Mode CLI Commands

To list all available commands at the top level of the CLI operational mode, use the following command (see Table 61):

```
user@host> ?
```

Possible completions:

clear	Clear information in the system
configure	Manipulate software configuration information
file	Perform file operations
help	Provide help information
mtrace	Trace mtrace packets from source to receiver.
monitor	Real-time debugging
ping	Ping a remote target
quit	Exit the management session
request	Make system-level requests
restart	Restart a software process
set	Set CLI properties, date, time, craft display text
show	Show information about the system
ssh	Open a secure shell to another host
start	Start a software process
telnet	Telnet to another host
test	Diagnostic debugging commands
traceroute	Trace the route to a remote host

```
user@host>
```

List CLI Commands that Start with a Particular Letter

To list all commands that start with the letter c, use the following CLI command:

```
user@host> c?
```

Possible completions:

clear	Clear information in the system
configure	Manipulate software configuration information

```
user@host> c
```

List All Available Commands of a Particular Type

To list all available clear commands, use the following CLI command:

```
user@host> clear ?
```

Possible completions:

arp	Clear address-resolution information
bgp	Clear BGP information
chassis	Clear chassis information
firewall	Clear firewall counters
igmp	Clear IGMP information
interfaces	Clear interface information
ilmi	Clear ILMI statistics information
isis	Clear IS-IS information
ldp	Clear LDP information
log	Clear contents of a log file
mpls	Clear MPLS information
msdp	Clear MSDP information
multicast	Clear Multicast information
ospf	Clear OSPF information
pim	Clear PIM information
rip	Clear RIP information
route	Clear routing table information
rsvp	Clear RSVP information
snmp	Clear SNMP information
system	Clear system status
vrrp	Clear VRRP statistics information

```
user@host> clear
```

Have the CLI Complete Commands

You do not always have to remember or type the full command or option name for the CLI to recognize it. To display all possible command or option completions, type the partial command followed by a question mark.

To complete a command or option that you have partially typed, press the **Tab** key or the spacebar. If the partially typed letters begin a string that uniquely identifies a command, the complete command name appears. Otherwise, a beep indicates that you have entered an ambiguous command, and the possible completions are displayed.

Command completion also applies to other strings, such as filenames and usernames. To display all possible values, type a partial string followed by a question mark. However, to complete these strings, press the **Tab** key; pressing the space bar does not work.

Use CLI Command Completion

To complete the `show interfaces` command, do the following:

```
user@host> show in<Spacebar>terfaces <Enter>
```

```
Physical interface: at-0/1/0, Enabled, Physical link is Up
Interface index: 11, SNMP ifIndex: 65
Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, SONET mode
Speed: OC12, Loopback: None, Payload scrambler: Enabled
Device flags   : Present Running
Link flags     : 0x01
[...Output truncated...]
```

To display a list of all log files whose names start with the string “messages,” and then display the contents of one of the files, do the following:

```
user@host> show log mes?
```

Possible completions:

<filename>	Log file to display
messages	Size: 1417052, Last changed: Mar 3 00:33
messages.0.gz	Size: 145575, Last changed: Mar 3 00:00
messages.1.gz	Size: 134253, Last changed: Mar 2 23:00
messages.10.gz	Size: 137022, Last changed: Mar 2 14:00
messages.2.gr	Size: 137112, Last changed: Mar 2 22:00
messages.3.gz	Size: 121633, Last changed: Mar 2 21:00
messages.4.gz	Size: 135715, Last changed: Mar 2 20:00
messages.5.gz	Size: 137504, Last changed: Mar 2 19:00
messages.6.gz	Size: 134591, Last changed: Mar 2 18:00
messages.7.gz	Size: 132670, Last changed: Mar 2 17:00
messages.8.gz	Size: 136596, Last changed: Mar 2 16:00
messages.9.gz	Size: 136210, Last changed: Mar 2 15:00

```
user@host> show log mes<Tab>sages.4<Tab>.gz<Enter>
Jan 15 21:00:00 myhost newsyslog[1381]: logfile turned over
[...Output truncated...]
```

Display CLI Command History

You can display a list of recent commands that you issued. To display the command history, use the `show cli history` command:

```
user@host> show cli history
```

```
03-03 01:00:50 -- show cli history
03-03 01:01:12 -- show interfaces terse
03-03 01:01:22 -- show interfaces lo0
03-03 01:01:44 -- show bgp next-hop-database
03-03 01:01:51 -- show cli history
```

By default, this command displays the last 100 commands issued in the CLI. Specify a number with the command to display that number of recent commands. For example:

```
user@host> show cli history 3
```

```
01:01:44 -- show bgp next-hop-database
01:01:51 -- show cli history
01:02:51 -- show cli history 3
```

CLI Configuration Mode

In configuration mode, you configure the JUNOS software by creating a hierarchy of configuration statements by using the CLI or by creating a text (ASCII) file that contains the statement hierarchy. (The statement hierarchy is identical in both the CLI and text configuration file.) You can configure all properties of the JUNOS software, including interfaces, general routing information, routing protocols, and user access, as well as several system hardware properties. When you have finished entering the configuration statements, you commit them, which activates the configuration on the router.

Table 62 explains each CLI configuration mode command. The commands are organized alphabetically.

Table 62: CLI Configuration Mode Commands

Command	Description
activate	Remove the inactive: tag from a statement, effectively reading the statement or identifier to the configuration. Statements or identifiers that have been activated take effect when you next issue the commit command. Syntax: activate (<i>statement</i> <i>identifier</i>)
annotate	Add comments to a configuration. You can add comments only at the current hierarchy level. Syntax: annotate <i>statement</i> " <i>comment-string</i> "
commit	Commit the set of changes to the database and cause the changes to take operational effect. Syntax: commit << at < <i>string</i> >> <and-quit> <check> <confirmed < <i>minutes</i> >> <synchronize>
copy	Make a copy of an existing statement in the configuration. Syntax: copy <i>existing-statement</i> to <i>new-statement</i>
deactivate	Add the inactive: tag to a statement, effectively commenting out the statement or identifier from the configuration. Statements or identifiers marked as inactive do not take effect when you issue the commit command. Syntax: deactivate (<i>statement</i> <i>identifier</i>)
delete	Delete a statement or identifier. All subordinate statements and identifiers contained within the specified statement path are deleted with it. Syntax: delete < <i>statement-path</i> > < <i>identifier</i> >
edit	Move inside the specified statement hierarchy. If the statement does not exist, it is created. Syntax: edit <i>statement-path</i>
exit	Exit the current level of the statement hierarchy, returning to the level prior to the last edit command, or exit from configuration mode. The quit and exit commands are synonyms. Syntax: exit < <i>configuration-mode</i> >
help	Display help about available configuration statements. Syntax: help (apropos topic reference) < <i>string</i> >
insert	Insert an identifier into an existing hierarchy. Syntax: insert < <i>statement-path</i> > <i>identifier1</i> (before after) <i>identifier2</i>
load	Load a configuration from an ASCII configuration file or from terminal input. Your current location in the configuration hierarchy is ignored when the load operation occurs. Syntax: load (replace merge override) (<i>filename</i> <i>terminal</i>)

Command	Description
quit	Exit the current level of the statement hierarchy, returning to the level prior to the last edit command, or exit from configuration mode. The quit and exit commands are synonyms. Syntax: quit <i><configuration-mode></i>
rename	Rename an existing configuration statement or identifier. Syntax: rename <i><statement-path> identifier1 to identifier2</i>
rollback	Return to a previously committed configuration. The software saves the last 10 committed configurations, including the rollback number, date, time, and name of the user who issued the commit configuration command. The currently operational JUNOS software configuration is stored in the file juniper.conf , and the last three committed configurations are stored in the files juniper.conf.1 , juniper.conf.2 , and juniper.conf.3 . These four files are located in the directory /config , which is on the router's flash drive. The remaining six previous committed configurations, the files juniper.conf.4 through juniper.conf.9 , are stored in the directory /var/db/config , which is on the router's hard disk. Syntax: rollback <i><number></i>
run	Run a top-level CLI command without exiting from configuration mode. Syntax: run <i>command</i>
save	Save the configuration to an ASCII file. The contents of the current level of the statement hierarchy (and below) are saved, along with the statement hierarchy containing it. This allows a section of the configuration to be saved, while fully specifying the statement hierarchy. Syntax: save <i>filename</i>
set	Create a statement hierarchy and set identifier values. This is similar to edit except that your current level in the hierarchy does not change. Syntax: set <i><statement-path> identifier</i>
show	Display the current configuration. Syntax: show <i><statement-path> <identifier></i>
status	Display the users currently editing the configuration.
top	Return to the top level of configuration command mode, which is indicated by the [edit] banner. Syntax: top <i><configuration-command></i>
up	Move up one level in the statement hierarchy. Syntax: up <i><number> <configuration-command></i>
update	Update a private database.

Configuration Statements and Identifiers

You configure all router properties by including statements in the configuration. A statement consists of a keyword, which is fixed text, and, optionally, an identifier. An identifier is an identifying name that you define, such as the name of an interface, or a username, which allows you and the CLI to discriminate among a collection of statements.

The following list shows the statements available at the top level of the configuration mode (that is, the trunk of the hierarchy tree). Table 63 on page 330 describes each statement.

user@host# **set ?**

Possible completions:

> accounting-options	Accounting data configuration
+ apply-groups	Groups from which to inherit configuration data
> chassis	Chassis configuration
> class-of-service	Class-of-service configuration
> firewall	Define a firewall configuration
> forwarding-options	Configure options to control packet sampling
> groups	Configuration groups
> interfaces	Interface configuration
> policy-options	Routing policy option configuration
> protocols	Routing protocol configuration
> routing-instances	Routing instance configuration
> routing-options	Protocol-independent routing option configuration
> snmp	Simple Network Management Protocol
> system	System parameters

An angle bracket (>) before the statement name indicates that it is a container statement and you can define other statements at levels below it.

If there is no angle bracket (>) before the statement name, the statement is a leaf statement; you cannot define other statements at hierarchy levels below it.

A plus sign (+) before the statement name indicates that it can contain a set of values. To specify a set, include the values in brackets. For example:

[edit]

```
user@host# set policy-options community my-as1-transit members [65535:10
65535:11]
```

In some statements, you can include an identifier. For some identifiers, such as interface names, you must specify the identifier in a precise format. For example, the interface name **so-0/0/0** refers to a SONET/SDH interface that is on the Flexible PIC Concentrator (FPC) in slot 0, in the first Physical Interface Card (PIC) location, and in the first port on the PIC. For other identifiers, such as interface descriptive text, policy, and firewall term names, you can specify any name, including special characters, spaces, and tabs.

You must enclose in quotation marks (double quotes) identifiers and any strings that include the following characters: space tab () [] { } ! @ # \$ % ^ & | ' = ?

Table 63 describes each top-level CLI configuration mode statement.

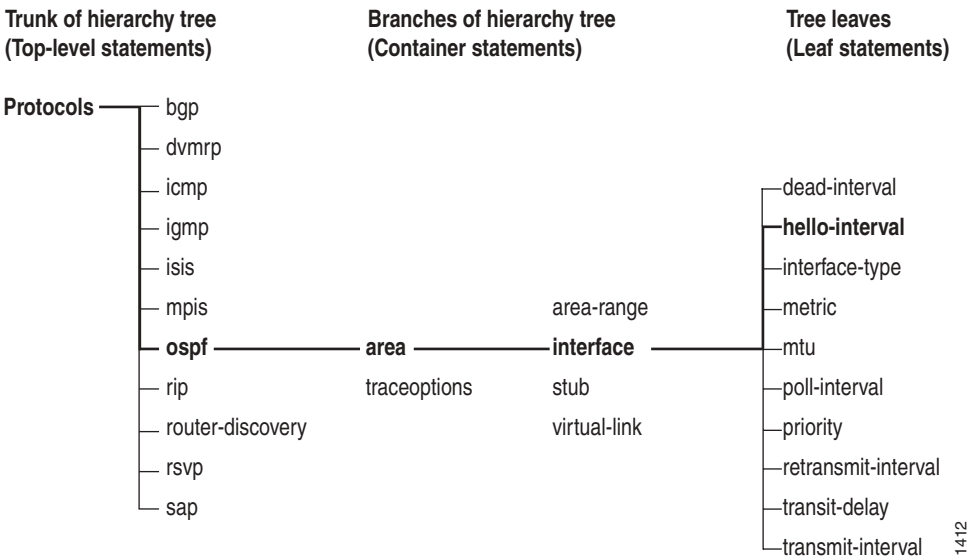
Table 63: Configuration Mode Top-Level Statements

Statement	Description
accounting-options	Configure accounting statistics data collection for interfaces and firewall filters. For information about the statements in this hierarchy, see the <i>JUNOS Network Management Configuration Guide</i> .
chassis	Configure properties of the router chassis, including the clock source, conditions that activate alarms, and SONET/SDH framing and concatenation properties. For information about the statements in this hierarchy, see the <i>JUNOS System Basics Configuration Guide</i> .
class-of-service	Configure class-of-service parameters. For information about the statements in this hierarchy, see the <i>JUNOS Class of Service Configuration Guide</i> .
firewall	Define filters that select packets based on their contents. For information about the statements in this hierarchy, see the <i>JUNOS Policy Framework Configuration Guide</i> .
forwarding-options	Define forwarding options, including traffic sampling options. For information about the statements in this hierarchy, see the <i>JUNOS Network Interfaces Configuration Guide</i> .
groups	Configure configuration groups. For information about statements in this hierarchy, see <i>JUNOS System Basics Configuration Guide</i> .
interfaces	Configure interface information, such as encapsulation, interfaces, virtual channel identifiers (VCIs), and data-link channel identifiers (DLCIs). For information about the statements in this hierarchy, see the <i>JUNOS Network Interfaces Configuration Guide</i> .
policy-options	Define routing policies, which allow you to filter and set properties in incoming and outgoing routes. For information about the statements in this hierarchy, see the <i>JUNOS Routing Protocols Configuration Guide</i> .
protocols	Configure routing protocols, including Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (IS-IS), Open Shortest Path First (OSPF), Routing Information Protocol (RIP), Multiprotocol Label Switching (MPLS), Label Distribution Protocol (LDP), and Resource Reservation Protocol (RSVP). For information about the statements in this hierarchy, see the chapters that discuss how to configure the individual routing protocols in the <i>JUNOS Routing Protocols Configuration Guide</i> and the <i>JUNOS MPLS Applications Configuration Guide</i> .
routing-instances	Configure multiple routing instances. For information about the statements in this hierarchy, see the <i>JUNOS Routing Protocols Configuration Guide</i> .
routing-options	Configure protocol-independent routing options, such as static routes, autonomous system (AS) numbers, confederation members, and global tracing (debugging) operations to log. For information about the statements in this hierarchy, see the <i>JUNOS Routing Protocols Configuration Guide</i> .
snmp	Configure Simple Network Management Protocol (SNMP) community strings, interfaces, traps, and notifications. For information about the statements in this hierarchy, see the <i>JUNOS Network Management Configuration Guide</i> .
system	Configure systemwide properties, including the hostname, domain name, Domain Name System (DNS) server, user logins and permissions, mappings between hostnames and addresses, and software processes.

Configuration Statement Hierarchy

The JUNOS software configuration consists of a hierarchy of *statements*. There are two types of statements: *container statements*, which are statements that contain other statements, and *leaf statements*, which do not contain other statements (see Figure 23). All of the container and leaf statements together form the *configuration hierarchy*.

Figure 23: Configuration Mode Hierarchy of Statements



Each statement at the top level of the configuration hierarchy resides at the trunk (or root level) of a hierarchy tree. The top-level statements are container statements, containing other statements that form the tree branches. The leaf statements are the leaves of the hierarchy tree. An individual hierarchy of statements, which starts at the trunk of the hierarchy tree, is called a *statement path*. Figure 23 illustrates the hierarchy tree, showing a statement path for the portion of the protocol configuration hierarchy that configures the hello interval on an interface in an OSPF area.

The **protocols** statement is a top-level statement at the trunk of the configuration tree. The **ospf**, **area**, and **interface** statements are all subordinate container statements of a higher statement (they are branches of the hierarchy tree), and the **hello-interval** statement is a leaf on the tree, which, in this case, contains a data value: the length of the hello interval in seconds.

The CLI represents the statement path shown in Figure 23 on page 331 as [`protocols ospf area area-number interface interface-name`], and displays the configuration as follows:

```

protocols {
  ospf {
    area 0.0.0.0 {
      interface so-0/0/0 {
        hello-interval 5;
      }
      interface so-0/0/1 {
        hello-interval 5;
      }
    }
  }
}

```

The CLI indents each level in the hierarchy to indicate each statement's relative position in the hierarchy and generally sets off each level with braces, using an open brace at the beginning of each hierarchy level and a closing brace at the end. If the statement at a hierarchy level is empty, the braces are not printed. Each leaf statement ends with a semicolon. If the hierarchy does not extend as far as a leaf statement, the last statement in the hierarchy ends with a semicolon.

The CLI uses this indented representation when it displays the current system configuration, and you use this format when creating ASCII files that contain the software configuration. However, the format of ASCII configuration files is not as strict as the CLI output of the configuration. Although the braces and semicolons are required, the indentation and use of new lines, as shown above, are not required in ASCII configuration files.

Use the CLI Configuration Mode

This section describes how to use the CLI configuration mode. You can do the following:

- Enter Configuration Mode on page 333
- Exit Configuration Mode on page 334
- Move among Levels of the Hierarchy on page 334
- Display the Current Configuration on page 335
- Modify the Configuration on page 336
- Remove a Statement on page 336
- Run Operational Mode CLI Commands from Configuration Mode on page 337

- Display Configuration Mode Command History on page 337
- Commit a Configuration on page 337
- Save a Configuration to a File on page 338
- Return to a Previously Committed Configuration on page 338
- Get Help about Statements on page 339

Enter Configuration Mode

If many users enter configuration mode at the same time, everyone can make configuration changes and commit all changes. If one user enters configuration mode when another user is also in configuration mode, a message indicates who the user is and what portion of the configuration that user is viewing or editing. To enter configuration mode, use the following CLI command:

```
user@host> configure
```

```
Entering configuration mode
```

```
Current configuration users:
```

```
  root terminal p3 (pid 1088) on since 1999-05-13 01:03:27 EDT
```

```
    [edit interfaces so-3/0/0 unit 0 family inet]
```

```
The configuration has been changed but not committed
```

- If, when you enter configuration mode, the configuration contains changes that have not been committed, a message appears:

```
user@host> configure
```

```
Entering configuration mode
```

```
The configuration has been changed but not committed
```

- If, while in configuration mode, you try to make a change while the configuration is locked by another user, a message indicates that the configuration database is locked, who the user is, and what portion of the configuration the user is viewing or editing:

```
user@host# set system host-name ipswitch
```

```
error: configuration database locked by:
```

```
  user2 terminal d0 (pid 1828) on since 19:47:58 EDT, idle 00:02:11
```

```
    exclusive [edit protocols]
```

- If you enter configuration mode with the **configure exclusive** command, you lock the candidate configuration for as long as you remain in configuration mode, allowing you to make changes without interference from other users. If another user is also in configuration mode and has the configuration locked, a message indicates who the user is and what portion of the configuration the user is viewing or editing:

```
user@host> configure exclusive
```

```
Entering configuration mode
Users currently editing the configuration:
  root terminal p3 (pid 1088) on since 2000-10-30 19:47:58 EDT, idle
00:00:44
  exclusive [edit interfaces so-3/0/0 unit 0 family inet]
```

Exit Configuration Mode

To exit configuration mode, use the **exit configuration-mode** configuration mode command from any level or use the **exit** command from the top level. If you try to exit from configuration mode using the **exit** command and the configuration contains changes that have not been committed, you see a message and prompt:

```
[edit]
user@host# exit

The configuration has been changed but not committed
Exit with uncommitted changes? [yes,no] (yes) <Enter>
Exiting configuration mode
user@host>
```

To exit with uncommitted changes without having to respond to a prompt, use the **exit configuration-mode** command.

Move among Levels of the Hierarchy

The CLI commands in Table 64 help you navigate the levels of the configuration statement hierarchy.

Table 64: CLI Configuration Mode Navigation Commands

Command	Description
edit	To move down through an existing configuration command hierarchy, or to create a hierarchy and move down to that level, use the edit configuration mode command, specifying the hierarchy level at which you want to be.
exit	To move up the hierarchy, use the exit configuration mode command. This command is, in effect, the opposite of the edit command.
up	To move up the hierarchy one level at a time, use the up configuration mode command.
top	To move directly to the top level, use the top configuration mode command.

Display the Current Configuration

You can display the following information about the current configuration:

- Display the Configuration at the Current Hierarchy Level on page 335
- Display the Last Committed Current Configuration on page 335
- Display Users Currently Editing the Configuration on page 335

Display the Configuration at the Current Hierarchy Level

To display the configuration at the current hierarchy level or at the specified level, use the **show** configuration mode command.

```
user@host> show <statement-path>
```

The configuration statements appear in a fixed order. The CLI indents each level in the hierarchy to indicate each statement's relative position in the hierarchy and generally sets off each level with braces, using an open brace at the beginning of each hierarchy level and a closing brace at the end. If the statement at a hierarchy level is empty, the braces are not printed. Each leaf statement ends with a semicolon. If the hierarchy does not extend as far as a leaf statement, the last statement in the hierarchy ends with a semicolon. Interfaces appear alphabetically by type, and then in numerical order by slot number, PIC number, and port number.

Display the Last Committed Current Configuration

You also can use the CLI operational mode **show configuration** command to display the last committed current configuration, which is the configuration currently running on the router:

```
user@host> show configuration
```

Display Users Currently Editing the Configuration

To display the users currently editing the configuration, use the **status configuration mode** command:

```
user@host# status
```

```
Current configuration users:
  user terminal p0 (pid 518) on since 2002-03-12 18:24:27 PST
    [edit protocols]
```

The system displays who is editing the configuration (**user**), how the user is logged in (**terminal p0**), the date and time the user logged in (**2002-03-12 18:24:27 PST**), and what level of the hierarchy the user is editing (**[edit protocols]**).

Modify the Configuration

To configure the router or to modify an existing router configuration, you add statements to the configuration. For each statement hierarchy, you create the hierarchy starting with a statement at the top level and continuing with statements that move progressively lower in the hierarchy.

To modify the hierarchy, you use two configuration mode commands:

- **set**—Creates a statement hierarchy and sets identifier values. After you issue a **set** command, you remain at the same level in the hierarchy. The **set** command has the following syntax:

```
set <statement-path> statement <identifier>
```

statement-path is the hierarchy to the configuration statement and the statement itself. If you have already moved to the statement's hierarchy level, you omit this. *statement* is the configuration statement itself. *identifier* is a string that identifies an instance of a statement.

- **edit**—Moves to a particular hierarchy level. If that hierarchy level does not exist, the **edit** command creates it and then moves to it. The **edit** command has the following syntax:

```
edit <statement-path> statement <identifier>
```

Remove a Statement

To delete a statement or identifier, use the **delete** configuration mode command. Deleting a statement or an identifier effectively “unconfigures” the functionality associated with that statement or identifier, returning that functionality to its default condition. When you delete a statement, the statement and all its subordinate statements and identifiers are removed from the configuration.

```
delete <statement-path> <identifier>
```

To delete the entire hierarchy starting at the current hierarchy level, do not specify a statement or an identifier in the **delete** command:

```
[edit]
user@host# delete
```

```
Delete everything under this level? [yes, no] (no) ?
```

```
Possible completions:
```

```
no          Don't delete everything under this level
```

```
yes         Delete everything under this level
```

```
Delete everything under this level? [yes, no] (no)
```


Run Operational Mode CLI Commands from Configuration Mode

To display the output of an operational mode `show` or other command while configuring the software, you can execute a single operational mode command by issuing the `run` configuration mode command and specifying the operational mode command:

```
[edit]
user@host# run operational-mode-command
```

Display Configuration Mode Command History

To display a list of the recent commands you issued while in configuration mode, use the `run show cli history` command. By default, this command displays the last 100 commands issued in the CLI.

```
user@host# run show cli history

12:40:08 -- show
12:40:17 -- edit protocols
12:40:27 -- set isis
12:40:29 -- edit isis
12:40:40 -- run show cli history
```

Commit a Configuration

To commit a configuration, you can do the following:

- Save Configuration Changes and Activate the Configuration on page 337
- Save Configuration Changes, Activate the Configuration, and Exit Configuration Mode on page 338

Save Configuration Changes and Activate the Configuration

To save software configuration changes to the configuration database and activate the configuration on the router, use the `commit` configuration mode command:

```
user@host# commit

commit complete
```

The configuration is checked for syntax errors. If the syntax is correct, the configuration is activated and becomes the current, operational router configuration. If the configuration contains syntax errors, a message indicates the location of the error and the configuration is not activated. You must correct the error before recommitting the configuration.

Save Configuration Changes, Activate the Configuration, and Exit Configuration Mode

To save software configuration changes, activate the configuration on the router, and exit configuration mode, use the **commit and-quit** configuration mode command. This command succeeds only if the configuration contains no errors.

```
[edit]
user@host# commit and-quit

commit complete
exiting configuration mode
user@host>
```

Save a Configuration to a File

To save the configuration to a text (ASCII) file so that you can edit it with a text editor of your choice, use the **save** configuration mode command. By default, the configuration is saved to that file in your home directory, which is on the flash disk.

```
[edit]
user@host# save filename
```

Return to a Previously Committed Configuration

To return to a previously committed configuration, you can do the following:

- Return to the Most Recent Committed Configuration on page 338
- Activate the Configuration You Loaded on page 338
- Return to a Configuration Prior to the Most Recently Committed One on page 339
- Display Previous Configurations on page 339

Return to the Most Recent Committed Configuration

To return to the most recently committed configuration and load it into configuration mode without activating it, use the **rollback** configuration mode command:

```
[edit]
user@host# rollback

load complete
```

Activate the Configuration You Loaded

To activate the configuration that you loaded, use the **commit** command:

```
[edit]
user@host# rollback
load complete
[edit]
user@host# commit
```

Return to a Configuration Prior to the Most Recently Committed One

To return to a configuration prior to the most recently committed one, include the number in the **rollback** command. *number* can be a number in the range 0 through 9. The most recently saved configuration is number 0 (which is the default configuration to which the system returns), and the oldest saved configuration is number 9.

```
[edit]
user@host# rollback number
load complete
```

Display Previous Configurations

To display previous configurations, including the rollback number, date, time, the name of the user who committed changes, and the method of commit, use the **rollback ?** command.

```
[edit]
user@host# rollback ?

Possible completions:
<[Enter]> Execute this command
<number> Numeric argument
0 2001-02-27 12:52:10 PST by abc via cli
1 2001-02-26 14:47:42 PST by cde via cli
2 2001-02-14 21:55:45 PST by fgh via cli
3 2001-02-10 16:11:30 PST by hij via cli
4 2001-02-10 16:02:35 PST by klm via cli
| Pipe through a command
[edit]
```

Get Help about Statements

In configuration mode, you can use the **help** command to display help based on a text string contained in a statement name. This command displays help for statements at the current hierarchy level and below.

```
user@host# help string
```

You can also display help based on a text string contained in a statement name using the **help topic** and **help reference** commands. The **help topic** command displays usage guidelines for the statement, whereas the **help reference** command displays summary information about the statement.

```
user@host# help topic string
user@host# help reference string
```

If you do not type an option for a statement that requires one, a message indicates the type of information expected. In this example, you need to type an area number to complete the command:

```
[edit]
user@host# set protocols ospf area<Enter>

syntax error, expecting <identifier>.
```

In this example, you need to type a value for the hello interval to complete the command:

```
[edit]
user@host# set protocols ospf area 45 interface so-0/0/0
             hello-interval<Enter>
```

syntax error, expecting <data>

If you have omitted a required statement at a particular hierarchy level, when you attempt to move from that hierarchy level or when you issue the **show** command in configuration mode, a message indicates which statement is missing. For example:

```
[edit protocols pim interface so-0/0/0]
user@host# top
Warning: missing mandatory statement: 'mode'
[edit]
user@host# show
protocols {
  pim {
    interface so-0/0/0 {
      priority 4;
      version 2;
      # Warning: missing mandatory statement(s): 'mode'
    }
  }
}
```

Part 7

Index

- Index on page 343

Index

Symbols

+, statement value indicator	329
>, container statement indicator	329
?, help command	
usage guidelines	323

A

ABR router	
description	130
show configuration command	135
show ospf interface command	135
access account	64
activate command, usage guidelines	20, 327
active	
BGP protocol session state	269
configuration, logging	21, 80
file system	63
adding, JUNOS software packages	69
address	
command	84
statement	84
addresses	
configuring router	84
default router	84
hostname	84
IP	84
machine name	84
adjacencies	
IS-IS, verifying	120
adjacency	
BGP, IS-IS, and OSPF information, logging	62
advertisement packets, OSPF protocol	297
alarmd, Routing Engine daemon	306
alert, severity level	265
all, tracing flag	276
altconfig, file system	63, 71, 82, 87
altroot, file system	63, 71, 82, 87
annotate command, usage guidelines	20, 327
any, system logging facility	264
apply-groups statement	236
apsd, Routing Engine daemon	306
architecture	
Packet Forwarding Engine, figure	4
Routing Engine, figure	4
archive files	268

ASBR router	
description	131
show configuration command	132
show ospf interface command	132
aspath, BGP protocol tracing flag	282
authorization, system logging facility	264
autonomous system	
boundary router (ASBR)	131
OSPF protocol	130

B

backbone, OSPF	130
backups	
copying to router	86
file systems	63, 82
software	71, 87
BGP protocol	
checklist for verifying	157
detail statement	284
displaying messages	270
edit protocol bgp traceoptions command	283
edit protocol command	284
establishment issues	284
flag statement	284
logging information	62, 81
network	
configuration topology, figure	158
topology, figure	165, 171
open statement	284
options	281
protocol statement	284
run show log command	285
send statement	283
session	
problems	268, 284
states, table	268
set flag command	283
show bgp summary command	62, 166
show configuration command	
border router	162
internal router	159
show log command	270
show route advertising-protocol bgp command	169
show route command	
EBGP over IBGP	175
IGP cost	176

local preference	173	CLI	
MED	174	cheat sheet, overview	17
show route forwarding-table command	177	configuration mode	
show route receive-protocol bgp command	170	+ , statement value indicator	329
state transitions, logging	268	> , container statement indicator	329
traceoptions statement	281, 283, 284	changes, uncommitted, exiting with	334
tracing		characters requiring quotation marks	329
configuring	281	command history, displaying list of	337
flags, table	282	commands, table	20, 327
update statement	283	configuration <i>See</i> configuration, router	
binary command	305, 309, 317	description	20, 327
boot floppy	83	entering	333
C		example configuration	332
change directory command	309, 317	exiting	334
change-log		help, displaying	339
facility, configuring	253	hierarchy tree, description	331
statement	253	navigation commands, table	334
system logging facility	264	operational mode commands, running within	337
chassis		statement path, example	332
environment information, logging	78	top-level commands, table	329
hardware		top-level statements, table	330
show chassis hardware command	228	configuration mode commands, table	20
version, logging	60, 77	description	321
information, checklist for displaying	227	operational mode	326
router		command completion	325
overview	9	command history, displaying	326
routers per rack	9	commands, table	18, 322
chassisd		description	18, 322
log file		entering	323
monitor in real time	98	help	323
multiple items, searching for	98	top-level commands, table	324
specific information, searching for	98	using	323
Routing Engine daemon	306	command output, configuration details	54
checklist for		command-line interface <i>See</i> CLI	
BGP protocol, verifying	157	commands	
chassis information, displaying	227	configuration mode CLI, table	20, 327
crash data	299	operational mode CLI, table	18, 322
error conditions, tracking	273	commands for	
files and directories, displaying	231	BGP protocol, verifying	157
IS-IS protocol, verifying	111	chassis information, displaying	227
MIBs	205	crash data, collecting	299
normal operations, tracking	261	error conditions, tracking	273
OSPF protocol, verifying	129	files and directories, displaying	231
physical interfaces	101	IS-IS protocol, verifying	111
ping and traceroute commands	199	MIBs, using	205
problems on your network	29	normal operations, tracking	261
reinstalling software	73	OSPF protocol, verifying	129
router configuration	49	physical interfaces, verifying	101
router hardware components, verifying	91	ping and traceroute commands	199
Routing Engine, CPU memory	179	problems on your network	29
starting and stopping JUNOS software	37	reinstalling software	73
time on a router	239	router configuration, checking	49
traffic and packets	189	router hardware components, verifying	91
upgrading	57	Routing Engine CPU memory, checking	179
user accounts and permissions	247	time on a router, checking	239
version information, displaying	45	traffic and packets, checking	189
clear command, usage guidelines	18, 322	upgrading	57
clear nvram command	314	user accounts and permissions	247
		version information, displaying	45
		commit command, usage guidelines	20, 327, 337

- components
 - alarms, displaying 97
 - CLI commands for monitoring, common 14
 - craft interface, checking 93
 - detailed operational status, CLI commands 96
 - environmental status, CLI commands 95
 - error messages
 - in chassisd log file 98
 - in messages log file 97
 - LED status, checking 94
 - problems
 - solving 99
 - verifying 98
 - router, returning 100
- compress
 - daemon core files 308
 - utility 304, 316
- config, file system 63, 71, 82, 87
- configuration details, displaying 54
- configuration mode commands, table 20
- configuration mode, CLI
 - + , statement value indicator 329
 - > , statement container indicator 329
- command history, displaying 337
- commands
 - activate 20, 327
 - annotate 20, 327
 - commit 20, 327
 - copy 20, 327
 - deactivate 20, 327
 - delete 20, 327
 - edit 20, 327
 - exit 20, 327
 - help 20, 327
 - insert 20, 327
 - load 20, 327
 - paste 21, 328
 - quit 21, 328
 - rollback 21, 328
 - run 21, 328
 - save 21, 328
 - set 21, 328
 - show 21, 328
 - status 21, 328
 - table 20, 327
 - top 21, 328
 - top-level, table 329
 - up 21, 328
 - update 21, 328
- configuration hierarchy, description 331
- configuration *See* configuration, router
- description 20, 327
- displaying current configuration 50
- entering 333
- example configuration 332
- help about statements, getting 339
- hierarchy tree, description 331
- identifier, description 329
- messages
 - locked database 333
 - uncommitted changes 333
 - user editing locked configuration 334
- navigation commands, table 334
- operational mode commands, running within 337
- statement
 - characters requiring quotation marks 329
 - container 331
 - deleting 336
 - description 329
 - leaf 331
- statement hierarchy, figure 331
- statement path, example 332
- statements, top-level
 - accounting-options 330
 - chassis 330
 - class-of-service 330
 - firewall 330
 - forwarding-options 330
 - groups 330
 - interfaces 330
 - policy-options 330
 - protocols 330
 - routing-instances 330
 - routing-options 330
 - snmp 330
 - system 330
 - table 330
- uncommitted changes, exiting with 334
- configuration, router
 - activating 338
 - active, logging 21, 80
 - at a specific level, displaying 335
 - at current hierarchy level, displaying 335
 - backup, copying 86
 - changing while configuration is locked 333
 - committed, most recent, returning to and loading
 - without activating 338
 - currently running on the router, displaying 335
 - edit command, using 336
 - entire hierarchy, deleting from current hierarchy level ..
 - 336
 - example 332
 - file, saving 338
 - format 335
 - group 235
 - hostname 235
 - last current committed, displaying 335
 - management interface 235
 - modifying 336
 - previous, displaying 339
 - prior to most recently committed, returning to 339
 - saving and activating 337
 - saving to a text file 338
 - saving, activating, and exiting 338
 - set command, using 336
 - statement, deleting 336
 - syntax checking 337
 - tracking changes 253

uncommitted changes, exiting with	334	database-description, OSPF protocol tracing flag	294
users currently editing the configuration, displaying.....	335	date, checking.....	302
configurations		dcd, Routing Engine daemon.....	306
configuration details, displaying	54	deactivate command	
current configuration, displaying.....	50	usage guidelines	20, 327
configure command		debug, severity level	265
backup configurations, copying.....	86	delete command	
names and addresses	84	usage guidelines	20, 327
usage guidelines	18, 322	delete routing-options static route command	33
conflict-log, system logging facility	264	detail statement	
connect, BGP protocol session state.....	269	BGP protocol	281, 284
console, logging to	267	IS-IS protocol	286, 291
conventions, documentation	xv	OSPF protocol	293, 297
cooling system, overview.....	13	detailed operational status commands, table	96
copy command		directories, checklist for displaying	231
usage guidelines	20, 327	disk space, displaying	63, 82
copying, JUNOS software packages.....	69	display detail command	
core files		usage guidelines	54
checking.....	302	documentation conventions	xv
compressing.....	304, 316	domain name, configuring.....	84
listing	303, 315	downgrade software, from 5.0 to 4.x	68
cost, IS-IS route selection	121	download JUNOS software	64–66
CPU utilization			
checking with snmpwalk command	219	E	
per process, snmpwalk command.....	220	edit command	
craft interface, table	93	usage guidelines	20, 327, 336
crash data		edit groups command.....	234
checklist for collecting.....	299	edit protocol bgp command.....	268, 284
opening a case with JTAC.....	305, 309, 317	edit protocol bgp traceoptions command	281, 283
overview	301	edit protocol traceoptions command	278
sending to JTAC.....	303, 307, 315	edit protocols isis traceoptions command	286, 289, 291
crash stack traceback.....	310, 311, 313	edit protocols ospf traceoptions command	293, 297
critical, severity level	265	edit routing-options traceoptions command	275
cron, system logging facility	264	edit snmp command	209
csn, IS-IS protocol tracing flag.....	288	edit system command	256
current configuration, displaying	50	edit system syslog command.....	250, 253, 265, 266, 267
current daemon core file	308	edit system syslog statement	263
customer support		emergency, severity level	265
contacting	xix	environment, logging information	78
D		environmental status commands, table	95
daemon		error	
core files		conditions, checklist for tracking	273
checking	306	OSPF protocol tracing flag	294
compressing	308	severity level	265
current	308	established, BGP protocol session state.....	270
previous	308	establishment issues	
run show log command.....	276	BGP protocol	281, 284
set file size command.....	275	OSPF protocol	293
system logging facility	264	Ethernet interface, configuring.....	84
tracing		events	
flags, table.....	276, 306	BGP protocol state transition	268
set flag command	275	OSPF protocol tracing flag	295
damping, BGP protocol tracing flag.....	282	exit command	
data flow		from configuration mode	85
M-series routers.....	6	usage guidelines	20, 327
Packet Forwarding Engine.....	5	exit configuration-mode command	
T-series platforms.....	7	usage guidelines	334
database link-state, examining.....	124		

F

facility, configuring 263, 265, 266, 267
 fans, showing environmental information 78
 FEB 14, 310
 file command, usage guidelines 18, 322
 file copy command 69, 86, 232
 file copy ftp command 232
 file delete command 238
 file list command 237
 daemon core files 306
 Packet Forwarding Engine core files, checking 315
 Routing Engine kernel 302
 file rename command 238
 file system
 /altconfig 63, 71, 82, 87
 /altroot 63, 71, 82, 87
 /config 63
 backing up 63, 71, 82
 disk space 63
 hard drive, backup file system 63
 root, backing up 63
 filenames, listing 326
 files
 checklist for displaying 231
 log, configuring 263
 firewalls
 filter, show firewall filter command 195
 show firewall log command 194
 system logging facility 264
 flag statements
 BGP protocol 283, 284
 IS-IS protocol 286, 289, 291
 OSPF protocol 293, 297
 routing options 275
 specific protocol 278
 flash drive, internal 82
 Flexible PIC Concentrator *See* FPC
 flooding, OSPF protocol tracing flag 295
 Forwarding Engine Board *See* FEB
 forwarding tables
 figure 5
 IS-IS route, verifying 123
 Packet Forwarding Engine 4
 FPC
 definition 310
 M-series and T-series platforms 10
 framing errors 107
 free disk space, displaying 63
 ftp command 305, 309, 318

G

general, tracing flag 276
 GFPC 310
 Gibson Flexible PIC Concentrator *See* GFPC
 global tracing flags 279
 groups statement 234
 groups, configuring 234
 gzip command 304, 308, 316

H

halting router software 38
 hard drives, backup file system 63
 hardware
 components
 overview 9
 router monitoring, table 14
 logging router chassis version 60, 77
 hello
 IS-IS protocol tracing flag 288
 OSPF protocol tracing flag 295
 statement
 IS-IS protocol 286, 289
 OSPF protocol 293
 help
 command, usage guidelines 18, 20, 322, 327
 reference command, usage guidelines 339
 topic command, usage guidelines 339
 hierarchy level 53
 host, configuring 265
 host-name command 235
 hostname, logging 59, 76, 304, 309

I

icons defined, notice xv
 idle, BGP protocol state 269
 insert command
 usage guidelines 20, 327
 interactive-commands, system logging facility 264
 interface address
 ping command 202
 traceroute command 202
 interfaces
 checking for problems 102
 detailed information, displaying 103
 monitor interface command 107
 monitor interface traffic command 190, 191
 monitor traffic interface command 193
 output control keys, table 108
 router, logging 61, 80
 show interfaces terse command 102
 summary information, displaying 102
 system logging 109
 internal flash drive 63
 IS-IS protocol 116
 adjacencies
 status 119
 verifying 120
 checklist for verifying 111
 configuration
 Level 1 router 116
 Level 1/Level 2 router 114
 Level 2 router 117
 detail statement 286, 291
 displaying details 286
 edit protocols isis traceoptions command 286, 289, 291
 flag statement 286, 289, 291
 hello statement 289
 isis traceoptions statement 286

logging information	62, 80
lsp statement	291
LSPs	
analyzing in detail	291
header, examining	126
monitor	
start command	279
stop command	280
network topology	
detailed figure	113
figure	119
levels, figure	112
PDUs	126
receive statement	289
route, examining cost	121
run show isis interface command	114
run show log command	287, 290
send statement	289
set flag command	286, 289, 291
show isis adjacency brief command	62
show isis adjacency command	120
show isis database command	124
show isis database extensive command	126
show isis route command	121
show route command	121
show route forwarding-table destination command	123
trace messages	286
traceoptions statement	289, 291
tracing	
configuring	289
flags, table	288
isis traceoptions statement	286
J	
jbundle package	68
jinstall package	68
JTAC, contacting	15, 99
JUNOS software	
architecture, figure	301
backing up	71, 87
checklist for	
reinstalling	73
upgrading	57
downgrading	68
downloading	64–66
logging version	59, 76, 304, 316
packages, list of	67
reconfiguring	83
reinstalling	83
router compatibility	3
starting	70
supported interfaces	106
upgrading	
comparing before and after	71
packages	67, 69
K	
keepalive, BGP protocol tracing flag	282
kernel	
Routing Engine	302
system logging facility	264
L	
LED status	94
Level 1, IS-IS adjacencies	120
Level 2, IS-IS adjacencies	120
level, configuring	263, 265, 266, 267
link	130
link-state	
advertisement packets	297
database	
description	130
examining	124
PDUs <i>See</i> IS-IS protocol	
load command	
usage guidelines	20, 327
load merge command	86
load replace command	86
log	
active configuration	50
command	251
log files	
configuring	263
displaying	254
firewall with log action	194
number and size	268
severity levels, table	251
start monitoring	271
stop monitoring	271
viewing	270
logging facilities, JUNOS, table	264
logical interfaces	
logging to	
log file	263
remote host	265
router console	267
user terminal	266
severity levels	265
summary information	80
loopback address	
ping command	201
traceroute command	201
lsa-ack, OSPF protocol tracing flag	296
lsa-request, OSPF protocol tracing flag	296
LSAs	130
lsa-update	
OSPF protocol	
statement	297
tracing flag	296
lsp	
IS-IS protocol	
statement	291
tracing flag	288
LSP packets, analyzing IS-IS	291
lsp-generation, IS-IS protocol tracing flag	288

M

management interface, configuring	234
memory utilization, snmpwalk command	213
messages	
force to all users	255
log file	
description	97
monitoring in real time	97
multiple items, searching for	97
specific information, searching for	97
what to display	97
logging to	
local file	263
remote host	265
router console	267
user terminal	266
severity levels, table	265
mgd, Routing Engine daemon	306
mib2d, Routing Engine daemon	307
MIBs	
checklist for using	205
query, snmp command	210
traceoptions output	211
microkernel	
crash stack trace	314
registration information	314
show nvram command	311, 317
mkdir command	305, 309, 317
model, logging router	60, 77
monitor	
command, usage guidelines	18, 322
interface command output fields, table	192
log file messages	271
start command	
BGP protocol	271
IS-IS protocol	279
stop command	
BGP protocol	271
IS-IS protocol	280
monitor interface command	107
monitor interface traffic command	190, 191
monitor traffic interface command	193
MPLS protocol, logging information	62, 81
M-series routers	
chassis	
overview	9
routers per rack	9
component LED locations	94
cooling system, overview	13
craft interface characteristics	93
data flow	
figure	6
process	6
FPC	10
hardware components	
monitoring	14
overview	9
overview	5
Packet Forwarding Engine	5
PICs supported, table	11

power supplies, overview	12
Routing Engine, overview	12
mtrace command	
usage guidelines	18, 322
Multiprotocol Label Switching <i>See</i> MPLS	

N

name	
configuring domain	84
configuring machine	84
prompt	317
near-real time	279
NET address	121
network	
connectivity, checking	83, 85
ping command	85
problems diagnosing, figure	30
problems, checklist	29
topology with a problem, figure	30
network topology	
BGP protocol, configuration topology, figure	158
BGP protocol, figure	165, 171
IS-IS protocol, detailed figure	113
IS-IS protocol, figure	119
levels IS-IS protocol, figure	112
LSA flooding scopes, figure	144
OSPF protocol multi-area, figure	130
OSPF protocol, figure	141
OSPF protocol, figure with details	131
ping command example, figure	200
traceroute command example, figure	200
nonvolatile random access memory <i>See</i> NVRAM	
normal, tracing flag	276
notice icons defined	xv
notice, severity level	265
NSSA, description	131
ntp	
associations, showing	243
status, showing	244
NVRAM	310
clear nvram command	314
clearing	314
PFE location, table	310
storage area	314

O

OID, specific	
snmpwalk command	207
open statement, BGP protocol	284
open, BGP protocol tracing flag	282
openConfirm, BGP session state	270
openSent, BGP protocol session state	269
operational mode, CLI	
command completion	325, 326
command history, displaying	326
commands	
all available, listing	325
clear	18, 322
configure	18, 322

file.....	18, 322
help.....	18, 322, 323
monitor.....	18, 322
mtrace.....	18, 322
of a particular letter, listing.....	324
ping.....	18, 322
pipe.....	18, 322
quit.....	18, 322
request.....	18, 322
restart.....	18, 322
set.....	18, 322
show.....	19, 322
ssh.....	19, 322
start.....	19, 322
table.....	18, 322
telnet.....	19, 322
test.....	19, 322
top-level, listing.....	324
traceroute.....	19, 322
description.....	18, 322
entering.....	323
exiting.....	334
filenames, listing.....	326
using.....	323
operational status.....	
CLI commands.....	96
commands, table.....	96
OSPF protocol.....	
ABR, description.....	130
AS.....	130
ASBR, description.....	131
autonomous system.....	130
autonomous system boundary router.....	131
backbone.....	130
checklist for verifying.....	129
detail statement.....	293, 297
edit protocols ospf traceoptions command.....	293, 297
flag statement.....	293, 297
hello statement.....	293
link-state advertisement packets.....	297
link-state database.....	130
LSA flooding scopes, figure.....	144
lsa-update statement.....	297
LSPs analyzing.....	297
multi-area network topology, figure.....	130
network topology with details, figure.....	131
network topology, figure.....	141
run show log command.....	294, 297
set flag command.....	293, 297
show configuration command.....	132
show ospf database asbrsummary extensive command.....	154
show ospf database command.....	145, 148
show ospf database extern extensive command.....	155
show ospf database netsummary extensive command.....	153
show ospf database nssa extensive command.....	156
show ospf database router extensive command.....	152
show ospf interface command.....	132, 135, 139
show ospf neighbor brief command.....	62
show ospf neighbor command.....	142
show route command.....	148
show route extensive command.....	151
show route forwarding-table destination command.....	151
single area border router.....	130
spf, tracing flag.....	296
traceoptions statement.....	293, 297
tracing.....	
flags, table.....	294
messages.....	293
P	
Packet Forwarding Engine.....	
architecture, figure.....	4
core files, checking.....	315
data flow.....	5
forwarding tables.....	
figure.....	5
overview.....	4
Layer 2 and 3 packet switching, route lookup and packet forwarding.....	4
microkernel.....	301, 310, 315
M-series routers.....	5
show nvram command.....	311, 317
show pfe statistics command.....	196
show syslog messages command.....	311, 317
show version command.....	316, 317
start shell command.....	311, 314
T-series platforms.....	5
packet-dump, OSPF protocol tracing flag.....	296
packets.....	
and traffic, checklist.....	189
BGP protocol tracing flag.....	283
IS-IS protocol tracing flag.....	288
OSPF protocol tracing flag.....	296
received.....	283, 289
sent.....	283, 289
password.....	
clear text.....	85
e-mail address.....	305
encrypted.....	85
prompt.....	317
root, setting.....	85
ssh public string.....	85
paste command.....	
usage guidelines.....	21, 328
PCMCIA card.....	83
PDU's <i>See</i> IS-IS protocol.....	
permissions, checklist.....	247
Personal Computer Memory Card International Association <i>See</i> PCMCIA.....	
pfe, system logging facility.....	264
Physical Interface Card <i>See</i> PIC.....	
physical interfaces, summary information.....	80
PIC.....	
media types, table.....	10
overview.....	10
supported, table.....	11, 12
PIM protocol, logging information.....	62, 81

- ping command
 - checklist 199
 - interface address 202
 - loopback address 201
 - network
 - connectivity, checking 85
 - problems, identifying 31
 - problems, identifying solutions 34
 - topology for checking reachability, figure 200
 - unsuccessful 203
 - usage guidelines 18, 322
 - pipe command
 - usage guidelines 18, 322
 - policy, tracing flag 277
 - power supplies
 - environmental information 78
 - overview 12
 - previous daemon core file 308
 - processes
 - restarting 40
 - viewing 59
 - prompts
 - ftp 305, 309
 - name 305, 309, 317
 - password 305, 309, 317
 - shell 303, 307, 309, 311, 315, 317
 - vty 314
 - protocol bgp statement 284
 - Protocol Independent Multicast *See* PIM
 - protocols
 - edit protocol traceoptions command 278
 - flag statement 278
 - peer information, logging 62
 - set file size files command 278
 - tracing, configuring 275, 278
 - psn, IS-IS protocol tracing flag 289
 - put command 305, 309, 317
- Q**
- quit command
 - usage guidelines 18, 21, 322, 328
- R**
- rack, maximum number of routers, table 9
 - RADIUS server, checking connectivity 256
 - rebooting router software 40
 - receive statements
 - BGP protocol statement 283
 - IS-IS protocol 289
 - reconfiguring JUNOS software 83
 - registration information 311, 313
 - location 310
 - reinstall JUNOS software
 - checklist 73
 - comparing configurations 87
 - saving log information 75
 - steps 83
 - remote host, logging messages to 265
 - request command
 - usage guidelines 18, 322
 - request message all message command 255
 - request support information
 - commands included
 - show chassis environment 99
 - show chassis firmware 99
 - show chassis hardware 99
 - show configuration 100
 - show interfaces extensive 99
 - show system virtual-memory 100
 - show version 99
 - request support information command 15
 - information to provide JTAC 99
 - usage guidelines 99
 - request system halt command 38
 - request system logout command 252
 - request system reboot command 39, 70
 - request system snapshot command 63, 71, 82, 87
 - request system software add command 69
 - Resource Reservation Protocol *See* RSVP
 - restart command
 - usage guidelines 18, 322
 - restart routing
 - command options, table 42
 - restart routing command 41
 - restarting software processes 40
 - rollback command
 - usage guidelines 21, 328
 - root
 - file system, backing up 63, 71, 82
 - password 85, 303, 307, 311, 314, 315
 - route
 - IS-IS
 - forwarding table, verifying 123
 - show isis route command 121
 - show route command 121
 - show route detail command 121
 - tracing flag 277
 - router software
 - halting 38
 - rebooting 40
 - routers
 - ABR router, configuring 135
 - active configuration, logging 21, 80
 - adjacency IS-IS, verifying 120
 - architecture
 - figure 4
 - overview 4
 - ASBR router, configuring 132
 - border router BGP, configuring 162
 - chassis
 - hardware version, logging 60, 77
 - overview 9
 - routers per rack, table 9
 - check network connectivity 85
 - components, checking 240

configuration	
BGP protocol	158
IS-IS protocol	116
OSPF protocol	130
configuring name and address	84
console, logging to	267
copying backup configuration	86
environment, logging	78
hardware monitoring, table	14
interfaces, logging	61, 80
internal router BGP, configuring	159
Juniper Networks, overview	3
JUNOS software, compatibility	3
Level 1 IS-IS, configuring	116
Level 1/Level 2 IS-IS, configuring	114
Level 2 IS-IS, configuring	117
maximum per rack, table	9
model, logging	60, 77, 304, 309
M-series, overview	5
Packet Forwarding Engine	4
PICs supported, table	11, 12
platforms, overview	3
Routing Engine	4
stub router, configuring	139
T-series platforms, overview	5
routing	
protocol daemon tracing	
configuring	275
flags, table	276
protocol process, IS-IS protocol	123
protocols trace options, table	279
table updates, figure	5
tables, Routing Engine	4
Routing Engine	
architecture, figure	4
core files	302
CPU memory, checklist for verifying	179
daemons	301, 306
table	306
environment information	78
kernel	301
crash, overview	302
file list command	302
Layer 3 routing	4
monitor traffic interface command	193
overview	12
routing tables	
figure	5
overview	4
show version command	304, 308
start shell command	303, 307
routing-options	
flag statement	275
statement	275
rpdp, Routing Engine daemon	306
RSVP protocol, logging information	62, 81
run command	
usage guidelines	21, 328, 337
run show isis interface command	114, 116, 117
run show log command	251, 254
BGP protocol	282, 284, 285
daemon	276
IS-IS protocol	287, 290, 291
OSPF protocol	278, 294, 297
S	
sampled, Routing Engine daemon	306
save command	
usage guidelines	21, 58, 75, 328, 338
SCB	14, 310
send statement	
BGP protocol	283
IS-IS protocol	289
server RADIUS, checking connectivity	256
service contract, software	64
session states, BGP protocol	268
set apply-groups command	236
set archive files command	268
set change-log command	253
set cli terminal command	107
set command	
usage guidelines	18, 21, 322, 328, 336
set console command	267
set file command	263
set file size command	275
set file size files command	278
set flag command	
BGP protocol	283
daemon tracing	275
IS-IS protocol	286, 289, 291
OSPF protocol	293, 297
specific protocol	278
set flag hello detail command	286, 293
set flag hello send command	289
set flag lsa-update detail command	297
set flag lsp detail command	291
set flag update detail command	281
set flag update send command	283
set host command	265
set host statement	265
set host-name command	235
set interfaces address command	84
set interfaces command	235
set log-updown command	268
set re0 command	234
set system backup-router command	84
set system domain-name command	84
set system host-name command	84
set system name-server command	84
set system root-authentication command	85
set system root-authentication encrypted-password	
command	85
set system root-authentication plain-text-password	
command	85
set traceoptions flag open detail command	284
set traceoptions flag pdu command	209
set user command	266
setting root password	85

- severity levels, table 251, 265
- SFM, Packet Forwarding Engine component 310
- shell prompts 317
- show bgp summary command 62, 81, 87, 166
- show chassis alarms command 14, 97
- show chassis command 14, 96, 240
 - table 96
- show chassis craft-interface command 14, 93, 94
- show chassis environment command 14, 78, 87, 95
- show chassis firmware command 14
- show chassis hardware command 14, 60, 77, 87, 100, 228
- show chassis hardware command output fields, table 230
- show chassis routing-engine command 213
- show cli history command
 - usage guidelines 337
- show command
 - usage guidelines 19, 21, 50, 322, 328, 335
- show configuration command 32, 50, 54
 - ABR 135
 - ASBR 132
 - BGP protocol
 - border router 162
 - internal router 159
 - log active 61, 80, 87
 - OSPF protocol 132, 135, 139
 - stub router 139
 - usage guidelines 50, 335
- show configuration snmp command 209
- show firewall filter command 195
- show firewall log command 194
- show interface terse command 61, 80, 87
- show interfaces extensive command 103
- show interfaces terse command 102
- show isis adjacency brief command 62, 81, 87
- show isis adjacency command 120
- show isis database command 124
- show isis database extensive command 126
- show isis route command 121
- show log chassisd command 15, 98
- show log command 254, 270
- show log messages command 14, 97, 109
- show log snmpd command
 - MIB, querying 211
- show ntp associations command 243
- show ntp status command 244
 - output fields, table 244
- show nvram command 311, 317
- show ospf database asbrsummary extensive command 154
- show ospf database command 145, 148
- show ospf database extern extensive command 155
- show ospf database netsummary extensive command 153
- show ospf database nssa extensive command 156
- show ospf database router extensive command 152
- show ospf interface command 132, 135, 139
- show ospf neighbor brief command 62, 81, 87
- show ospf neighbor command 142
 - output fields, table 143
- show pfe statistics traffic command 196
- show route advertising-protocol bgp command 169
- show route command 31, 33, 34, 85, 121
 - BGP protocol
 - EBGP over IBGP 175
 - IGP cost 176
 - local preference 173
 - MED 174
 - OSPF protocol 148
- show route extensive command 151
- show route forwarding-table command 177
- show route forwarding-table destination command 123, 151
- show route receive-protocol bgp command 170
- show route summary command 183
- show syslog messages command 311, 317
- show system boot-messages command 78, 87
- show system processes extensive command 40, 42, 180
 - output, table 41
- show system storage command 63, 82, 87
- show system uptime command 240
- show system users command 248
 - output fields, table 248
- show task command 185
- show task memory command 185
- show task memory detail command 183
- show version brief command, JUNOS software packages 47
- show version command 76, 305
 - compare information 87
 - core files, Packet Forwarding Engine 316, 317
 - daemon core files, Routing Engine 308
 - definition 14
 - JUNOS software 46
 - reinstalling software 76
 - Routing Engine 304
 - upgrading software 59
- single area border router, OSPF 130
- SNMP, show chassis routing-engine command 213
- snmpd, Routing Engine daemon 306
- snmpget command
 - MIB, querying 210
- snmpwalk command
 - CPU utilization 219
 - CPU utilization per process, checking 220
 - memory utilization
 - checking 213
 - checking per process 215
 - specific OID 207
 - version information 223
- software processes, restarting 40
- software, JUNOS 69
 - architecture, figure 301
 - backing up 71, 87
 - checklist for reinstalling 73
 - comparing before and after upgrade 71
 - downloading 64–66
 - logging hardware version 60, 77
 - logging software version 59, 76, 304, 316
 - packages
 - adding new 67, 69
 - copying 69
 - list of 67
 - logging 59, 76

reconfiguring.....	83
reinstalling.....	83
starting.....	70
upgrading.....	67, 69
software, router	
halting.....	38
rebooting.....	40
spf	
IS-IS protocol tracing flag.....	289
OSPF protocol tracing flag.....	296
SPMB.....	310
SSB.....	14, 310
ssh command	
usage guidelines.....	19, 322
start command	
usage guidelines.....	19, 322
start shell command	
Packet Forwarding Engine	
clear NVRAM.....	314
core files.....	315
traceback.....	311
Routing Engine	
core files.....	303
daemon core files.....	307
starting and stopping JUNOS software.....	37
starting, JUNOS Software packages.....	69
state transition events.....	268
state, tracing flag.....	277
statements	
configuration mode CLI, top-level.....	330
states, BGP protocol.....	268
statistics	
real-time display.....	107
set cli terminal command.....	107
status command	
usage guidelines.....	21, 328
storage area, NVRAM.....	314
stub area.....	131
stub router, configuration.....	139
su command.....	303, 307, 311, 314, 315
support, technical	
customer support, contacting.....	xix
Switched Printed Mezzanine Board <i>See</i> SPMB	
Switching and Forwarding Module <i>See</i> SFM	
syslogd, Routing Engine daemon.....	307
system	
architecture, figure.....	5
backup-router statement.....	84
boot-message, logging.....	78
information, obtaining.....	71, 87
kernel messages, displaying.....	79
logging	
configuring.....	263
daemon, facility.....	264
facilities, table.....	264
interfaces.....	109
message severity levels, table.....	265
storage, logging information.....	63, 82
System Control Board <i>See</i> SCB	
System Switching Board <i>See</i> SSB	
system syslog command.....	250, 253
system users command.....	248
T	
task, tracing flag.....	277
technical support	
customer support, contacting.....	xix
telnet command	
usage guidelines.....	19, 322
temperature, environmental information.....	78
test command	
usage guidelines.....	19, 322
time on a router, checklist for.....	239
time, checking system uptime.....	240
timer, tracing flag.....	277
timestamp, kernel core files.....	302
top command	
usage guidelines.....	21, 328
trace files	
start monitoring.....	279
stop monitoring.....	280
viewing.....	270
traceoptions statement	
BGP protocol.....	283, 284
daemon tracing.....	275
IS-IS protocol.....	286, 289, 291
OSPF protocol.....	293, 297
routing protocols, table.....	279
specific protocol.....	278
traceroute command	
checking.....	199
identifying solutions to network problems.....	34
interface address.....	202
loopback address.....	201
network problems, identifying.....	31
network topology for checking packets.....	200
unsuccessful.....	203
usage guidelines.....	19, 322
tracing flags	
BGP protocol, table.....	282
IS-IS protocol, table.....	288
OSPF protocol, table.....	294
routing protocol daemon, table.....	276
standard for routing protocols, table.....	279
traffic and packets, checklist.....	189
T-series platforms	
chassis	
overview.....	9
routers per rack.....	9
component LED locations.....	94
cooling system, overview.....	13
data flow	
figure.....	7
process.....	7
FPC.....	10
hardware components	
monitoring.....	14
overview.....	9
overview.....	5

Packet Forwarding Engine	5
PICs supported, table	12
power supplies, overview	12
Routing Engine, overview	12
typefaces, documentation conventions	xv

U

up command	
usage guidelines	21, 328
update command	
usage guidelines	21, 328
update statement, BGP protocol	281, 283
update, BGP protocol tracing flag	283
upgrade JUNOS software	
backing up	71
checklist for upgrading	57
comparing software	71, 87
copying, adding and starting	69
from 4x to 5.0	68
reinstalling	73
saving log information	75
user	
accounts, checklist	247
configuring	266
currently editing the configuration, checking	249
forcing messages to all	255
system logging facility	264
terminal, logging messages to	266
utility, gzip	304, 316

V

version information	
JUNOS software	304, 316
snmpwalk command	223
vmcore file, compressing	304
vrrpd, Routing Engine daemon	307
vty command	314
vty session	311, 314

W

warning, severity level	265
-------------------------------	-----

