



JUNOS® Software

Routing Matrix with a TX Matrix Plus Router Feature Guide

Juniper Networks, Inc.

1194 North Mathilda Avenue
Sunnyvale, California 94089
USA

408-745-2000

www.juniper.net

Published: 2009-10-20

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.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

Juniper Networks, the Juniper Networks logo, JUNOS, NetScreen, ScreenOS, and Steel-Belted Radius are registered trademarks of Juniper Networks, Inc. in the United States and other countries. JUNOSe is a trademark 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.

JUNOS® Software Routing Matrix with a TX Matrix Plus Router Feature Guide,
Copyright © 2009, Juniper Networks, Inc.
All rights reserved. Printed in USA.

Writing: Donna Ono
Editing: Sonia Saruba
Illustration: Faith Bradford, Fawn Damitio, Nathaniel Woodward, and Richard Hendricks
Cover Design: Edmonds Design

Revision History
August 2009—

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.

END USER LICENSE AGREEMENT

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

agreements relating to the Software, whether oral or written (including any inconsistent terms contained in a purchase order), except that the terms of a separate written agreement executed by an authorized Juniper representative and Customer shall govern to the extent such terms are inconsistent or conflict with terms contained herein. No modification to this Agreement nor any waiver of any rights hereunder shall be effective unless expressly assented to in writing by the party to be charged. If any portion of this Agreement is held invalid, the Parties agree that such invalidity shall not affect the validity of the remainder of this Agreement. This Agreement and associated documentation has been written in the English language, and the Parties agree that the English version will govern. (For Canada: Les parties aux 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

Part 1	Overview	
Chapter 1	Product Overview	3
Chapter 2	Product Reference Information	11
Part 2	Configuration	
Chapter 3	Product Configuration	15
Chapter 4	Product Configuration Examples	31
Part 3	Administration	
Chapter 5	Upgrading the JUNOS Software	61
Chapter 6	Managing Files and Processes	97
Chapter 7	Managing Hardware Components	103
Part 4	Index	
	Index	119

Table of Contents

Part 1	Overview	
Chapter 1	Product Overview	3
	Overview of a Routing Matrix with a TX Matrix Plus Router	3
	Components of a Routing Matrix with a TX Matrix Plus Router	3
	Architecture of a Routing Matrix with a TX Matrix Plus Router	4
	Managing the Routing Matrix as a Single Router	5
	Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router	5
	Routing Engine Connectivity in a Routing Matrix with a TX Matrix Plus Router	7
	Redundant Host Subsystems in a Routing Matrix with a TX Matrix Plus Router	8
Chapter 2	Product Reference Information	11
	System Requirements for a Routing Matrix with a TX Matrix Plus Router	11
Part 2	Configuration	
Chapter 3	Product Configuration	15
	Roadmap for Configuring a Routing Matrix with a TX Matrix Plus Router	15
	Connecting to a Routing Matrix with a TX Matrix Plus Router	16
	Accessing a Routing Engine by Direct Console Connection to a Serial Port	16
	Accessing a Routing Engine by Telnet Access to a Management Ethernet Port	17

Accessing a Routing Engine by Logging In from the CLI of Another Routing Engine	17
Connecting to a Backup Routing Engine on the TX Matrix Plus Router	18
Using Configuration Groups and Inheritance in a Routing Matrix with a TX Matrix Plus Router	18
Overview of Configuration Groups and Inheritance in a Routing Matrix	18
Creating Configuration Groups in a Routing Matrix	19
Applying a Configuration Group in a Routing Matrix	20
Displaying Inherited Statements	20
Disabling Inheritance of a Configuration Group in the Routing Matrix	21
Using Special Configuration Groups for the Routing Engines in the Routing Matrix	21
Using Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router	23
Configuring Protocols and Other Features on a Routing Matrix with a TX Matrix Plus Router	24
Configuring Chassis-Specific Features on a Routing Matrix with a TX Matrix Plus Router	25
Configuring PIC-Specific Features on a T1600 Router in a Routing Matrix	25
Configuring an Alarm to Trigger if an LCC in the Routing Matrix Does Not Come Online	26
Configuring an LCC in the Routing Matrix to Stay Offline	26
For More Information	26
Committing Configurations on a Routing Matrix with a TX Matrix Plus Router	27
Committing a Configuration to Both Master and Backup Routing Engines in the Routing Matrix	27
Committing a Configuration to the Master Routing Engines (Only) in the Routing Matrix	28
Synchronizing to the Configuration on the Other Routing Engine	28

Chapter 4 Product Configuration Examples 31

Merging Examples	31
Merging a Full Example	31
Merging a Snippet	32
For More Information	32
Example Configuration for a Routing Matrix with a TX Matrix Plus	
Router	33
Example Routing Matrix Topology	33
Example Routing Matrix Configuration	34
Verifying the Example Configuration	39
Verifying the Configuration of a Routing Matrix with a TX Matrix Plus	
Router	40
Displaying JUNOS Software Versions	41
Displaying JUNOS Software Versions for All Routers	41
Displaying JUNOS Software Version for the SFC Only	42
Displaying JUNOS Software Version for a Specific LCC	43
Displaying JUNOS Software Versions for All LCCs	43
Displaying Configured Interfaces	44
Displaying Available Routes	45
Displaying Alarms and System Uptime	46
Displaying Craft Interface Messages	46
Displaying System Uptime	49
Displaying Chassis Hardware and Status	49
Displaying Information About All Hardware Components	50
Displaying Information About SIBs	53
Displaying Information About Routing Engines	54
Displaying Information About FPCs	56
Displaying Information About LCCs	57

Part 3 Administration

Chapter 5 Upgrading the JUNOS Software 61

Upgrading the JUNOS Software on a Routing Matrix with a TX Matrix Plus	
Router	61
Logging In to the Master Routing Engine on the TX Matrix Plus	
Router	63
Disabling Routing Engine Redundancy and GRES on the Routing Matrix	63
Installing JUNOS Software on the Backup Routing Engines	65
Loading Software on the Backup Routing Engines	73
Switching Routing Engine Mastership to the Backup Routing Engines	75
Installing Software on the New Default Master Routing Engines	77
Loading Software on the New Default Master Routing Engines	78

Optional: Switching Routing Engine Mastership Back to the Default Master Routing Engines	80
Restoring the Original Configuration	82
Backing Up the Routing Engines	83
Troubleshooting: Software Upgrade Failure Due to Insufficient Free Disk Space on a Routing Matrix with a TX Matrix Plus Router	85
Troubleshooting: Software Upgrade Failure Due to Insufficient Available DRAM on a Routing Matrix with a TX Matrix Plus Router	91
Troubleshooting: Software Upgrade Failure Due to Mixed Software Versions on a Routing Matrix with a TX Matrix Plus Router	91

Chapter 6 **Managing Files and Processes** **97**

Managing Files on Routing Engines in a Routing Matrix with a TX Matrix Plus Router	97
Displaying a List of Files	97
Displaying the Contents of a File	98
Copying Files	99
Renaming Files	99
Deleting Files	99
Managing System Processes in a Routing Matrix with a TX Matrix Plus Router	100

Chapter 7 **Managing Hardware Components** **103**

Displaying Chassis Physical Locations for a Routing Matrix with a TX Matrix Plus Router	103
Displaying the Status of the Switching Fabric Topology of a Routing Matrix with a TX Matrix Plus Router	104
Displaying the Status of Control Board Ethernet Switch Ports in a Routing Matrix with a TX Matrix Plus Router	107
Ethernet Switches on Control Boards in the Routing Matrix	107
Connectivity Between SFC and LCC Routing Engines	107
Connectivity Between SFC Master and Backup Routing Engines	108
Connectivity Between LCC Master and Backup Routing Engines	108
Status of Ports on Control Board Ethernet Switches	109
Error Statistics for Ports on Control Board Ethernet Switches	109
Traffic Statistics for Ports on Control Board Ethernet Switches	110
Enabling and Disabling Hardware Components of a Routing Matrix with a TX Matrix Plus Router	111
Rebooting and Halting Hardware Components of the Routing Matrix with a TX Matrix Plus Router	114

Part 4 **Index**

Index	119
-------------	-----

List of Figures

Part 1	Overview	
Chapter 1	Product Overview	3
	Figure 1: Architecture of a Routing Matrix with a TX Matrix Plus Router	4
Part 2	Configuration	
Chapter 4	Product Configuration Examples	31
	Figure 2: Example Topology of a Routing Matrix with a TX Matrix Plus Router	33

List of Tables

Part 1	Overview	
Chapter 1	Product Overview	3
	Table 1: FPC Numbering for T1600 Routers in a Routing Matrix — Basic Correspondence	6
	Table 2: FPC Numbering for T1600 Routers in a Routing Matrix — Conversion Chart	6

Part 1

Overview

- Product Overview on page 3
- Product Reference Information on page 11

Chapter 1

Product Overview

- Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
- Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router on page 5
- Routing Engine Connectivity in a Routing Matrix with a TX Matrix Plus Router on page 7
- Redundant Host Subsystems in a Routing Matrix with a TX Matrix Plus Router on page 8

Overview of a Routing Matrix with a TX Matrix Plus Router

A routing matrix based on a Juniper Networks TX Matrix Plus Router is a multichassis architecture composed of one TX Matrix Plus router and from one to four interconnected T1600 routers.

- Components of a Routing Matrix with a TX Matrix Plus Router on page 3
- Architecture of a Routing Matrix with a TX Matrix Plus Router on page 4
- Managing the Routing Matrix as a Single Router on page 5

Components of a Routing Matrix with a TX Matrix Plus Router

A routing matrix contains two types of chassis: a Juniper Networks TX Matrix Plus Router and Juniper Networks T1600 Core Router.

- TX Matrix Plus router—A routing matrix can contain only one Juniper Networks TX Matrix Plus router. A TX Matrix Plus router is often referred to as the *switch-fabric chassis* (SFC). In the JUNOS command-line interface (CLI), `sfc0` is used to refer to the TX Matrix Plus router. For information about installing and connecting to a TX Matrix Plus router, see the *TX Matrix Plus Router Hardware Guide*.
- T1600 routers—A routing matrix can contain from one to four T1600 routers, numbered from 0 through 3. The T1600 router number is set by the hardware. A T1600 router in a routing matrix is often referred to as a *line-card chassis* (LCC). In the JUNOS CLI, `lcc0` through `lcc3` are used to refer to a T1600 router in a routing matrix. For information about connecting to a T1600 router in a routing matrix, see the *TX Matrix Plus Router Hardware Guide*.

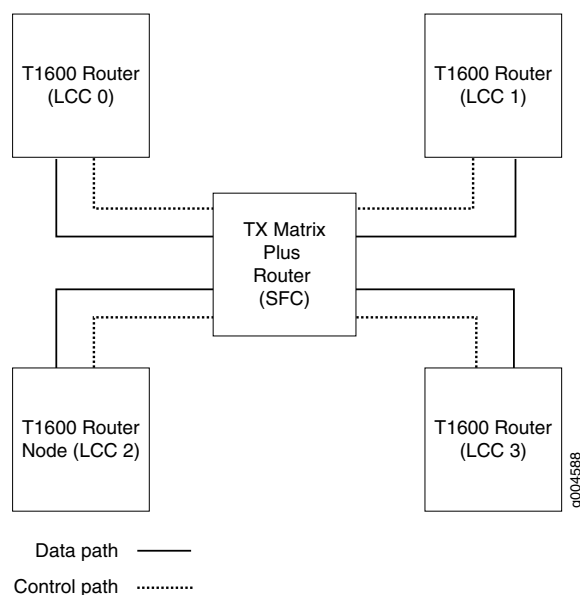


NOTE: Nonstop active routing (NSR) is not supported for T1600 routers in a routing matrix.

Architecture of a Routing Matrix with a TX Matrix Plus Router

In a routing matrix, the TX Matrix Plus router controls all the connected T1600 routers, as shown in Figure 1 on page 4.

Figure 1: Architecture of a Routing Matrix with a TX Matrix Plus Router



A key element of the routing matrix design is the ability to migrate existing standalone T1600 routers and connect them with a TX Matrix Plus router through fiber-optic cables and Switch Interface Boards (SIBs).

The TX Matrix Plus router connection between the T1600 routers uses a scalable, three-stage switch fabric. This system architecture provides terabit bandwidth expansion capacity and eliminates the use of subscriber line cards to connect devices within points of presence (POPs). As a result, the primary application for the routing matrix is to collapse aggregation and core layers in large POPs and central offices.

Similarly, you can limit which portions of the routing matrix are modified during configuration or maintenance procedures (for example, performing software upgrades or halting Routing Engines).

Managing the Routing Matrix as a Single Router

Although a routing matrix can contain up to five separate physical components, from the perspective of the user interface, the routing matrix appears as a single router for most operations and uses the existing JUNOS CLI.

To manage the multichassis system, some enhancements have been made to the CLI to enable you to select the amount of output you want to display when you issue operational commands. For example, you can specify the entire routing matrix, the TX Matrix Plus router, a specific T1600 router and its Flexible PIC Concentrators (FPCs), or a combination thereof.

When you issue configuration and operational commands on a TX Matrix Plus router, your view of the routing matrix shows a single routing device with a single range of FPC numbers (from 0 through 31 rather than from 0 through 7 for each FPC). As a result, you will need to adjust certain configuration statements to accommodate the global numbering of FPCs installed on the routing matrix, as described in “Using Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router” on page 23.

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Global FPC Numbering for Interfaces in the Routing Matrix on page 5
 - Routing Engine Connectivity in the Routing Matrix on page 7
 - Redundant Host Subsystems in the Routing Matrix on page 8
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router

A routing matrix with a TX Matrix Plus router can contain up to four T1600 routers (line-card chassis) that are assigned numbers, from 0 through 3, depending on the hardware setup and connectivity to the TX Matrix Plus router. In the JUNOS CLI, `lcc 0` through `lcc 3` are used to refer to a T1600 router in a routing matrix. Each T1600 router can contain up to eight Flexible PIC Concentrators (FPCs) in FPC hardware slots labeled 0 through 7. Therefore, a routing matrix with a TX Matrix Plus router can contain up to 32 FPCs.

In the JUNOS CLI, an interface name has the following format:

type-fpc/pic/port

When you specify the *fpc* number for a T1600 router in a routing matrix, the JUNOS Software determines which T1600 router contains the specified FPC. Table 1 on page 6 shows the basic correspondence between FPC hardware slot numbers (0 through

7), which are labeled on the T1600 chassis, and routing matrix FPC numbers (0 through 31), which are used in the JUNOS CLI.

Table 1: FPC Numbering for T1600 Routers in a Routing Matrix — Basic Correspondence

T1600 Router in a Routing Matrix	Basic Correspondence of FPC Numbering	
	FPC Hardware Slot Numbers	Global FPC Numbers
LCC 0	0 – 7	0 – 7
LCC 1	0 – 7	8 – 15
LCC 2	0 – 7	16 – 23
LCC 3	0 – 7	24 – 31

To easily convert an FPC hardware slot number in a specific LCC (T1600 router) in a routing matrix to the corresponding global FPC number, use the conversion chart shown in Table 2 on page 6. You can use the converted FPC number to configure the interfaces on the TX Matrix Plus router in your routing matrix.

Table 2: FPC Numbering for T1600 Routers in a Routing Matrix — Conversion Chart

Conversion Chart for FPC Numbering										
LCC 0										
FPC Hardware Slot Numbers	0	1	2	3	4	5	6	7	Example: In the interface name se-1/0/0, the global FPC number 1 refers to FPC hardware slot 1 in the T1600 router lcc0.	
Global FPC Numbers	0	1	2	3	4	5	6	7		
LCC 1										
FPC Hardware Slot Numbers	0	1	2	3	4	5	6	7	Example: In the interface name t1-11/2/0, the global FPC number 11 refers to FPC hardware slot 3 in the T1600 router lcc1.	
Global FPC Numbers	8	9	10	11	12	13	14	15		
LCC 2										
FPC Hardware Slot Numbers	0	1	2	3	4	5	6	7	Example: In the interface name so-20/0/1, the global FPC number 20 refers to FPC hardware slot 4 in the T1600 router lcc2.	
Global FPC Numbers	16	17	18	19	20	21	22	23		
LCC 3										
FPC Hardware Slot Numbers	0	1	2	3	4	5	6	7	Example: In the interface name t3-31/1/0, the global FPC number 31 refers to the FPC in slot 7 in the T1600 router lcc3.	
Global FPC Numbers	24	25	26	27	28	29	30	31		

You can use a CLI operational command to display the numbers of the connected FPCs in a routing matrix. To display the FPC numbers in both "global numbering" mode (0 through 31, as used to specify an interface name in a routing matrix) and associated "local numbering" mode (LCC number and FPC physical slot number), you can use various forms of the **show chassis location** command. For more information, see "Displaying Chassis Physical Locations for a Routing Matrix with a TX Matrix Plus Router" on page 103.

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Routing Engine Connectivity in a Routing Matrix with a TX Matrix Plus Router

The TX Matrix Plus router contains two Connector Interface Panels (CIPs)—one for each Routing Engine—that are referred to as **TXP-CIP-0** and **TXP-CIP-1**. In a routing matrix with a TX Matrix Plus router, all Routing Engines are connected to their respective Control Boards, which in turn are connected to ports on the two CIPs on the TX Matrix Plus router:

- On a TX Matrix Plus router, the Routing Engine (RE-TXP-SFC) and Control Board (TXP-CB) function as a unit, or host subsystem. For each of the two host subsystems in the router, the JUNOS Software automatically creates two internal Ethernet interfaces, **ixgbe0** and **ixgbe1**, for the two 10-Gigabit Ethernet ports on the Routing Engine. The port at **ixgbe0** connects the TX Matrix Plus Routing Engine to the Routing Engines of every T1600 router configured in the routing matrix.
- On a T1600 router configured in a routing matrix, the Routing Engine (RE-TXP-LCC) and Control Board (LCC-CB) function as a unit, or host subsystem. For each of the two host subsystems in the router, the JUNOS Software automatically creates two internal Ethernet interfaces, **bcm0** and **em1**, for the two Gigabit Ethernet ports on the Routing Engine. The port at **bcm0** connects the LCC Routing Engine to the Routing Engines of every other T1600 router configured in the routing matrix.



NOTE: The Routing Engines in the TX Matrix Plus router and in the T1600 routers configured in a routing matrix do not support the management Ethernet interface **fxp0** or the internal Ethernet interfaces **fxp1** or **fxp2**.

Configurations and automated scripts that have been developed for standalone T1600 routers (T1600 routers not configured in a routing matrix) might contain references to the **fxp0**, **fxp1**, or **fxp2** interfaces.

Before reusing the configurations or scripts on T1600 routers in a routing matrix, update the files as appropriate:

- Edit any command lines that reference the T1600 router management Ethernet interface `fxp0` by replacing “`fxp0`” with “`em0`.”
- Ensure that any `show interfaces` commands that are intended to list the T1600 router internal Ethernet interfaces refer to the `bcm0` or `em1` interfaces.

For more detailed information, see the *JUNOS Network Interfaces Configuration Guide* and the *JUNOS Interfaces Command Reference*.

Related Topics

- Routing Matrix with a TX Matrix Plus Router Solutions Page
- Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
- Roadmap for Configuring the Routing Matrix on page 15
- Example Configuration for the Routing Matrix on page 33
- Upgrading the JUNOS Software on the Routing Matrix on page 61

Redundant Host Subsystems in a Routing Matrix with a TX Matrix Plus Router

The TX Matrix Plus router and every T1600 router in the routing matrix is configured with redundant host subsystems.

- In a TX Matrix Plus router, the independent control planes are connected by two physical links between the two 10-Gigabit Ethernet ports on their respective Routing Engines. The primary link to the remote Routing Engine is at the `ixgbe0` internal 10-Gigabit Ethernet interface. The alternate link to the remote Routing Engine is at the `ixgbe1` internal 10-Gigabit Ethernet interface. If one of the two links between the host subsystems fails, both Routing Engines can use the other link for IP communication.
- In a T1600 router in a routing matrix, the independent control planes are connected by two physical links between the Gigabit Ethernet ports on their respective Routing Engines. The primary link to the remote Routing Engine is at the `bcm0` internal Ethernet interface. The alternate link to the remote Routing Engine is at the `em1` internal Ethernet interface. If one of the two links between the host subsystems fails, both Routing Engines can use the other link for IP communication.

Two Routing Engines provide redundancy and graceful Routing Engine switchover (GRES) capabilities.



NOTE: If GRES is configured, the CLI command prompt indicates Routing Engine mastership ({**master**} or {**backup**}) and physical slot number (**-re0** or **-re1**).

For example, the following CLI prompt indicates that you are logged in to the master Routing Engine in slot **RE0** of the router with hostname **mylcc3**:

```
{master}  
user@mylcc3-re0>
```

The following CLI prompt indicates that you are logged in to the backup Routing Engine in slot **RE1** of the router with hostname **mylcc3**:

```
{backup}  
user@mylcc3-re1>
```

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Chapter 2

Product Reference Information

- System Requirements for a Routing Matrix with a TX Matrix Plus Router on page 11
- Routing Matrix Terms and Acronyms on page 12

System Requirements for a Routing Matrix with a TX Matrix Plus Router

To implement a routing matrix with a TX Matrix Plus router, your system must meet the following minimum requirements:

- JUNOS Release 9.6 or later.
- One Juniper Networks TX Matrix Plus Router.
- Two to four Juniper Networks T1600 Core Routers.

To connect a T1600 router to the routing matrix, the T1600 router must contain the following components:

- Two C1800 Routing Engines—Model RE-DUO-C1800-8G
- Two LCC-CB Control Boards (CBs)—Model CB-LCC
- Five TXP-T1600 Switch Interface Boards (SIBs)—Model SIB-TXP-T1600
- One rear fan tray—Model FAN-REAR-TXP-LCC

For information about upgrading a standalone T1600 router and integrating it into a routing matrix, see the *TX Matrix Plus Router Hardware Guide*.

- Physical Interface Cards (PICs) of your choice. To view a list of supported PICs, see the *T1600 Core Router PIC Guide*.

Related Topics

- Routing Matrix with a TX Matrix Plus Router Solutions Page
- Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
- Roadmap for Configuring the Routing Matrix on page 15
- Example Configuration for the Routing Matrix on page 33
- Upgrading the JUNOS Software on the Routing Matrix on page 61

Routing Matrix Terms and Acronyms

L

line-card chassis (LCC) A T640 router or T1600 router installed in a routing matrix.

R

routing matrix A high capacity, multichassis router. The routing matrix with a TX Matrix router combines multiple T640 routers with a TX Matrix router switch fabric. The routing matrix with a TX Matrix Plus router combines multiple T1600 routers with a TX Matrix Plus router switch fabric.

S

Switch Interface Board (SIB) On T640 and T1600 routers and on TX Matrix router and TX Matrix Plus routers, a switch fabric plane component that forwards packets from a source Packet Forwarding Engine to a destination Packet Forwarding Engine.

switch-card chassis (SCC) A TX Matrix router installed in a routing matrix.

switch-fabric chassis (SFC) A TX Matrix Plus router installed in a routing matrix.

T

TX Matrix Plus router A high-speed centralized switch fabric that connects multiple T1600 routers in a routing matrix.

TX Matrix router A high-speed centralized switch fabric that connects multiple T640 routers in a routing matrix.

Part 2

Configuration

- Product Configuration on page 15
- Product Configuration Examples on page 31

Chapter 3

Product Configuration

- Roadmap for Configuring a Routing Matrix with a TX Matrix Plus Router on page 15
- Connecting to a Routing Matrix with a TX Matrix Plus Router on page 16
- Using Configuration Groups and Inheritance in a Routing Matrix with a TX Matrix Plus Router on page 18
- Using Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router on page 23
- Configuring Protocols and Other Features on a Routing Matrix with a TX Matrix Plus Router on page 24
- Configuring Chassis-Specific Features on a Routing Matrix with a TX Matrix Plus Router on page 25
- Committing Configurations on a Routing Matrix with a TX Matrix Plus Router on page 27

Roadmap for Configuring a Routing Matrix with a TX Matrix Plus Router

This topic summarizes the procedure for configuring a routing matrix with a TX Matrix Plus router.

To configure a routing matrix with a TX Matrix Plus router, complete the following tasks:

1. Connect to the Routing Engines of the routing matrix, as described in “Connecting to a Routing Matrix with a TX Matrix Plus Router” on page 16.
2. Create and apply Routing Engine configuration groups, as described in “Using Configuration Groups and Inheritance in a Routing Matrix with a TX Matrix Plus Router” on page 18.

In general, configuration groups and inheritance of the statements in configuration groups support configuration of various router components. The special Routing Engine configuration groups (**re0**, **re1**, **lccn-re0**, and **lccn-re1**, where *n* is a value from 0 through 3) offer a simple way to establish hostnames, management interfaces, and default routes for the Routing Engines in the routing matrix.

3. Adjust certain configuration statements to accommodate the global number of FPCs installed on the routing matrix, as described in “Using Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router” on page 23.

4. (Optional) Configure protocols and other features on the routing matrix, as described in “Configuring Protocols and Other Features on a Routing Matrix with a TX Matrix Plus Router” on page 24.

Other than the expanded range of FPC numbers for interfaces and the requirement to create groups for the routers, you can configure protocols in exactly the same manner as you would for other Juniper Networks routers.

5. (Optional) For T1600 routers in a routing matrix, you can configure PIC-specific features, create an alarm LCCs that do not come online, and take LCCs offline. For more information, see “Configuring Chassis-Specific Features on a Routing Matrix with a TX Matrix Plus Router” on page 25.
6. Commit configurations on the routing matrix, as described in “Committing Configurations on a Routing Matrix with a TX Matrix Plus Router” on page 27.

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Connecting to a Routing Matrix with a TX Matrix Plus Router

On a routing matrix with a TX Matrix Plus router, you can perform management and service operations on the routing matrix through a JUNOS command-line interface (CLI) session with the TX Matrix Plus router.



BEST PRACTICE: We recommend that you access a routing matrix by connecting to the master Routing Engine on the TX Matrix Plus router. Under normal operating conditions, you do not need to access or configure the T1600 routers directly.

This topic contains the following information:

- Accessing a Routing Engine by Direct Console Connection to a Serial Port on page 16
- Accessing a Routing Engine by Telnet Access to a Management Ethernet Port on page 17
- Accessing a Routing Engine by Logging In from the CLI of Another Routing Engine on page 17
- Connecting to a Backup Routing Engine on the TX Matrix Plus Router on page 18

Accessing a Routing Engine by Direct Console Connection to a Serial Port

For an in-band connection to a Routing Engine in the routing matrix, you can attach one or more management console or auxiliary devices to the appropriate serial ports on a Routing Engine in a routing matrix. On a TX Matrix Plus router, the **CONSOLE** or **AUXILIARY** serial ports are located on each control board (TXP-CB). On a T1600 router in a routing matrix, the **CONSOLE** or **AUXILIARY** serial ports are located on the

Connector Interface Panel (CIP). For more information, see the *TX Matrix Plus Router Hardware Guide*.

Accessing a Routing Engine by Telnet Access to a Management Ethernet Port

For an out-of-band management connection to a Routing Engine in a routing matrix, you can establish a Telnet session over the network connection to that Routing Engine. On a TX Matrix Plus router, the **ETHERNET** port is located on each TXP-CB. On a T1600 router in a routing matrix, the **ETHERNET** port is located on the CIP.

For a TX Matrix Plus router and for T1600 routers in a routing matrix, the CLI syntax for operational commands and configuration statements refers to the **ETHERNET** management port as the **em0** management Ethernet interface. To use **em0** as a management port, you must configure its logical port, **em0.0**, with a valid IP address. For more information, see the *JUNOS Network Interfaces Configuration Guide*.

Accessing a Routing Engine by Logging In from the CLI of Another Routing Engine

After you are logged in to one Routing Engine in a routing matrix, you can issue the **request routing-engine login** command to connect to another Routing Engine in the routing matrix. For example:

```
{master}
user@hostA-re0> request routing-engine login ?
Possible completions:
  backup          Log in to backup RE
  lcc              Log in to specific LCC (0..3)
  master          Log in to master RE
  other-routing-engine Log in to the other Routing Engine
  re0             Log in to RE0
  re1             Log in to RE1
  sfc             Log in to SFC (0..0)

{master}
user@hostA-re0> request routing-engine login sfc 0 ?
Possible completions:
  backup          Log in to backup RE
  master          Log in to master RE
  re0             Log in to RE0
  re1             Log in to RE1

{master}
user@hostA-re0> request routing-engine login lcc 3 ?
Possible completions:
  backup          Log in to backup RE
  master          Log in to master RE
  re0             Log in to RE0
  re1             Log in to RE1

{master}
user@hostA-re0> request routing-engine login lcc 3 backup re1

--- JUNOS 1.1-1111111.1 built 1111-11-11 11:11:11 UTC
% cli
```

```
{backup}
user@hostB-re1>
```

Connecting to a Backup Routing Engine on the TX Matrix Plus Router

To manage the backup Routing Engines on all routers in the routing matrix (for example, to upgrade JUNOS Software), we recommend that you connect to the backup Routing Engine of the TX Matrix Plus router.



NOTE: If you access a Routing Engine on a T1600 router, the following warning displays:

```
user@host> request routing-engine login lcc 0 re0
--- JUNOS 9.6I built 2009-07-13 8:13:04 UTC
% cli
warning: This chassis is a Line Card Chassis (LCC) in a multichassis system.
warning: Use of interactive commands should be limited to debugging.
warning: Normal CLI access is provided by the Switch Fabric Chassis (SFC).
warning: Please logout and log into the SFC to use CLI.
```

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Using Configuration Groups and Inheritance in a Routing Matrix with a TX Matrix Plus Router

To simplify configuration of individual Routing Engines in a routing matrix with a TX Matrix Plus router, you can create a configuration group for each Routing Engine in the routing matrix:

- Overview of Configuration Groups and Inheritance in a Routing Matrix on page 18
- Creating Configuration Groups in a Routing Matrix on page 19
- Applying a Configuration Group in a Routing Matrix on page 20
- Displaying Inherited Statements on page 20
- Disabling Inheritance of a Configuration Group in the Routing Matrix on page 21
- Using Special Configuration Groups for the Routing Engines in the Routing Matrix on page 21

Overview of Configuration Groups and Inheritance in a Routing Matrix

The configuration groups feature in the JUNOS Software enables you to create a group containing configuration statements and to direct the inheritance of that group's statements in the rest of the configuration. The same group can be applied to different

sections of the configuration, and different sections of one group's configuration statements can be inherited in different places in the configuration.

Configuration groups enable you to create smaller, more logically constructed configuration files, making it easier to configure and maintain the JUNOS Software. For example, you can group statements that are repeated in many places in the configuration, such as when configuring interfaces, and thereby limit updates to just the group.

Configuration groups use true inheritance, which involves a dynamic, ongoing relationship between the source of the configuration data and the target of that data. Data values changed in the configuration group are automatically inherited by the target. The target need not contain the inherited information, although the inherited values can be overridden in the target without affecting the source from which they were inherited.

This inheritance model allows you to see only the instance-specific information without seeing the inherited details. A command pipe in configuration mode allows you to display the inherited data.

Creating Configuration Groups in a Routing Matrix

For areas of your configuration to inherit configuration statements, you must first put the statements into a configuration group and then apply that group to the levels in the configuration hierarchy that require the statements.

To create configuration groups in a routing matrix with a TX Matrix Plus router, you can include the **groups** statement at the [edit] hierarchy level on the TX Matrix Plus router:

```
[edit]
groups {
  group-name {
    ...configuration-data...
  }
}
```

The **group-name** statement specifies the name of a configuration group. You can configure more than one configuration group by specifying multiple **group-name** statements at the [edit groups] hierarchy level.

You cannot use the prefix **junos-** in a group name because it is reserved for use by the JUNOS Software. One reason for the naming restriction is a configuration group called **junos-defaults**. This preset configuration group is applied to the configuration automatically. You cannot modify or remove the **junos-defaults** configuration group. For more information about the JUNOS default configuration group, see “Using JUNOS Defaults Groups” in the *JUNOS CLI User Guide*.

Similarly, the configuration group **juniper-ais** is reserved exclusively for Juniper Advanced Insight Solutions (AIS)-related configuration. For more information about the **juniper-ais** configuration group, see the *Juniper Networks Advanced Insight Solutions Guide*.

Applying a Configuration Group in a Routing Matrix

To have a configuration inherit the statements in a configuration group, include the **apply-groups** statement on the TX Matrix Plus router:

```
apply-groups [ group-names ];
```

Include the **apply-groups [group-names]** statement anywhere in the configuration that the configuration statements contained in a configuration group are needed.

If you specify more than one group name, list them in order of inheritance priority. The configuration statements in the first group take priority over configuration statements in subsequent groups.

You can include only one **apply-groups** statement at each specific level of the configuration hierarchy. The **apply-groups** statement at a specific hierarchy level lists the configuration groups to be added to the containing statement's list of configuration groups.

Values specified at the specific hierarchy level override values inherited from the configuration group.

Groups listed in nested **apply-groups** statements take priority over groups in outer statements. In the following example, the BGP neighbor 10.0.0.1 inherits configuration data from group one first, then from groups two and three. Configuration data in group one overrides data in any other group. Data from group ten is used only if a statement is not contained in any other group.

```
apply-groups [ eight nine ten ];
protocols {
  apply-groups seven;
  bgp {
    apply-groups [ five six ];
    group some-bgp-group {
      apply-groups four;
      neighbor 10.0.0.1 {
        apply-groups [ one two three ];
      }
    }
  }
}
```

Displaying Inherited Statements

Configuration groups can add some confusion regarding the actual values used by the router, because configuration data can be inherited from configuration groups. To view the actual values used by the router, use the **display inheritance** command after the pipe in a **show** command. This command displays the inherited statements at the level at which they are inherited and the group from which they have been inherited.

The following example shows part of the output of the `show | display inheritance` command that shows configuration statements inherited from the special configuration group `re0`, which contains configuration statements that apply to the Routing Engine in slot 0 (labeled RE0) on the TX Matrix Plus router:

```
[edit]
user@host# show | display inheritance
...
system {
  ## 'mysfc0' was inherited from group 're0'
  host-name mysfc0;
  ## 'backup-router' was inherited from group 're0'
  backup-router 192.168.35.254;
}
interfaces {
  ## 'em0' was inherited from group 're0'
  em0 {
    ## '0' was inherited from group 're0'
    unit 0 {
      ## 'inet' was inherited from group 're0'
      family inet {
        ## '192.168.35.95/24' was inherited from group 're0'
        address 192.168.35.95/24;
      }
    }
  }
}
...
```

Disabling Inheritance of a Configuration Group in the Routing Matrix

To disable inheritance of a configuration group at any level except the top level of the hierarchy, include the `apply-groups-except` statement:

```
apply-groups-except [ group-names ];
```

This statement is useful when you use the `apply-group` statement at a specific hierarchy level but also want to override the values inherited from the configuration group for a specific parameter.

Using Special Configuration Groups for the Routing Engines in the Routing Matrix

Using special configuration group names for all Routing Engines in the routing matrix allows you to configure the individual Routing Engines in each router differently. Parameters that are not configured at the `[edit groups]` hierarchy level apply to all Routing Engines in the routing matrix.

To configure configuration groups and apply inheritance for the Routing Engines in a routing matrix with a TX Matrix Plus router, you can include the `groups` statement at the `[edit]` hierarchy level on the TX Matrix Plus router and then include special configuration group names for all the Routing Engines in the routing matrix:

```
[edit]
groups {
  re0 { # Create the group for the SFC master Routing Engine.
```

```

system {
    host-name sfc0-re0-hostname;
    backup-router ip-address;
    ...
}
interfaces {
    em0 {
        unit logical-unit-number {
            ...
        }
    }
}
}
re1 { # Create the group for the SFC backup Routing Engine.
    system {
        host-name sfc0-re1-hostname;
        backup-router ip-address;
        ...
    }
    interfaces {
        em0 {
            unit logical-unit-number {
                ...
            }
        }
    }
}
lccn-re0 { # Create the group for the master Routing Engine in a specific LCC.
    system {
        host-name lccN-re0-hostname;
        backup-router ip-address;
        ...
    }
    interfaces {
        em0 {
            unit logical-unit-number {
                ...
            }
        }
    }
}
lccn-re1 { # Create the group for the backup Routing Engine in a specific LCC.
    system {
        host-name lccN-re1-hostname;
        backup-router ip-address;
        ...
    }
    interfaces {
        em0 {
            unit logical-unit-number {
                ...
            }
        }
    }
}
...additional-routing-engine-group-configurations...
}
apply-groups [ re0 re1 lccn-re0 lccn-re1 ... ]; # Enable inheritance of the groups.

```

For routers in a routing matrix with a TX Matrix Plus router, you can specify the following special group names:

- **re0**—Configuration statements apply to the Routing Engine in slot 0 (labeled RE0) on the TX Matrix Plus router.
- **re1**—Configuration statements apply to the Routing Engine in slot 1 (labeled RE1) on the TX Matrix Plus router.
- **lccn-re0**—Configuration statements apply to the Routing Engine in slot 0 (labeled RE0) on the T1600 router identified as **lccn**, where *n* is a value from 0 through 3.
- **lccn-re1**—Configuration statements applied to the Routing Engine in slot 1 (labeled RE1) on the T1600 router identified **lccn**, where *n* is a value from 0 through 3.

Because the configuration statements in the special configuration groups for Routing Engines apply to specific Routing Engines in the routing matrix, you can create a single configuration for all of the routers, with each Routing Engine using only the configuration statements that apply to it.

Each Routing Engine configuration group contains at a minimum the configuration for the Routing Engine hostname and the management Ethernet interface (**em0**). In addition, if each Routing Engine uses a different management interface, the group also should contain the configuration for the backup router and static routes.

Note that apply groups can be nested. For example, any configuration statements that are common to **lcc0-re0** and **lcc0-re1** can be included in a separate configuration group and then added as an apply group to the **lcc0-re0** and **lcc0-re1** groups, which in turn are applied to the main configuration.

For more information about configuring and applying configuration groups, see the *JUNOS CLI User Guide*.

Related Topics

- Routing Matrix with a TX Matrix Plus Router Solutions Page
- Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
- Roadmap for Configuring the Routing Matrix on page 15
- Example Configuration for the Routing Matrix on page 33
- Upgrading the JUNOS Software on the Routing Matrix on page 61

Using Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router

When configuring interfaces on a routing matrix with a TX Matrix Plus router, you must adjust the routing matrix configuration to accommodate increased FPC numbers used in interface names.

For example, if you have a Gigabit Ethernet interface installed in FPC slot 7, PIC slot 0, port 0 of T1600 router lcc 3, you can configure this interface on the TX Matrix Plus router by including the `ge-31/0/0` statement at the `[edit interfaces]` hierarchy level:

```
[edit]
interfaces {
  ge-31/0/0 { # In a standalone T1600 router, the interface is 'ge-7/0/0'.
    unit 0 {
      family inet {
        address ip-address;
      }
    }
  }
}
```

For more information about converting FPC hardware slot numbers on a T1600 router to the global FPC numbers used for interfaces in a routing matrix and vice versa, see “Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router” on page 5 and “Displaying Chassis Physical Locations for a Routing Matrix with a TX Matrix Plus Router” on page 103.



NOTE: When you include the `fpc` statement at the `[edit chassis lcc lcc-number]` hierarchy level, specify the FPC hardware slot number (0 through 7) as labeled on the T1600 router chassis. Do not specify the corresponding global FPC number (0 through 31).

For more information about physically connecting a TX Matrix Plus router and two to four T1600 routers together in a routing matrix, *TX Matrix Plus Router Hardware Guide* For more information about the interface naming conventions for a routing matrix, see the *JUNOS Network Interfaces Configuration Guide*.

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Configuring Protocols and Other Features on a Routing Matrix with a TX Matrix Plus Router

Other than the expanded range of FPC numbers for interfaces and the requirement to create groups for the T1600 routers, the configuration of a routing matrix with a TX Matrix Plus router is exactly the same as for all other Juniper Networks routers. You can configure routing protocols, Multiprotocol Label Switching (MPLS) applications, virtual private networks (VPNs), routing and forwarding options, and other software features as usual.

For more information about configuring JUNOS-based routers, see the JUNOS Software configuration guides.

Related Topics

- Routing Matrix with a TX Matrix Plus Router Solutions Page
- Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
- Roadmap for Configuring the Routing Matrix on page 15
- Example Configuration for the Routing Matrix on page 33
- Upgrading the JUNOS Software on the Routing Matrix on page 61

Configuring Chassis-Specific Features on a Routing Matrix with a TX Matrix Plus Router

You can configure chassis-specific features on specific T1600 routers in a routing matrix with a TX Matrix Plus router:

- Configuring PIC-Specific Features on a T1600 Router in a Routing Matrix on page 25
- Configuring an Alarm to Trigger if an LCC in the Routing Matrix Does Not Come Online on page 26
- Configuring an LCC in the Routing Matrix to Stay Offline on page 26
- For More Information on page 26

Configuring PIC-Specific Features on a T1600 Router in a Routing Matrix

You can configure PIC-specific features, such as SONET/SDH framing, on specific T1600 routers in a routing matrix with a TX Matrix Plus router. To configure, include the `lcc lcc-number` statement at the [edit chassis] hierarchy level and specify the PIC-specific feature to configure.

```
[edit]
chassis {
  lcc lcc-number {
    fpc slot-number { # Use the T1600 router FPC hardware slot number.
      pic pic-number {
        ...pic-specific-configuration...
      }
    }
  }
}
```



NOTE: When you include statements at the [edit chassis lcc lcc-number] hierarchy level, specify the actual FPC hardware slot number as labeled on the T1600 router chassis. Do not specify the corresponding global FPC number (0 through 31). Global FPC numbering, described in “Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router” on page 5 and in “Displaying Chassis Physical Locations for a Routing Matrix with a TX Matrix Plus Router” on page 103, is used to specify the interface in a routing matrix.

Configuring an Alarm to Trigger if an LCC in the Routing Matrix Does Not Come Online

By default, the JUNOS Software allows all T1600 routers in the routing matrix to come online. Optionally, you can configure the TX Matrix Plus router to generate an alarm if the T1600 routers in the routing matrix do not come online.

To configure, include the `online-expected` statement at the `[edit chassis lcc lcc-number]` hierarchy level on the TX Matrix Plus router:

```
[edit]
chassis{
  lcc lcc-number {
    online-expected;
  }
}
```

If you do not include the `online-expected` statement at any `[edit chassis lcc lcc-number]` hierarchy level on the TX Matrix Plus router, no alarm triggers if any T1600 fails to come online.

Configuring an LCC in the Routing Matrix to Stay Offline

If you do not want a T1600 router to be part of the routing matrix, such as while you are performing maintenance on that LCC, you can configure the router to be offline.

To configure a T1600 router to be offline, include the `offline` statement at the `[edit chassis lcc lcc-number]` hierarchy level:

```
[edit]
chassis {
  lcc lcc-number {
    offline;
  }
}
```

When you are ready to bring the T1600 router back online, delete the `offline` configuration statement at the `[edit chassis lcc number]` hierarchy level.



NOTE: If you do not include the `offline` statement at any `[edit chassis lcc lcc-number]` hierarchy level on the TX Matrix Plus router, any T1600 router that is part of the routing matrix is allowed to come online.

For More Information

For more information about chassis-specific statements, see the *JUNOS System Basics Configuration Guide*.

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15

- Example Configuration for the Routing Matrix on page 33
- Upgrading the JUNOS Software on the Routing Matrix on page 61

Committing Configurations on a Routing Matrix with a TX Matrix Plus Router

On a routing matrix with a TX Matrix Plus router, you must commit configuration changes on the TX Matrix Plus router rather than on the individual T1600 routers. All configuration changes you commit on the TX Matrix Plus router are distributed to all the T1600 routers in the routing matrix and override any configuration changes committed directly on a T1600 router.



NOTE: If you commit a configuration directly on a T1600 router in a routing matrix, the configuration is not distributed to the TX Matrix Plus router or to the other T1600 routers in the routing matrix.

There are three main ways to commit configurations on a TX Matrix Plus router:

- Committing a Configuration to Both Master and Backup Routing Engines in the Routing Matrix on page 27
- Committing a Configuration to the Master Routing Engines (Only) in the Routing Matrix on page 28
- Synchronizing to the Configuration on the Other Routing Engine on page 28

Committing a Configuration to Both Master and Backup Routing Engines in the Routing Matrix

To commit the same configuration to both the master and backup Routing Engines in the routing matrix, issue the `commit operational` command with the `synchronize` option.

The Routing Engine on which you execute the `commit synchronize` command (the requesting Routing Engine) copies and loads its candidate configuration to the other Routing Engine (the responding Routing Engine). Both Routing Engines then perform a syntax check on the candidate configuration file being committed. If no errors are found, the configuration is activated and becomes the current operational configuration on both Routing Engines.

The `commit synchronize` command makes the active or applied configuration the same for both Routing Engines with the exception of two special configuration groups for Routing Engines:

- Configuration statements specified in the `re0` configuration group are applied only to Routing Engines in slot 0 (designated `re0`).
- Configuration statements specified in the `re1` configuration group are applied only to Routing Engines in slot 1 (designated `re0`).



NOTE: If you do not synchronize the configurations between two Routing Engines and one of them fails, the router may not forward traffic correctly because the backup Routing Engine may have a different configuration.

The following example shows command output for the **commit** command issued on the TX Matrix Plus router with the **synchronize** option:

```
[edit groups]
user@host> set re0 system hostname sfc0-re0-hostname

user@sfc0-re0-hostname> commit synchronize;
sfc0-re0:
configuration check succeeds
lcc0-re1:
commit complete
lcc0-re0:
commit complete
lcc2-re1:
commit complete
lcc2-re0:
commit complete
sfc0-re1:
commit complete
sfc0-re0:
commit complete
```

Committing a Configuration to the Master Routing Engines (Only) in the Routing Matrix

In a routing matrix with a TX Matrix Plus router, issuing the basic form of the **commit** operational command on the TX Matrix Plus router commits the candidate configuration only to the master Routing Engines in the routing matrix.

The following example shows command output for the basic form of the **commit** command:

```
user@host# commit
sfc0-re0:
configuration check succeeds
lcc0-re0:
commit complete
lcc1-re0:
commit complete
sfc0-re0:
commit complete
```

Synchronizing to the Configuration on the Other Routing Engine

In a routing matrix with at TX Matrix Plus router, issuing the **commit synchronize** command with the **force** option directs one Routing Engine to synchronize its configuration with the other.



NOTE: We recommend that you use the **force** option only if you are unable to resolve the issues that caused the **commit synchronize** command to fail.

The Routing Engine on which you issue this command (the requesting Routing Engine) copies and loads its candidate configuration to the other Routing Engine (the responding Routing Engine). Both Routing Engines then perform a syntax check on the candidate configuration file being committed. If no errors are found, the configuration is activated and becomes the current operational configuration on both Routing Engines.

The **commit synchronize** command does not work if the responding Routing Engine has uncommitted configuration changes. However, you can enforce commit synchronization on the Routing Engines by using the **force** option.



NOTE: When you issue the **commit synchronize** command with the **force** option from one Routing Engine, the configuration sessions on the other Routing Engine will be terminated and its configuration synchronized with that on the Routing Engine from which you issued the command.

Related Topics

- Routing Matrix with a TX Matrix Plus Router Solutions Page
- Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
- Roadmap for Configuring the Routing Matrix on page 15
- Example Configuration for the Routing Matrix on page 33
- Upgrading the JUNOS Software on the Routing Matrix on page 61

Chapter 4

Product Configuration Examples

- Merging Examples on page 31
- Example Configuration for a Routing Matrix with a TX Matrix Plus Router on page 33
- Verifying the Configuration of a Routing Matrix with a TX Matrix Plus Router on page 40

Merging Examples

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 on page 31
- Merging a Snippet on page 32
- For More Information on page 32

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 router.

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 router.

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

```

}
interfaces {
  em0 {
    disable;
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
    }
  }
}

```

2. Merge the contents of the file into your router 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 router.

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 router.

```

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 router 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

For more information about using the **load** command, see Loading a Configuration from a File and Examples: Loading a Configuration from a File in the *JUNOS CLI User Guide*.

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Example Configuration for a Routing Matrix with a TX Matrix Plus Router

The following sections describe an example configuration for a routing matrix with a TX Matrix Plus router:

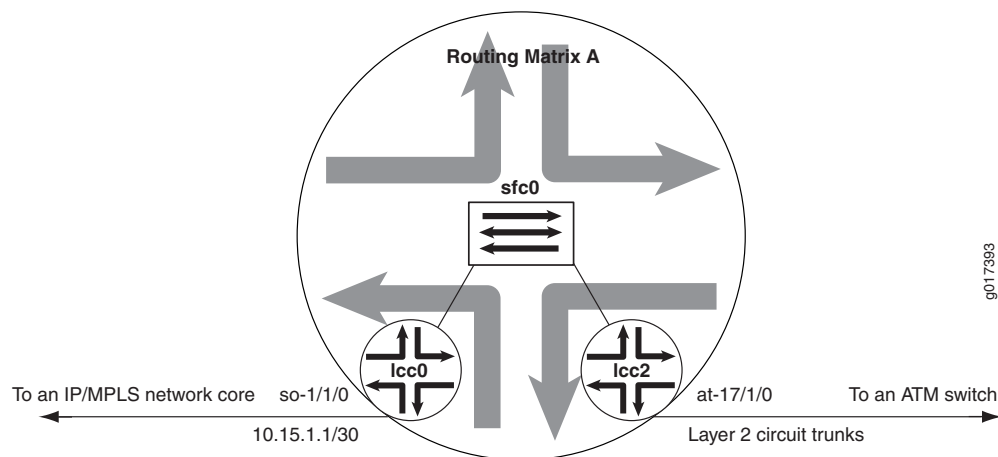
- Example Routing Matrix Topology on page 33
- Example Routing Matrix Configuration on page 34
- Verifying the Example Configuration on page 39

Example Routing Matrix Topology

This example is based on **Routing Matrix A**, a basic routing matrix consisting of a TX Matrix Plus router (**sfc0**) and two T1600 routers (**lcc0** and **lcc2**). Figure 2 on page 33 shows that **Routing Matrix A** is acting as a provider edge (PE) router in a Layer 2 circuit network. SONET interface **so-1/1/0** in the T1600 router designated as **lcc0** connects to an IP/MPLS core network. Asynchronous Transfer Mode 2 (ATM2) intelligent queuing (IQ) interface **at-17/1/0** in the T1600 router designated as **lcc2** runs Layer 2 circuit trunk mode to connect to an ATM switch.

For more information about Layer 2 circuit networks, see the *JUNOS VPNs Configuration Guide*.

Figure 2: Example Topology of a Routing Matrix with a TX Matrix Plus Router



Note the following key considerations for this routing matrix configuration:

- For most operations, you will manage the routing matrix as a single router, executing operational commands and committing configurations on the TX Matrix Plus router designated as **sfc0**.
- Create configuration groups for each Routing Engine in the routing matrix by using the special configuration groups for a routing matrix consisting of routers with dual Routing Engines: **re0**, **re1**, **lcc0-re0**, **lcc2-re0**, **lcc0-re1**, and **lcc2-re1**. In these Routing Engine configuration groups, configure hostnames, default routes, and management Ethernet interfaces.
- To configure interfaces, use the routing matrix FPC numbering convention of slots **0** through **31**.
- To enable ATM2 IQ trunk mode and other chassis-based commands, include the **lcc lcc-number** statement at the **[edit chassis]** hierarchy level and use the hardware FPC slot numbers **0** through **7** of node **lcc2**.
- Configure most other processes as usual, such as routing, class of service (CoS), and firewalls.

Example Routing Matrix Configuration

To configure the example Routing Matrix A described in the previous section, commit the following configuration at the TX Matrix Plus router:

```
[edit]
groups { # Create special configuration groups in a routing matrix.
  re0 { # Create a group for the master Routing Engine on TX Matrix Plus router sfc0.
    system {
      host-name mysfc0;
      backup-router 192.168.17.254;
    }
    interfaces { # Configure an IP address for the management interface logical port.
      em0 {
        unit 0 {
          family inet {
            address 192.168.77.158/21;
          }
        }
      }
    }
  }
re1 { # Create a group for the backup Routing Engine on TX Matrix Plus router sfc0.
  system {
    host-name mysfc0_alt_re;
    backup-router 192.168.17.254;
  }
  interfaces { # Configure an IP address for the management interface logical port.
    em0 {
      unit 0 {
        family inet {
          address 192.168.77.168/21;
        }
      }
    }
  }
}
```

```

    }
  }
}
lcc0-re0 { # Create a group for the master Routing Engine on T1600 router lcc0.
  system {
    host-name mylcc0;
    backup-router 192.168.17.254 destination [10.0.0.0/8 192.168.0.0/16];
  }
  interfaces { # Configure an IP address for the management interface logical port.
    em0 {
      unit 0 {
        family inet {
          address 192.168.77.157/21;
        }
      }
    }
  }
}
lcc0-re1 { # Create a group for the backup Routing Engine on T1600 router lcc0.
  system {
    host-name mylcc0_alt_re;
    backup-router 192.168.17.254 destination [10.0.0.0/8 192.168.0.0/16];
  }
  interfaces { # Configure an IP address for the management interface logical port.
    em0 {
      unit 0 {
        family inet {
          address 192.168.77.169/21;
        }
      }
    }
  }
}
lcc2-re0 { # Create a group for the master Routing Engine on T1600 router lcc2.
  system {
    host-name mylcc2;
    backup-router 192.168.17.254 destination [10.0.0.0/8 192.168.0.0/16];
  }
  interfaces { # Configure an IP address for the management interface logical port.
    em0 {
      unit 0 {
        family inet {
          address 192.168.77.159/21;
        }
      }
    }
  }
}
lcc2-re1 { # Create a group for the backup Routing Engine on T1600 router lcc2.
  system {
    host-name mylcc2_alt_re;
    backup-router 192.168.17.254 destination [10.0.0.0/8 192.168.0.0/16];
  }
  interfaces { # Configure an IP address for the management interface logical port.
    em0 {
      unit 0 {

```

```

        family inet {
            address 192.168.77.192/21;
        }
    }
}

apply-groups [ re0 re1 lcc0-re1 lcc2-re1 lcc0-re0 lcc2-re0 ]; # Enable inheritance.
system { # Configure system management properties.
    syslog {
        file messages {
            any any;
        }
    }
}

chassis { # You must apply chassis commands to a specific T1600 router.
    lcc 2 { # Specify the T1600 router and the FPC hardware slot of the node.
        fpc 1 { # This FPC is equivalent to slot 17 in the routing matrix.
            pic 1 {
                atm-l2circuit-mode {
                    trunk nni;
                }
            }
        }
    }
}

interfaces {
    so-1/1/0 { # This is a SONET interface at FPC 1, PIC 1, port 0
        mtu 9192; # on the T1600 router LCC0.
        unit 0 {
            family inet {
                address 10.15.1.1/30 {
                    destination 10.15.1.2;
                }
            }
            family iso;
            family mpls {
                filter {
                    input filter_1;
                }
            }
        }
    }
    at-17/1/0 { # This is an ATM2 IQ interface at FPC 1, PIC 1, port 0
        encapsulation atm-ccc-cell-relay; # on the T1600 router LCC2.
        atm-options {
            pic-type atm2;
            scheduler-maps { # CoS on an ATM2 IQ PIC works the same in a routing matrix.
                cos1 { # as it does in a standalone T1600 router.
                    forwarding-class ubr {
                        priority low;
                        transmit-weight percent 25;
                    }
                    forwarding-class nrtvbr {
                        priority low;
                    }
                }
            }
        }
    }
}

```

```

        transmit-weight percent 25;
    }
    forwarding-class rtvbr {
        priority low;
        transmit-weight percent 25;
    }
    forwarding-class cbr {
        priority high;
        transmit-weight percent 25;
    }
}
cos2 {
    forwarding-class ubr {
        priority low;
        transmit-weight percent 10;
    }
    forwarding-class nrtvbr {
        priority low;
        transmit-weight percent 20;
    }
    forwarding-class rtvbr {
        priority low;
        transmit-weight percent 30;
    }
    forwarding-class cbr {
        priority high;
        transmit-weight percent 40;
    }
}
cos3 {
    forwarding-class ubr {
        priority low;
        transmit-weight percent 40;
    }
    forwarding-class nrtvbr {
        priority low;
        transmit-weight percent 30;
    }
    forwarding-class rtvbr {
        priority low;
        transmit-weight percent 20;
    }
    forwarding-class cbr {
        priority high;
        transmit-weight percent 10;
    }
}
}
}
unit 0 {
    trunk-id 0;
    trunk-bandwidth 10m;
    cell-bundle-size 2;
}
unit 1 {
    trunk-id 1;

```

```

        trunk-bandwidth 10m;
        cell-bundle-size 1;
        atm-scheduler-map cos1;
    }
    unit 2 {
        trunk-id 2;
        trunk-bandwidth 10m;
        cell-bundle-size 2;
        atm-scheduler-map cos2;
    }
    unit 3 {
        trunk-id 3;
        trunk-bandwidth 10m;
        cell-bundle-size 3;
        atm-scheduler-map cos3;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 127.0.0.1/32;
            address 10.255.77.158/32 {
                primary;
            }
        }
        family iso {
            address 47.0005.80ff.f800.0000.0108.0001.0102.5507.0158.00;
        }
        family inet6 {
            address 2001:db8::10:255:77:158/32 {
                primary;
            }
        }
    }
}
}

protocols { # You can configure protocols in the routing matrix as usual.
    mpls {
        interface so-1/1/0.0;
    }
    isis {
        interface so-1/1/0.0;
        interface lo0.0;
    }
    ldp {
        interface so-1/1/0.0;
        interface lo0.0;
    }
    l2circuit {
        neighbor 10.255.71.97 {
            interface at-17/1/0.0 {
                virtual-circuit-id 100;
            }
            interface at-17/1/0.1 {
                virtual-circuit-id 101;
            }
        }
    }
}

```

```

        interface at-17/1/0.2 {
            virtual-circuit-id 102;
        }
        interface at-17/1/0.3 {
            virtual-circuit-id 103;
        }
    }
}

class-of-service { # You can configure CoS in the routing matrix as usual.
    forwarding-classes {
        queue 0 ubr;
        queue 1 nrtvbr;
        queue 2 rtvbr;
        queue 3 cbr;
    }
    traceoptions {
        flag all;
    }
}

firewall { # You can configure firewalls in the routing matrix as usual.
    family mpls {
        filter filter_1 {
            term plp0 {
                from {
                    exp [ 0 2 4 6 ];
                }
                then {
                    count LOW;
                    loss-priority low;
                }
            }
            term plp1 {
                from {
                    exp [ 1 3 5 7 ];
                }
                then {
                    count HIGH;
                    loss-priority high;
                }
            }
        }
    }
}

```

Verifying the Example Configuration

After you commit the example configuration, described in the previous section, to example Routing Matrix A, verify proper operation of the routing matrix. For the example routing matrix topology and configuration described in this topic, you would issue various forms of the following commands on the TX Matrix Plus router:

- `show chassis alarms < lcc lcc-number | sfc sfc-number >`
- `show chassis craft-interface < lcc lcc-number | sfc sfc-number >`

- `show chassis ethernet-switch < lcc lcc-number | sfc sfc-number >`
- `show chassis hardware < lcc lcc-number | sfc sfc-number >`
- `show chassis fpc < lcc lcc-number >`
- `show chassis lccs`
- `show chassis location < fpc | interface | lcc lcc-number | sfc sfc-number >`
- `show chassis routing-engine < lcc lcc-number | sfc sfc-number >`
- `show chassis sibs < lcc lcc-number | sfc sfc-number >`
- `show interfaces terse`
- `show route summary`
- `show system uptime < all-lcc | lcc lcc-number | sfc sfc-number >`
- `show version < all-lcc | lcc lcc-number | sfc sfc-number >`

For detailed descriptions of how to use these commands to verify the routing matrix configuration, see “Verifying the Configuration of a Routing Matrix with a TX Matrix Plus Router” on page 40.

Related Topics

- Routing Matrix with a TX Matrix Plus Router Solutions Page
- Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
- Roadmap for Configuring the Routing Matrix on page 15
- Verifying the Configuration of the Routing Matrix on page 40
- Merging Examples on page 31
- Upgrading the JUNOS Software on the Routing Matrix on page 61

Verifying the Configuration of a Routing Matrix with a TX Matrix Plus Router

In general, when you issue standard operational commands on a TX Matrix Plus router, you receive output from the master Routing Engines of all components in the routing matrix. To display information for the TX Matrix Plus router only, include the `sfc sfc-number` option. To limit the output of information for a specific T1600 router within the routing matrix, include the `lcc lcc-number` option. To display information for all T1600 routers within the routing matrix (selected commands only), include the `all-lcc` option. Any exceptions to this general rule are mentioned next to the appropriate commands.

The following sections contain examples of specific **show** operational commands you can use to verify the configuration of the example **Routing Matrix A**, described in “Example Configuration for a Routing Matrix with a TX Matrix Plus Router” on page 33:

- Displaying JUNOS Software Versions on page 41
- Displaying Configured Interfaces on page 44
- Displaying Available Routes on page 45
- Displaying Alarms and System Uptime on page 46

- Displaying Craft Interface Messages on page 46
- Displaying System Uptime on page 49
- Displaying Chassis Hardware and Status on page 49

Displaying JUNOS Software Versions

The `show version` command provides an excellent example of how you can select output for various components of the routing matrix with a TX Matrix Plus router. If the TX Matrix Plus router (`sfc sfc-number`) or a T1600 router (`lcc lcc-number`) is not specified in the command, the command displays output for all components.

```
user@host> show version ?
Possible completions:
<[Enter]>      Execute this command
brief          Display brief output
detail         Display detailed output
invoke-on      Remote command execution
lcc            Show software version on specific LCC (0..3)
sfc            Show software version on SFC (0..0)
|              Pipe through a command
```

You can display information about individual software components in the TX Matrix Plus router, in a specific T1600 router, or the entire routing matrix:

- Displaying JUNOS Software Versions for All Routers on page 41
- Displaying JUNOS Software Version for the SFC Only on page 42
- Displaying JUNOS Software Version for a Specific LCC on page 43
- Displaying JUNOS Software Versions for All LCCs on page 43

Displaying JUNOS Software Versions for All Routers

To display the software version for all routing matrix components, issue the `show version` command on the TX Matrix Plus router:

```
user@mysfc0> show version
sfc0-re0:
-----
Hostname: mysfc0
Model: txp
JUNOS Base OS boot [9.6-20090713.1]
JUNOS Base OS Software Suite [9.6-20090713.1]
JUNOS Kernel Software Suite [9.6-20090713.1]
JUNOS Crypto Software Suite [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.1]
JUNOS Online Documentation [9.6-20090713.1]
JUNOS Voice Services Container package [9.6-20090713.1]
JUNOS Border Gateway Function package [9.6-20090713.1]
JUNOS Services ACL Container package [9.6-20090713.1]
JUNOS Services LL-PDF Container package [9.6-20090713.1]
JUNOS Services Stateful Firewall [9.6-20090713.1]
JUNOS AppId Services [9.6-20090713.1]
JUNOS IDP Services [9.6-20090713.1]
```

JUNOS Routing Software Suite [9.6-20090713.1]

lcc0-re0:

```
-----
Hostname: mylcc0
Model: t1600
JUNOS Base OS boot [9.6-20090713.1]
JUNOS Base OS Software Suite [9.6-20090713.1]
JUNOS Kernel Software Suite [9.6-20090713.1]
JUNOS Crypto Software Suite [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.1]
JUNOS Online Documentation [9.6-20090713.1]
JUNOS Voice Services Container package [9.6-20090713.1]
JUNOS Border Gateway Function package [9.6-20090713.1]
JUNOS Services AACL Container package [9.6-20090713.1]
JUNOS Services LL-PDF Container package [9.6-20090713.1]
JUNOS Services Stateful Firewall [9.6-20090713.1]
JUNOS AppId Services [9.6-20090713.1]
JUNOS IDP Services [9.6-20090713.1]
JUNOS Routing Software Suite [9.6-20090713.1]
```

lcc2-re0:

```
-----
Hostname: mylcc2
Model: t1600
JUNOS Base OS boot [9.6-20090713.1]
JUNOS Base OS Software Suite [9.6-20090713.1]
JUNOS Kernel Software Suite [9.6-20090713.1]
JUNOS Crypto Software Suite [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.1]
JUNOS Online Documentation [9.6-20090713.1]
JUNOS Voice Services Container package [9.6-20090713.1]
JUNOS Border Gateway Function package [9.6-20090713.1]
JUNOS Services AACL Container package [9.6-20090713.1]
JUNOS Services LL-PDF Container package [9.6-20090713.1]
JUNOS Services Stateful Firewall [9.6-20090713.1]
JUNOS AppId Services [9.6-20090713.1]
JUNOS IDP Services [9.6-20090713.1]
JUNOS Routing Software Suite [9.6-20090713.1]
```

Displaying JUNOS Software Version for the SFC Only

To display the software version for the TX Matrix Plus router only, include the *sfc number* option:

```
user@mysfc> show version sfc 0
sfc0-re0
```

```
-----
Hostname: mysfc0
Model: txp
JUNOS Base OS boot [9.6-20090713.1]
JUNOS Base OS Software Suite [9.6-20090713.1]
JUNOS Kernel Software Suite [9.6-20090713.1]
JUNOS Crypto Software Suite [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.1]
```

```

JUNOS Online Documentation [9.6-20090713.1]
JUNOS Voice Services Container package [9.6-20090713.1]
JUNOS Border Gateway Function package [9.6-20090713.1]
JUNOS Services AACL Container package [9.6-20090713.1]
JUNOS Services LL-PDF Container package [9.6-20090713.1]
JUNOS Services Stateful Firewall [9.6-20090713.1]
JUNOS AppId Services [9.6-20090713.1]
JUNOS IDP Services [9.6-20090713.1]
JUNOS Routing Software Suite [9.6-20090713.1]

```

Displaying JUNOS Software Version for a Specific LCC

To display the software version for a specific T1600 router, include the `lcc` option:

```

user@host> show version lcc 0
lcc0-re0:
-----
Hostname: mylcc0
Model: t1600
JUNOS Base OS boot [9.6-20090713.1]
JUNOS Base OS Software Suite [9.6-20090713.1]
JUNOS Kernel Software Suite [9.6-20090713.1]
JUNOS Crypto Software Suite [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.1]
JUNOS Online Documentation [9.6-20090713.1]
JUNOS Voice Services Container package [9.6-20090713.1]
JUNOS Border Gateway Function package [9.6-20090713.1]
JUNOS Services AACL Container package [9.6-20090713.1]
JUNOS Services LL-PDF Container package [9.6-20090713.1]
JUNOS Services Stateful Firewall [9.6-20090713.1]
JUNOS AppId Services [9.6-20090713.1]
JUNOS IDP Services [9.6-20090713.1]
JUNOS Routing Software Suite [9.6-20090713.1]

```

Displaying JUNOS Software Versions for All LCCs

To display the software versions for all T1600 routers, include the `all-lcc` option:

```

user@host> show version all-lcc
lcc0-re0:
-----
Hostname: mylcc0
Model: t1600
JUNOS Base OS boot [9.6-20090713.1]
JUNOS Base OS Software Suite [9.6-20090713.1]
JUNOS Kernel Software Suite [9.6-20090713.1]
JUNOS Crypto Software Suite [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.1]
JUNOS Online Documentation [9.6-20090713.1]
JUNOS Voice Services Container package [9.6-20090713.1]
JUNOS Border Gateway Function package [9.6-20090713.1]
JUNOS Services AACL Container package [9.6-20090713.1]
JUNOS Services LL-PDF Container package [9.6-20090713.1]

```

```
JUNOS Services Stateful Firewall [9.6-20090713.1]
JUNOS AppId Services [9.6-20090713.1]
JUNOS IDP Services [9.6-20090713.1]
JUNOS Routing Software Suite [9.6-20090713.1]
```

lcc2-re0

```
-----
Hostname: mylcc0
Model: t1600
JUNOS Base OS boot [9.6-20090713.1]
JUNOS Base OS Software Suite [9.6-20090713.1]
JUNOS Kernel Software Suite [9.6-20090713.1]
JUNOS Crypto Software Suite [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.1]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.1]
JUNOS Online Documentation [9.6-20090713.1]
JUNOS Voice Services Container package [9.6-20090713.1]
JUNOS Border Gateway Function package [9.6-20090713.1]
JUNOS Services ACL Container package [9.6-20090713.1]
JUNOS Services LL-PDF Container package [9.6-20090713.1]
JUNOS Services Stateful Firewall [9.6-20090713.1]
JUNOS AppId Services [9.6-20090713.1]
JUNOS IDP Services [9.6-20090713.1]
JUNOS Routing Software Suite [9.6-20090713.1]
```

Displaying Configured Interfaces

Although individual FPCs are installed in each of the T1 600 routers, the routing matrix is designed to collect interface information centrally at the TX Matrix Plus router. To display available interfaces in the routing matrix, issue a **show interfaces** command on the TX Matrix Plus router:

```
user@host> show interfaces terse
```

Interface	Admin	Link	Proto	Local	Remote
so-1/0/0	up	up			
so-1/1/0	up	up			
so-1/1/0.0	up	up	inet	10.15.1.1	--> 10.15.1.2
			iso		
			mpls		
so-1/3/0	up	down			
at-2/1/0	up	up			
ge-2/2/0	up	up			
so-3/3/0	up	up			
so-3/3/1	up	up			
so-3/3/2	up	down			
so-3/3/3	up	down			
so-16/0/0	up	down			
so-16/0/1	up	down			
so-16/0/2	up	down			
so-16/0/3	up	up			
ge-16/1/0	up	down			
so-17/0/0	up	down			
at-17/1/0	up	up			
at-17/1/0.0	up	up	ccc		
at-17/1/0.1	up	up	ccc		
at-17/1/0.2	up	up	ccc		
at-17/1/0.3	up	up	ccc		
at-17/1/1	up	up			

```

ge-17/2/0          up    up
ge-17/2/1          up    up
so-17/3/0          up    down
so-19/0/0          up    down
so-19/1/0          up    down
so-19/2/0          up    down
so-19/3/0          up    down
bcm0               up    up
bcm0.0             up    up    tnp    4
dsc                up    up
em0                up    up
em0.0              up    up    tnp    4
fxp0               up    up
fxp0.0             up    up    inet   192.168.77.158/21
gre                up    up
ipip               up    up
lo0                up    up
lo0.0              up    up    inet   10.255.70.158      --> 0/0
                                   127.0.0.1        --> 0/0
                                   iso
47.0005.80ff.f800.0000.0108.0001.0102.5507.0158.00
                                   inet6 2001:db8::10:255:70:158
                                   fe80::280:42ff:fe13:269d
lo0.16385          up    up    inet
                                   inet6 fe80::280:42ff:fe13:269d
lsi                up    up
mtun               up    up
pimd               up    up
pime               up    up
tap                up    up

```

Displaying Available Routes

When you need to verify route information for a routing matrix, you must issue operational commands on the TX Matrix Plus router. To display available routes for the routing matrix, issue a `show route` command:

```

user@host> show route summary
Router ID: 10.255.77.158
inet.0: 13 destinations, 14 routes (12 active, 0 holddown, 1 hidden)
    Direct:    4 routes,      3 active
    Local:     2 routes,      2 active
    Static:    6 routes,      6 active
    IS-IS:     2 routes,      1 active
inet.3: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
    LDP:       1 routes,      1 active
iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
    Direct:    1 routes,      1 active
mpls.0: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
    MPLS:      3 routes,      3 active
    LDP:       2 routes,      2 active
    L2CKT:     2 routes,      2 active
inet6.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
    Direct:    2 routes,      2 active
__juniper_private1__inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0
hidden)
    Direct:    1 routes,      1 active
l2circuit.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)

```

```

LDP:      1 routes,      1 active
L2CKT:    4 routes,      4 active

```

Displaying Alarms and System Uptime

To display alarms for all routing matrix components, issue the `show chassis alarms` command at the TX Matrix Plus router:

```

user@host> show chassis alarms
sfc0-re0:
-----
2 alarms currently active
Alarm time           Class  Description
2009-07-10 02:27:46 PDT  Minor  LCC 0 Minor Errors
2009-07-09 17:12:29 PDT  Major  LCC 2 Major Errors

```

```

1cc0-re0:
-----
1 alarms currently active
Alarm time           Class  Description
2009-07-10 02:27:46 PDT  Minor  PEM 1 Absent

```

```

1cc2-re0:
-----
1 alarms currently active
Alarm time           Class  Description
2009-07-09 17:12:29 PDT  Major  PEM 1 Not OK

```

Displaying Craft Interface Messages

To display the messages that are currently displayed on the craft interface for all routing matrix components, issue the `show chassis craft-interface` command at the master Routing Engine of the TX Matrix Plus router:

```

user@host> show chassis craft-interface
sfc0-re0:

```

```

-----
FPM Display Contents:
+-----+
|mvsfc0      |
|2 Alarms active|
|R: LCC 0 Minor Error|
|R: LCC 2 Major Error|
+-----+

```

```

Front Panel System LEDs:
Routing Engine    0    1
-----
OK                *    *
Fail              .    .
Master            *    .

```

```

Front Panel Alarm Indicators:
-----
Red LED          *
Yellow LED       *
Major relay      *

```

Minor relay *

Front Panel F13 SIB LEDs:

SIB	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Fail	*	*
OK	*	*	.	*	*	.	.	.	*	*	.	*	*	.	.	.
Active	*	*	.	*	*	.	.	.	*	*	.	*	*	.	.	.

PS LEDs:

PS	0	1
Red	.	.
Green	.	*

Fan Tray LEDs:

FT	0	1	2	3	4	5
Red
Green	*	*	*	*	*	*

CB LEDs:

CB	0	1
Amber	.	.
Green	*	*
Blue	*	.

1cc0-re0:

FPM Display contents:

mylcc0	
1 Alarms active	
R: PEM 1 Absent	

Front Panel System LEDs:

Routing Engine	0	1
OK	.	*
Fail	.	.
Master	*	.

Front Panel Alarm Indicators:

Red LED	*
Yellow LED	*
Major relay	*
Minor relay	*

Front Panel FPC LEDs:

FPC	0	1	2	3	4	5	6	7
Red
Green	*

CB LEDs:

CB	0	1
Amber	.	.

Green * *
Blue * .

SCG LEDs:
SCG 0 1

Amber . .
Green * *
Blue * .

SIB LEDs:
SIB 0 1 2 3 4

Red . . * . .
Green . * . * *

lcc2-re0:

FPM Display contents:

```
+-----+  
|mylcc2      |  
|1 Alarms active|  
|R: PEM 1 Not OK|  
+-----+
```

Front Panel System LEDs:
Routing Engine 0 1

OK * *
Fail . .
Master * .

Front Panel Alarm Indicators:

Red LED *
Yellow LED *
Major relay *
Minor relay *

Front Panel FPC LEDs:
FPC 0 1 2 3 4 5 6 7

Red
Green * * *

CB LEDs:
CB 0 1

Amber . .
Green * *
Blue * .

SCG LEDs:
SCG 0 1

Amber . .
Green * .
Blue * .

SIB LEDs:
SIB 0 1 2 3 4

```

-----
Red      .      .      .      .      .
Green    *      *      *      *      *

```

Displaying System Uptime

To display the amount of time the routing matrix components have been in operation, issue the `show system uptime` command on the TX Matrix Plus router:

```

user@host> show system uptime
sfc0-re0:
-----
Current time: 2009-07-10 07:55:56 PDT
System booted: 2009-07-09 17:08:41 PDT (14:47:15 ago)
Protocols started: 2009-07-09 17:09:22 PDT (14:46:34 ago)
Last configured: 2009-07-09 17:08:28 PDT (14:47:28 ago) by root
  7:55AM up 14:47, 1 user, load averages: 0.00, 0.00, 0.00

lcc0-re0:
-----
Current time: 2009-07-10 07:55:56 PDT
System booted: 2009-07-09 17:07:40 PDT (14:48:16 ago)
Last configured: 2009-07-09 17:08:43 PDT (14:47:13 ago) by root
  7:55AM up 14:48, 0 users, load averages: 0.07, 0.02, 0.01

lcc2-re0:
-----
Current time: 2009-07-10 07:55:56 PDT
System booted: 2009-07-09 17:07:33 PDT (14:48:23 ago)
Last configured: 2009-07-09 17:08:47 PDT (14:47:09 ago) by root
  7:55AM up 14:48, 0 users, load averages: 0.00, 0.00, 0.00

```

Displaying Chassis Hardware and Status

To display the hardware inventory for a routing matrix with a TX Matrix Plus router, you can select output for the TX Matrix Plus router only, a specific T1600 router, or all components. If a specific component (`sfc` or `lcc`) is not specified as an option in the command, the default output displays information for the entire routing matrix.

```

user@host> show chassis hardware ?
Possible completions:
  <[Enter]>      Execute this command
  clei-models    Display CLEI barcode and model number for orderable FRUs
  detail         Include RAM and disk information in output
  extensive      Display ID EEPROM information
  lcc            Display chassis-specific information (0..3)
  models        Display serial number and model number for orderable FRUs

  sfc           Display chassis-specific information (0..0)
  |             Pipe through a command

```

You can display information about individual hardware components in the TX Matrix Plus router, in a specific T1600 router, or the entire routing matrix:

- Displaying Information About All Hardware Components on page 50
- Displaying Information About SIBs on page 53

- Displaying Information About Routing Engines on page 54
- Displaying Information About FPCs on page 56
- Displaying Information About LCCs on page 57

Displaying Information About All Hardware Components

To display all hardware components in a routing matrix, issue the `show chassis hardware` command on the TX Matrix Plus router:

```
user@host> show chassis hardware
sfc0-re0:
```

```
-----
Hardware inventory:
```

Item	Version	Part number	Serial number	Description
Chassis			JN1131103AHB	TXP
Midplane	REV 05	710-022574	TS4035	SFC Midplane
FPM Display	REV 01	710-024027	DR4982	TXP FPM Display
CIP 0	REV 02	710-023792	DS4568	TXP CIP
CIP 1	REV 02	710-023792	DS4562	TXP CIP
PEM 1	VER 01	740-027463	123456	Power Entry Module
Routing Engine 0	REV 01	740-026942	737A-1024	SFC RE
Routing Engine 1	REV 01	740-026942	737A-1008	SFC RE
CB 0	REV 01	710-022606	DP8889	SFC Control Board
CB 1	REV 05	710-022606	DW1103	SFC Control Board
SPMB 0		BUILTIN		SFC Switch CPU
SPMB 1		BUILTIN		SFC Switch CPU
SIB F13 0	REV 03	750-024564	DT9486	F13 SIB
B Board	REV 02	710-023431	DT6550	F13 SIB Mezz
SIB F13 1	REV 03	750-024564	DT9456	F13 SIB
B Board	REV 02	710-023431	DT6562	F13 SIB
SIB F13 3	REV 04	710-022600	DX0900	F13 SIB
B Board	REV 03	710-023431	DX0957	F13 SIB Mezz
SIB F13 4	REV 04	750-024564	DW5753	F13 SIB
B Board	REV 03	710-023431	DW9034	F13 SIB Mezz
SIB F13 6	REV 03	750-024564	DT9483	F13 SIB
B Board	REV 02	710-023431	DT6558	F13 SIB Mezz
SIB F13 7	REV 04	750-024564	DW5790	F13 SIB
B Board	REV 03	710-023431	DW9072	F13 SIB Mezz
SIB F13 8	REV 04	710-022600	DX0833	F13 SIB
B Board	REV 03	710-023431	DX0938	F13 SIB Mezz
SIB F13 9	REV 03	750-024564	DT9465	F13 SIB
B Board	REV 02	710-023431	DT6574	F13 SIB Mezz
SIB F13 11	REV 04	750-024564	DW5756	F13 SIB
B Board	REV 03	710-023431	DW9072	F13 SIB Mezz
SIB F13 12	REV 04	750-024564	DW5749	F13 SIB
B Board	REV 03	710-023431	DW9050	F13 SIB Mezz
SIB F2S 0/0	REV 03	710-022603	DV0063	F2S SIB
B Board	REV 03	710-023787	DT9917	F2S SIB Mezz
SIB F2S 0/2	REV 03	710-022603	DV0090	F2S SIB
B Board	REV 03	710-023787	DT9994	F2S SIB Mezz
SIB F2S 0/4	REV 03	710-022603	DV0076	F2S SIB
B Board	REV 03	710-023787	DT9930	F2S SIB Mezz
SIB F2S 0/6	REV 03	710-022603	DV0035	F2S SIB
B Board	REV 03	710-023787	DT9951	F2S SIB Mezz
SIB F2S 1/0	REV 03	710-022603	DV0067	F2S SIB
B Board	REV 03	710-023787	DT9922	F2S SIB Mezz
SIB F2S 1/2	REV 03	710-022603	DV0036	F2S SIB
B Board	REV 03	710-023787	DT9950	F2S SIB Mezz

SIB F2S 1/4	REV 03	710-022603	DV0085	F2S SIB
B Board	REV 03	710-023787	DT9935	F2S SIB Mezz
SIB F2S 1/6	REV 03	710-022603	DV0028	F2S SIB
B Board	REV 03	710-023787	DT9960	F2S SIB Mezz
SIB F2S 2/0	REV 03	710-022603	DV0047	F2S SIB
B Board	REV 03	710-023787	DT9965	F2S SIB Mezz
SIB F2S 2/2	REV 03	710-022603	DV0023	F2S SIB
B Board	REV 03	710-023787	DT9927	F2S SIB Mezz
SIB F2S 2/4	REV 03	710-022603	DV0046	F2S SIB
B Board	REV 03	710-023787	DT9946	F2S SIB Mezz
SIB F2S 2/6	REV 03	710-022603	DV0025	F2S SIB
B Board	REV 03	710-023787	DT9914	F2S SIB Mezz
SIB F2S 3/0	REV 03	710-022603	DV0110	F2S SIB
B Board	REV 03	710-023787	DT9944	F2S SIB Mezz
SIB F2S 3/2	REV 03	710-022603	DV0056	F2S SIB
B Board	REV 03	710-023787	DT9979	F2S SIB Mezz
SIB F2S 3/4	REV 03	710-022603	DV0024	F2S SIB
B Board	REV 03	710-023787	DT9920	F2S SIB Mezz
SIB F2S 3/6	REV 03	710-022603	DV0108	F2S SIB
B Board	REV 03	710-023787	DT9989	F2S SIB Mezz
SIB F2S 4/0	REV 02	710-022603	DT2821	F2S SIB
B Board	REV 02	710-023787	DT1719	F2S SIB Mezz
SIB F2S 4/2	REV 03	710-022603	DV0031	F2S SIB
B Board	REV 03	710-023787	DT9953	F2S SIB Mezz
SIB F2S 4/4	REV 03	710-022603	DV0094	F2S SIB
B Board	REV 03	710-023787	DT9918	F2S SIB Mezz
SIB F2S 4/6	REV 03	710-022603	DV0022	F2S SIB
B Board	REV 03	710-023787	DT9943	F2S SIB Mezz
Fan Tray 0	REV 02	710-024029	DP5653	Front Fan Tray
Fan Tray 1	REV 02	760-024497	DP5661	Front Fan Tray
Fan Tray 2	REV 02	760-024502	DP5660	Rear Fan Tray
Fan Tray 3	REV 02	760-024502	DR8267	Rear Fan Tray
Fan Tray 4	REV 02	760-024502	DR8265	Rear Fan Tray
Fan Tray 5	REV 02	760-024502	DR8272	Rear Fan Tray

lcc0-re0:-----
Hardware inventory:

Item	Version	Part number	Serial number	Description
Chassis			JN1090A04AHA	T1600
Midplane	REV 01	710-017247	RB8601	T-series Backplane
FPM GBUS	REV 09	710-002901	JW7364	T640 FPM Board
FPM Display	REV 05	710-002897	JY6782	FPM Display
CIP	REV 06	710-002895	JW7565	T-series CIP
PEM 0	Rev 14	740-002595	SL26454	Power Entry Module
PEM 1	Rev 15	740-002595	SM30230	Power Entry Module
SCG 0	REV 03	710-003423	HA4517	T640 Sonet Clock Gen.
SCG 1	REV 04	710-003423	HF6049	T640 Sonet Clock Gen.
Routing Engine 0	REV 00	740-026941	737F-1051	LCC RE
Routing Engine 1	REV 01	740-026941	737F-1095	LCC RE
CB 0	REV 05	710-022597	DV4262	LCC Control Board
CB 1	REV 06	710-022597	DX4009	LCC Control Board
FPC 7	REV 01	710-010845	JB7158	FPC Type 4
CPU	REV 02	710-011481	JB6048	FPC CPU-Enhanced
MMB 0	REV 06	710-010842	JP1703	ST-MMB
SPMB 0	REV 04	710-023321	DV3863	LCC Switch CPU
SPMB 1	REV 04	710-023321	DW3634	LCC Switch CPU
SIB 1	REV 07	710-022594	DW4208	LCC SIB
B Board	REV 07	710-023185	DW3944	LCC SIB Mezz
SIB 2	REV 07	710-022594	DW4205	LCC SIB
B Board	REV 07	710-023185	DW3945	LCC SIB Mezz

SIB 3	REV 07	710-022594	DW4218	LCC SIB
B Board	REV 07	710-023185	DW3931	LCC SIB Mezz
SIB 4	REV 07	710-022594	DW4183	LCC SIB
B Board	REV 07	710-023185	DW3936	LCC SIB Mezz
Fan Tray 0				Front Top Fan Tray
Fan Tray 1				Front Bottom Fan Tray
Fan Tray 2				Rear Fan Tray

lcc2-re0:-----
Hardware inventory:

Item	Version	Part number	Serial number	Description
Chassis			JN1102900AHA	T1600
Midplane	REV 03	710-017247	RC3766	T-series Backplane
FPM GBUS	REV 10	710-002901	DR1415	T640 FPM Board
FPM Display	REV 01	710-021387	DN7019	T1600 FPM Display
CIP	REV 06	710-002895	DP6012	T-series CIP
PEM 0	Rev 07	740-017906	UC26613	Power Entry Module 3x80
PEM 1	Rev 07	740-017906	UC26544	Power Entry Module 3x80
SCG 0	REV 15	710-003423	DR0914	T640 Sonet Clock Gen.
Routing Engine 0	REV 01	740-026941	737F-1032	LCC RE
Routing Engine 1	REV 01	740-026942	737F-1024	LCC RE
CB 0	REV 05	710-022597	DV4260	LCC Control Board
CB 1	REV 01	710-022597	DM1159	LCC Control Board
FPC 0	REV 12	710-013037	DR1172	FPC Type 4-ES
CPU	REV 08	710-016744	DR0997	ST-PMB2
PIC 0	REV 11	750-017405	DP8833	4x 10GE (LAN/WAN) XFP
Xcvr 0		NON-JNPR	344512071800039	XFP-10G-SR
MMB 0	REV 04	710-016036	DR0651	ST-MMB2
MMB 1	REV 04	710-016036	DR0803	ST-MMB2
FPC 1	REV 07	710-013560	DP9980	E2-FPC Type 3
CPU	REV 05	710-013563	DR0088	FPC CPU-Enhanced
PIC 0	REV 22	750-007141	DR1955	10x 1GE(LAN), 1000 BASE
PIC 1	REV 22	750-007141	DR1944	10x 1GE(LAN), 1000 BASE
PIC 2	REV 22	750-007141	DR1963	10x 1GE(LAN), 1000 BASE
PIC 3	REV 07	750-015217	DN4835	8x 1GE(TYPE3), IQ2
Xcvr 0	REV 01	740-011613	P9F1AQL	SFP-SX
MMB 0	REV 07	710-010171	DP1350	MMB-5M3-288mbit
MMB 1	REV 07	710-010171	DP1386	MMB-5M3-288mbit
FPC 2	REV 14	710-010845	DN1255	FPC Type 4
CPU	REV 06	710-011481	DM9414	FPC CPU-Enhanced
MMB 0	REV 01	710-016606	DP1869	ST-MMB
SPMB 0	REV 04	710-023321	DV3859	LCC Switch CPU
SPMB 1	REV 01	710-023321	DM1161	LCC Switch CPU
SIB 0	REV 07	710-022594	DW4197	LCC SIB
B Board	REV 07	710-023185	DW3916	LCC SIB Mezz
SIB 1	REV 06	710-022594	DT8266	LCC SIB
B Board	REV 06	710-023185	DT5793	LCC SIB Mezz
SIB 2	REV 07	710-022594	DW4184	LCC SIB
B Board	REV 07	710-023185	DW3915	LCC SIB Mezz
SIB 3	REV 06	710-022594	DT8271	LCC SIB
B Board	REV 04	710-023185	DS2323	LCC SIB Mezz
SIB 4	REV 06	710-022594	DT8238	LCC SIB
B Board	REV 06	710-023185	DT5783	LCC SIB Mezz
Fan Tray 0				Front Top Fan Tray
Fan Tray 1				Front Bottom Fan Tray

Fan Tray 2

Rear Fan Tray -- Rev 2

Displaying Information About SIBs

To display all the SIBs in the entire routing matrix, issue the `show chassis sibs` command on the TX Matrix Plus router.

```
user@host> show chassis sibs
sfc0-re0:
```

Slot	State	Type	Uptime
0	Online	SIB F13	1 hour, 27 minutes, 13 seconds
1	Online	SIB F13	1 hour, 26 minutes, 56 seconds
2	Invalid		
3	Online	SIB F13	1 hour, 28 minutes, 23 seconds
4	Online	SIB F13	1 hour, 28 minutes, 20 seconds
5	Invalid		
6	Fault	SIB F13	
7	Online	SIB F13	1 hour, 27 minutes, 56 seconds
8	Online	SIB F13	1 hour, 27 minutes, 48 seconds
9	Online	SIB F13	1 hour, 27 minutes, 31 seconds
10	Invalid		
11	Online	SIB F13	1 hour, 27 minutes, 24 seconds
12	Online	SIB F13	1 hour, 27 minutes, 21 seconds
13	Invalid		
14	Invalid		
15	Invalid		
0/0	Online	SIB F2S	1 hour, 27 minutes, 20 seconds
0/2	Online	SIB F2S	1 hour, 27 minutes, 19 seconds
0/4	Online	SIB F2S	1 hour, 27 minutes, 17 seconds
0/6	Online	SIB F2S	1 hour, 27 minutes, 16 seconds
1/0	Online	SIB F2S	1 hour, 28 minutes, 30 seconds
1/2	Online	SIB F2S	1 hour, 28 minutes, 28 seconds
1/4	Online	SIB F2S	1 hour, 28 minutes, 27 seconds
1/6	Online	SIB F2S	1 hour, 28 minutes, 26 seconds
2/0	Online	SIB F2S	1 hour, 28 minutes, 19 seconds
2/2	Online	SIB F2S	1 hour, 28 minutes, 18 seconds
2/4	Online	SIB F2S	1 hour, 28 minutes, 17 seconds
2/6	Online	SIB F2S	1 hour, 28 minutes, 16 seconds
3/0	Online	SIB F2S	1 hour, 27 minutes, 55 seconds
3/2	Online	SIB F2S	1 hour, 27 minutes, 53 seconds
3/4	Online	SIB F2S	1 hour, 27 minutes, 52 seconds
3/6	Online	SIB F2S	1 hour, 27 minutes, 51 seconds
4/0	Online	SIB F2S	1 hour, 27 minutes, 31 seconds
4/2	Online	SIB F2S	1 hour, 27 minutes, 29 seconds
4/4	Online	SIB F2S	1 hour, 27 minutes, 28 seconds
4/6	Online	SIB F2S	1 hour, 27 minutes, 27 seconds

```
1cc0-re0:
```

Slot	State	Uptime
0	Empty	
1	Online	1 hour, 1 minute, 55 seconds
2	Fault	
3	Online	1 hour, 1 minute, 51 seconds
4	Online	1 hour, 1 minute, 49 seconds

```
1cc2-re0:
```

```

-----
Slot  State                Uptime
0      Online              1 hour, 1 minute, 2 seconds
1      Online              1 hour, 1 minute, 17 seconds
2      Check               1 hour, 1 minute, 15 seconds
3      Online              1 hour, 1 minute, 6 seconds
4      Online              1 hour, 1 minute, 4 seconds

```

Displaying Information About Routing Engines

To display information about all master Routing Engines in the routing matrix, issue the `show chassis routing-engine` command on the TX Matrix Plus router:

```

user@host> show chassis routing-engine
sfc0-re0:

```

Routing Engine status:

Slot 0:

```

Current state                Master
Election priority            Master (default)
Temperature                  28 degrees C / 82 degrees F
CPU temperature              42 degrees C / 107 degrees F
DRAM                        3327 MB
Memory utilization           13 percent
CPU utilization:
  User                       0 percent
  Background                 0 percent
  Kernel                     2 percent
  Interrupt                  0 percent
  Idle                       98 percent
Model                        RE-TXP-SFC
Serial ID                    737A-1024
Start time                   2009-07-09 17:08:13 PDT
Uptime                       14 hours, 55 minutes, 39 seconds
Last reboot reason           Router rebooted after a normal shutdown.
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                  29 degrees C / 84 degrees F
CPU temperature              45 degrees C / 113 degrees F
DRAM                        3327 MB
Memory utilization           11 percent
CPU utilization:
  User                       0 percent
  Background                 0 percent
  Kernel                     0 percent
  Interrupt                  0 percent
  Idle                       100 percent
Model                        RE-TXP-SFC
Serial ID                    737A-1008
Start time                   2009-07-08 12:38:29 PDT
Uptime                       1 day, 19 hours, 25 minutes, 35 seconds
Last reboot reason           Router rebooted after a normal shutdown.

```

```

1cc0-re0:

```

Routing Engine status:**Slot 0:**

Current state	Master
Election priority	Master (default)
Temperature	33 degrees C / 91 degrees F
CPU temperature	47 degrees C / 116 degrees F
DRAM	3327 MB
Memory utilization	12 percent
CPU utilization:	
User	0 percent
Background	0 percent
Kernel	2 percent
Interrupt	0 percent
Idle	98 percent
Model	RE-TXP-LCC
Serial ID	737F-1051
Start time	2009-07-09 17:06:52 PDT
Uptime	14 hours, 56 minutes, 37 seconds
Last reboot reason	Router rebooted after a normal shutdown.
Load averages:	1 minute 5 minute 15 minute
	0.00 0.02 0.00

Routing Engine status:**Slot 1:**

Current state	Backup
Election priority	Backup (default)
Temperature	33 degrees C / 91 degrees F
CPU temperature	47 degrees C / 116 degrees F
DRAM	3327 MB
Memory utilization	9 percent
CPU utilization:	
User	0 percent
Background	0 percent
Kernel	0 percent
Interrupt	0 percent
Idle	100 percent
Model	RE-TXP-LCC
Serial ID	737F-1095
Start time	2009-07-08 12:40:14 PDT
Uptime	1 day, 19 hours, 24 minutes, 1 second
Last reboot reason	Router rebooted after a normal shutdown.

1cc2-re0:-----
Routing Engine status:**Slot 0:**

Current state	Master
Election priority	Master (default)
Temperature	30 degrees C / 86 degrees F
CPU temperature	45 degrees C / 113 degrees F
DRAM	3327 MB
Memory utilization	12 percent
CPU utilization:	
User	0 percent
Background	0 percent
Kernel	2 percent
Interrupt	0 percent
Idle	98 percent
Model	RE-TXP-LCC
Serial ID	737F-1032
Start time	2009-07-09 17:06:53 PDT

```

Uptime                14 hours, 56 minutes, 48 seconds
Last reboot reason    Router rebooted after a normal shutdown.
Load averages:        1 minute   5 minute  15 minute
                      0.01       0.01     0.00

```

Routing Engine status:**Slot 1:**

```

Current state          Backup
Election priority      Backup (default)
Temperature            30 degrees C / 86 degrees F
CPU temperature        44 degrees C / 111 degrees F
DRAM                  3327 MB
Memory utilization     9 percent
CPU utilization:
  User                 0 percent
  Background           0 percent
  Kernel               0 percent
  Interrupt            0 percent
  Idle                 100 percent
Model                  RE-TXP-LCC
Serial ID              737F-1024
Start time             2009-07-08 12:37:52 PDT
Uptime                1 day, 19 hours, 26 minutes, 22 seconds
Last reboot reason    Router rebooted after a normal shutdown.

```

Displaying Information About FPCs

To display information about FPCs in a routing matrix, issue the `show chassis fpc` command. Because there are no FPCs in a TX Matrix Plus router, there is no `sfc` option available for this command.

```
user@host> show chassis fpc
```

1cc0-re0:

```

-----
Slot State      Temp  CPU Utilization (%)  Memory  Utilization (%)
              (C)  Total  Interrupt    DRAM (MB)  Heap    Buffer
0 Empty
1 Empty
2 Empty
3 Empty
4 Empty
5 Empty
6 Empty
7 Online        49    4          0      1024      7      49

```

1cc2-re0:

```

-----
Slot State      Temp  CPU Utilization (%)  Memory  Utilization (%)
              (C)  Total  Interrupt    DRAM (MB)  Heap    Buffer
0 Online        47    8          0      2048      6      24
1 Online        28    4          0      1024      4      49
2 Online        39    5          0      1024      7      49
3 Empty
4 Empty
5 Empty
6 Empty
7 Empty

```

Displaying Information About LCCs

You can also check to see if the TX Matrix Plus router and T1600 routers are communicating correctly within the routing matrix. To verify that the T1600 routers have proper connectivity to the routing matrix, issue the **show chassis lccs** command. In this example, there are two T1600 routers in the routing matrix.

```
user@host> show chassis lccs
Slot  State                Uptime
 0    Online                15 hours, 5 minutes, 58 seconds
 1    Empty
 2    Online                15 hours, 5 minutes, 58 seconds
 3    Empty
```

- Related Topics**
- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)
 - [Overview of a Routing Matrix with a TX Matrix Plus Router on page 3](#)
 - [Roadmap for Configuring the Routing Matrix on page 15](#)
 - [Example Configuration for the Routing Matrix on page 33](#)
 - [Upgrading the JUNOS Software on the Routing Matrix on page 61](#)

Part 3

Administration

- Upgrading the JUNOS Software on page 61
- Managing Files and Processes on page 97
- Managing Hardware Components on page 103

Chapter 5

Upgrading the JUNOS Software

- Upgrading the JUNOS Software on a Routing Matrix with a TX Matrix Plus Router on page 61
- Troubleshooting: Software Upgrade Failure Due to Insufficient Free Disk Space on a Routing Matrix with a TX Matrix Plus Router on page 85
- Troubleshooting: Software Upgrade Failure Due to Insufficient Available DRAM on a Routing Matrix with a TX Matrix Plus Router on page 91
- Troubleshooting: Software Upgrade Failure Due to Mixed Software Versions on a Routing Matrix with a TX Matrix Plus Router on page 91

Upgrading the JUNOS Software on a Routing Matrix with a TX Matrix Plus Router

When you upgrade the JUNOS Software on a routing matrix with a TX Matrix Plus router, the new image is loaded onto the TX Matrix Plus router (sfc0) and distributed to all of the connected T1600 routers (lcc0, lcc1, lcc2, and lcc3).



NOTE: Throughout this task description, the terms *TX Matrix Plus router* and *T1600 router* refer to the routing matrix switch-fabric chassis (SFC) and line-card chassis (LCC), respectively.

Before you begin upgrading the JUNOS Software on a routing matrix, be sure you have:

- Sufficient free disk space on each Routing Engine in the routing matrix. To determine the amount of disk space currently available on all Routing Engines of the routing matrix, issue the **show system storage** operational command.
- Sufficient DRAM on each Routing Engine in the routing matrix. To determine the amount of DRAM currently available on all the Routing Engines in the routing matrix, issue the **show chassis routing-engine** operational command.
- Compatible models of Routing Engines (master and backup) in the routing matrix. For this release of JUNOS Software, the TX Matrix Plus router must contain two model RE-DUO-C2600-16G Routing Engines, and each T1600 router in the routing matrix must contain two model RE-DUO-C1800-8G Routing Engines. To determine the Routing Engine models installed in the Routing Engines in a routing matrix,

issue the **show chassis hardware** operational command with the **models** option. Hardware component model numbers are listed in the **FRU model number** column.

- The same version of JUNOS Software running on all Routing Engines in the routing matrix.



NOTE: The routing matrix does not operate properly unless all master Routing Engines are running the same version of JUNOS Software.

Different versions of the JUNOS Software can use different message formats that might be incompatible, particularly when graceful Routing Engine switchover (GRES) is enabled. For more information about GRES, see “Understanding Graceful Routing Engine Switchover in the JUNOS Software” and “Configuring Graceful Routing Engine Switchover” in the *JUNOS High Availability Configuration Guide*.

Because the procedure for upgrading the JUNOS Software includes switching of the Routing Engine mastership, we recommend that the same version of JUNOS Software is running on all Routing Engines in the routing matrix. To determine the versions of JUNOS Software running on the Routing Engines in a routing matrix, issue the **show version invoke-on** operational command.



BEST PRACTICE: By default, the Routing Engine in slot 0 (**re0**) is the master and the one in slot 1 (**re1**) is the backup. You can modify the default Routing Engine mastership by including the **routing-engine slot-number (master | backup | disabled)** statement at the **[edit chassis redundancy]** hierarchy level. However, it is considered best practice to make sure that all master Routing Engines are **re0** and all backup Routing Engines are **re1** (or vice versa). Throughout this task description, the master Routing Engine is **re0** and the backup Routing Engine is **re1** in all routers in the routing matrix.

This topic includes the following tasks:

1. Logging In to the Master Routing Engine on the TX Matrix Plus Router on page 63
2. Disabling Routing Engine Redundancy and GRES on the Routing Matrix on page 63
3. Installing JUNOS Software on the Backup Routing Engines on page 65
4. Loading Software on the Backup Routing Engines on page 73
5. Switching Routing Engine Mastership to the Backup Routing Engines on page 75
6. Installing Software on the New Default Master Routing Engines on page 77
7. Loading Software on the New Default Master Routing Engines on page 78
8. Optional: Switching Routing Engine Mastership Back to the Default Master Routing Engines on page 80
9. Restoring the Original Configuration on page 82
10. Backing Up the Routing Engines on page 83

Logging In to the Master Routing Engine on the TX Matrix Plus Router

Log in to the master Routing Engine on the TX Matrix Plus router.



NOTE: During the upgrade process, in-band management connections to the routing matrix are lost each time you reboot the system (first, to load the new software onto the backup Routing Engines and later, to load the new software onto the master Routing Engines). Therefore, we recommend that you perform the software upgrade procedure out-of-band, over a direct console connection to the TX Matrix Plus router using Secure Shell (SSH) or Telnet. Connect the system console device to the asynchronous serial port (labeled **CONSOLE**) on the Control Board (CB) associated with the master Routing Engine.

To log in to the master Routing Engine on the TX Matrix Plus router and start the JUNOS CLI:

1. At a management console attached to the master Routing Engine (**re0**) on the TX Matrix Plus router, log in to the JUNOS CLI using a root administration account:

```
login: root
Password: .....

— JUNOS 9.6R1.0 built 2009-07-13 08:52:23 UTC
```



NOTE: From the factory, the root administration user account is not associated with a password.

Amnesiac (ttyd0)

login: **root**

However, you must add a password to the root administration account before you can successfully commit a configuration.

2. At the shell prompt (%), start the JUNOS CLI:

```
% cli
{master}
root@mysfc0-re0>
```

Disabling Routing Engine Redundancy and GRES on the Routing Matrix

By default, graceful Routing Engine switchover (GRES) is disabled. To enable GRES on a router, you can include the **graceful-switchover** statement at the [edit chassis redundancy] hierarchy level:

```
[edit]
chassis {
  redundancy { # This enables Routing Engine redundancy on the router
```

```

    graceful-switchover; # This enables GRES on the router
  }
}

```

The procedure for upgrading the JUNOS Software on a routing matrix with a TX Matrix Plus router includes tasks that entail disruption of traffic processing by the affected Routing Engines:

- Upgrading the software on all the backup Routing Engines.
- Switching the Routing Engine mastership to the backup Routing Engines.
- Upgrading the software on all the master Routing Engines.
- Switching the Routing Engine mastership back to the master Routing Engines.

If GRES is enabled on any routers in the routing matrix, disable this feature.

To disable GRES on all routers in the routing matrix:

1. Enter configuration mode:

```

{master}
root@mysfc0-re0> configure
Entering configuration mode

```



NOTE: In the example shown in this step, the form of CLI prompt (beginning with either {master} or {backup} and displaying either -re0 or -re1 in the hostname portion of the prompt) indicates that GRES is enabled on that router.

2. Display the Routing Engine redundancy stanza:

```

{master}
[edit]
root@mysfc0-re0# show chassis
...
    redundancy {
        graceful-switchover;
    }
...

```

3. Disable Routing Engine redundancy in the candidate configuration. If GRES is enabled, it is removed with the **redundancy** stanza:

```

{master}
[edit]
root@mysfc0-re0# delete chassis redundancy

```

4. Verify that the **graceful-routing** statement has been removed (if it was previously enabled) with the **redundancy** stanza from the [edit chassis] hierarchy level of the candidate configuration:

```

[edit]
root@mysfc0# show chassis

```



NOTE: You can also issue the `show system switchover all-chassis operational` command to display Routing Engine graceful switchover information

5. Commit the configuration on all Routing Engines in the routing matrix and, if the configuration contains no errors and the commit succeeds, exit CLI configuration mode:

```
[edit]
root@mysfc0# commit synchronize and-quit
sfc0-re0:
configuration check succeeds
lcc0-re1:
commit complete
lcc0-re0:
commit complete
lcc2-re1:
commit complete
lcc2-re0:
commit complete
sfc0-re1:
commit complete
sfc0-re0:
commit complete

root@mysfc0>
```

Installing JUNOS Software on the Backup Routing Engines

Install the new version of JUNOS Software on the backup Routing Engines (**re1**) of all the routers in the routing matrix while leaving the currently running version of JUNOS Software running on the master Routing Engines (**re0**). This allows the master Routing Engines to continue operations, minimizing the disruption to the routing matrix and your network.

To install the new version of JUNOS Software on all backup Routing Engines in the routing matrix:

1. At a management console attached to the backup Routing Engine (**re1**) on the TX Matrix Plus router, log in to the JUNOS CLI using a root administration account:

```
login: root
Password: ••••••••
```

```
— JUNOS 9.6R1.0 built 2009-07-13 08:52:23 UTC
```

2. At the shell prompt (%), start the JUNOS CLI:

```
% cli
root@mysfc0_alt_re>
```

3. To install the new JUNOS Software on the backup Routing Engines (**re1**) in the routing matrix, issue the `request system software add` operational command at the TX Matrix Plus router, and specify the new JUNOS Software image file.

The following example shows sample output for the `request system software add` command:

```

root@mysfc0_alt_re> request system software add
/var/tmp/jinstall-9.6R1.2-domestic-signed.tgz

Pushing bundle to lcc0-re1
Pushing bundle to lcc2-re1

Validating on lcc0-re1
Checking compatibility with configuration
Initializing...
Using jbase-9.6R1.0
Verified manifest signed by PackageProduction_9_6_0
Using /var/tmp/jinstall-9.6R1.2-domestic-signed.tgz
Verified jinstall-9.6R1.2-domestic.tgz signed by PackageProduction_9_6_0
Using jinstall-9.6R1.2-domestic.tgz
Using jbundle-9.6R1.2-domestic.tgz
Checking jbundle requirements on /
Using jbase-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using /var/validate/chroot/tmp/jbundle/jboot-9.6R1.2.tgz
Using jkernel-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jcrypto-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jpfe-9.6R1.2.tgz
Using jdocs-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jroute-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jservices-9.6R1.2.tgz
Auto-deleting old jservices-voice ...
Removing /opt/sdk/jservices-voice ...
Removing jservices-voice-bsg-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-voice ...
Verified jservices-voice-bsg-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /var/sw/pkg ...
Creating /opt/sdk/jservices-voice ...
Storing jservices-voice-bsg-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-voice/jservices-voice-bsg ->
/var/sw/pkg/jservices-voice-bsg-9.6R1.2.tgz...
Auto-deleting old jservices-bgf ...
Removing /opt/sdk/jservices-bgf ...
Removing jservices-bgf-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-bgf ...
Verified jservices-bgf-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-bgf ...
Storing jservices-bgf-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-bgf/jservices-bgf-pic ->
/var/sw/pkg/jservices-bgf-pic-9.6R1.2.tgz...
Auto-deleting old jservices-aac1 ...
Removing /opt/sdk/jservices-aac1 ...
Removing jservices-aac1-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-aac1 ...
Verified jservices-aac1-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0

```

```

Creating /opt/sdk/jservices-aac1 ...
Storing jservices-aac1-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-aac1/jservices-aac1-pic ->
/var/sw/pkg/jservices-aac1-pic-9.6R1.2.tgz...
Auto-deleting old jservices-llpdf ...
Removing /opt/sdk/jservices-llpdf ...
Removing jservices-llpdf-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-llpdf ...
Verified jservices-llpdf-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-llpdf ...
Storing jservices-llpdf-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-llpdf/jservices-llpdf-pic ->
/var/sw/pkg/jservices-llpdf-pic-9.6R1.2.tgz...
Auto-deleting old jservices-sfw ...
Removing /opt/sdk/jservices-sfw ...
Removing jservices-sfw-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-sfw ...
Verified jservices-sfw-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-sfw ...
Storing jservices-sfw-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-sfw/jservices-sfw-pic ->
/var/sw/pkg/jservices-sfw-pic-9.6R1.2.tgz...
Auto-deleting old jservices-appid ...
Removing /opt/sdk/jservices-appid ...
Removing jservices-appid-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-appid ...
Verified jservices-appid-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-appid ...
Storing jservices-appid-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-appid/jservices-appid-pic ->
/var/sw/pkg/jservices-appid-pic-9.6R1.2.tgz...
Auto-deleting old jservices-idp ...
Removing /opt/sdk/jservices-idp ...
Removing jservices-idp-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-idp ...
Verified jservices-idp-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-idp ...
Storing jservices-idp-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-idp/jservices-idp-pic ->
/var/sw/pkg/jservices-idp-pic-9.6R1.2.tgz...
Hardware Database regeneration succeeded
Validating against /config/juniper.conf.gz
mgd: commit complete
Validation succeeded

Validating on lcc2-re1
Checking compatibility with configuration
Initializing...
Using jbase-9.6R1.0
Verified manifest signed by PackageProduction_9_6_0
Using /var/tmp/jinstall-9.6R1.2-domestic-signed.tgz
Verified jinstall-9.6R1.2-domestic.tgz signed by PackageProduction_9_6_0
Using jinstall-9.6R1.2-domestic.tgz
Using jbundle-9.6R1.2-domestic.tgz
Checking jbundle requirements on /
Using jbase-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0

```

```

Using /var/validate/chroot/tmp/jbundle/jboot-9.6R1.2.tgz
Using jkernel-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jcrypto-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jpfe-9.6R1.2.tgz
Using jdocs-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jroute-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jservices-9.6R1.2.tgz
Auto-deleting old jservices-voice ...
Removing /opt/sdk/jservices-voice ...
Removing jservices-voice-bsg-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-voice ...
Verified jservices-voice-bsg-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /var/sw/pkg ...
Creating /opt/sdk/jservices-voice ...
Storing jservices-voice-bsg-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-voice/jservices-voice-bsg ->
/var/sw/pkg/jservices-voice-bsg-9.6R1.2.tgz...
Auto-deleting old jservices-bgf ...
Removing /opt/sdk/jservices-bgf ...
Removing jservices-bgf-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-bgf ...
Verified jservices-bgf-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-bgf ...
Storing jservices-bgf-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-bgf/jservices-bgf-pic ->
/var/sw/pkg/jservices-bgf-pic-9.6R1.2.tgz...
Auto-deleting old jservices-aacl ...
Removing /opt/sdk/jservices-aacl ...
Removing jservices-aacl-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-aacl ...
Verified jservices-aacl-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-aacl ...
Storing jservices-aacl-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-aacl/jservices-aacl-pic ->
/var/sw/pkg/jservices-aacl-pic-9.6R1.2.tgz...
Auto-deleting old jservices-llpdf ...
Removing /opt/sdk/jservices-llpdf ...
Removing jservices-llpdf-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-llpdf ...
Verified jservices-llpdf-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-llpdf ...
Storing jservices-llpdf-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-llpdf/jservices-llpdf-pic ->
/var/sw/pkg/jservices-llpdf-pic-9.6R1.2.tgz...
Auto-deleting old jservices-sfw ...
Removing /opt/sdk/jservices-sfw ...
Removing jservices-sfw-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-sfw ...
Verified jservices-sfw-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-sfw ...
Storing jservices-sfw-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-sfw/jservices-sfw-pic ->

```

```

/var/sw/pkg/jservices-sfw-pic-9.6R1.2.tgz...
Auto-deleting old jservices-appid ...
Removing /opt/sdk/jservices-appid ...
Removing jservices-appid-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-appid ...
Verified jservices-appid-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-appid ...
Storing jservices-appid-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-appid/jservices-appid-pic ->
/var/sw/pkg/jservices-appid-pic-9.6R1.2.tgz...
Auto-deleting old jservices-idp ...
Removing /opt/sdk/jservices-idp ...
Removing jservices-idp-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-idp ...
Verified jservices-idp-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-idp ...
Storing jservices-idp-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-idp/jservices-idp-pic ->
/var/sw/pkg/jservices-idp-pic-9.6R1.2.tgz...
Hardware Database regeneration succeeded
Validating against /config/juniper.conf.gz
mgd: commit complete
Validation succeeded

```

```

Validating on sfc0-re1
Checking compatibility with configuration
Initializing...
Using jbase-9.6R1.0
Verified manifest signed by PackageProduction_9_6_0
Using /var/tmp/jinstall-9.6R1.2-domestic-signed.tgz
Verified jinstall-9.6R1.2-domestic.tgz signed by PackageProduction_9_6_0
Using jinstall-9.6R1.2-domestic.tgz
Using jbundle-9.6R1.2-domestic.tgz
Checking jbundle requirements on /
Using jbase-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using /var/validate/chroot/tmp/jbundle/jboot-9.6R1.2.tgz
Using jkernel-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jcrypto-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jpfe-9.6R1.2.tgz
Using jdocs-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jroute-9.6R1.2.tgz
Verified manifest signed by PackageProduction_9_6_0
Using jservices-9.6R1.2.tgz
Auto-deleting old jservices-voice ...
Removing /opt/sdk/jservices-voice ...
Removing jservices-voice-bsg-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-voice ...
Verified jservices-voice-bsg-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /var/sw/pkg ...
Creating /opt/sdk/jservices-voice ...
Storing jservices-voice-bsg-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-voice/jservices-voice-bsg ->
/var/sw/pkg/jservices-voice-bsg-9.6R1.2.tgz...
Auto-deleting old jservices-bgf ...

```

```

Removing /opt/sdk/jservices-bgf ...
Removing jservices-bgf-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-bgf ...
Verified jservices-bgf-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-bgf ...
Storing jservices-bgf-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-bgf/jservices-bgf-pic ->
/var/sw/pkg/jservices-bgf-pic-9.6R1.2.tgz...
Auto-deleting old jservices-aac1 ...
Removing /opt/sdk/jservices-aac1 ...
Removing jservices-aac1-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-aac1 ...
Verified jservices-aac1-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-aac1 ...
Storing jservices-aac1-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-aac1/jservices-aac1-pic ->
/var/sw/pkg/jservices-aac1-pic-9.6R1.2.tgz...
Auto-deleting old jservices-llpdf ...
Removing /opt/sdk/jservices-llpdf ...
Removing jservices-llpdf-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-llpdf ...
Verified jservices-llpdf-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-llpdf ...
Storing jservices-llpdf-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-llpdf/jservices-llpdf-pic ->
/var/sw/pkg/jservices-llpdf-pic-9.6R1.2.tgz...
Auto-deleting old jservices-sfw ...
Removing /opt/sdk/jservices-sfw ...
Removing jservices-sfw-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-sfw ...
Verified jservices-sfw-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-sfw ...
Storing jservices-sfw-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-sfw/jservices-sfw-pic ->
/var/sw/pkg/jservices-sfw-pic-9.6R1.2.tgz...
Auto-deleting old jservices-appid ...
Removing /opt/sdk/jservices-appid ...
Removing jservices-appid-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-appid ...
Verified jservices-appid-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-appid ...
Storing jservices-appid-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-appid/jservices-appid-pic ->
/var/sw/pkg/jservices-appid-pic-9.6R1.2.tgz...
Auto-deleting old jservices-idp ...
Removing /opt/sdk/jservices-idp ...
Removing jservices-idp-pic-9.6R1.0.tgz from /var/sw/pkg ...
Notifying mspd ...
Installing new jservices-idp ...
Verified jservices-idp-pic-9.6R1.2.tgz signed by PackageProduction_9_6_0
Creating /opt/sdk/jservices-idp ...
Storing jservices-idp-pic-9.6R1.2.tgz in /var/sw/pkg ...
Link: /opt/sdk/jservices-idp/jservices-idp-pic ->
/var/sw/pkg/jservices-idp-pic-9.6R1.2.tgz...
Hardware Database regeneration succeeded
Validating against /config/juniper.conf.gz

```

```

mgd: commit complete
Validation succeeded
Done with validate on all chassis

```

1cc0-rel:

```

Installing package '/var/tmp/jinstall-9.6R1.2-domestic-signed.tgz' ...
Verified jinstall-9.6R1.2-domestic.tgz signed by PackageProduction_9_6_0
Adding jinstall...
Verified manifest signed by PackageProduction_9_6_0

```

```

WARNING: This package will load JUNOS 9.6R1.2 software.
WARNING: It will save JUNOS configuration files, and SSH keys
WARNING: (if configured), but erase all other files and information
WARNING: stored on this machine. It will attempt to preserve dumps
WARNING: and log files, but this can not be guaranteed. This is the
WARNING: pre-installation stage and all the software is loaded when
WARNING: you reboot the system.

```

```

Saving the config files ...
NOTICE: uncommitted changes have been saved in
/var/db/config/juniper.conf.pre-install
Installing the bootstrap installer ...

```

```

WARNING: A REBOOT IS REQUIRED TO LOAD THIS SOFTWARE CORRECTLY. Use
the
WARNING: 'request system reboot' command when software installation
is
WARNING: complete. To abort the installation, do not reboot your
system,
WARNING: instead use the 'request system software delete jinstall'
WARNING: command as soon as this operation completes.

```

```

Saving package file in /var/sw/pkg/jinstall-9.6R1.2-domestic-signed.tgz
...
Saving state for rollback ...

```

1cc2-rel:

```

Installing package '/var/tmp/jinstall-9.6R1.2-domestic-signed.tgz' ...
Verified jinstall-9.6R1.2-domestic.tgz signed by PackageProduction_9_6_0
Adding jinstall...
Verified manifest signed by PackageProduction_9_6_0

```

```

WARNING: This package will load JUNOS 9.6R1.2 software.
WARNING: It will save JUNOS configuration files, and SSH keys
WARNING: (if configured), but erase all other files and information
WARNING: stored on this machine. It will attempt to preserve dumps
WARNING: and log files, but this can not be guaranteed. This is the
WARNING: pre-installation stage and all the software is loaded when
WARNING: you reboot the system.

```

```

Saving the config files ...
NOTICE: uncommitted changes have been saved in
/var/db/config/juniper.conf.pre-install
Installing the bootstrap installer ...

```

```

WARNING: A REBOOT IS REQUIRED TO LOAD THIS SOFTWARE CORRECTLY. Use
the
WARNING: 'request system reboot' command when software installation
is
WARNING: complete. To abort the installation, do not reboot your
system,

```

```

WARNING:      instead use the 'request system software delete jinstall'
WARNING:      command as soon as this operation completes.

Saving package file in /var/sw/pkg/jinstall-9.6R1.2-domestic-signed.tgz
...
Saving state for rollback ...

sfc0-rel:
Installing package '/var/tmp/jinstall-9.6R1.2-domestic-signed.tgz' ...
Verified jinstall-9.6R1.2-domestic.tgz signed by PackageProduction_9_6_0
Adding jinstall...
Verified manifest signed by PackageProduction_9_6_0

WARNING:      This package will load JUNOS 9.6R1.2 software.
WARNING:      It will save JUNOS configuration files, and SSH keys
WARNING:      (if configured), but erase all other files and information
WARNING:      stored on this machine. It will attempt to preserve dumps
WARNING:      and log files, but this can not be guaranteed. This is the
WARNING:      pre-installation stage and all the software is loaded when
WARNING:      you reboot the system.

Saving the config files ...
NOTICE: uncommitted changes have been saved in
/var/db/config/juniper.conf.pre-install
Installing the bootstrap installer ...

WARNING:      A REBOOT IS REQUIRED TO LOAD THIS SOFTWARE CORRECTLY. Use
the
WARNING:      'request system reboot' command when software installation
is
WARNING:      complete. To abort the installation, do not reboot your
system,
WARNING:      instead use the 'request system software delete jinstall'
WARNING:      command as soon as this operation completes.

Saving package file in /var/sw/pkg/jinstall-9.6R1.2-domestic-signed.tgz
...
Saving state for rollback ...

root@host>

```

For more information on the **request system software add** command, see the *JUNOS System Basics and Services Command Reference*.

Loading Software on the Backup Routing Engines

To start running the newly installed JUNOS Software on all the backup Routing Engines in the routing matrix, you must reboot those Routing Engines. The Routing Engines reboot from the boot device on which the software was just installed. Each reboot operation can take between 5 and 10 minutes to complete.



NOTE: This is your last chance to abort the upgrade of the JUNOS Software on the backup Routing Engines. If you want to abort the software upgrade, do not reboot the backup Routing Engines.

If you choose to abort the software upgrade, you can remove the new JUNOS Software package or bundle from the router by issuing the `request system software delete` operational command and specifying the `jinstall` command.

To start running the new JUNOS Software on the backup Routing Engines:

1. To reboot all the backup Routing Engines (**re1**), issue the `request system reboot` operational command at the backup Routing Engine one theTX Matrix Plus router. At the warning message and prompt, type **yes** to allow the command to proceed:

```
root@mysfc0_alt_re> request system reboot

Reboot the system ? [yes,no] (no) yes

rebooting lcc0-re1
Rebooting lcc2-re1
Shutdown NOW!
Reboot consistency check bypassed - jinstall 9.6R1.2 will complete
installation upon reboot
[pid 23517]

root@mysfc0_alt_re>

*** FINAL System shutdown message from root@host-sfc0-re1> ***

System going down IMMEDIATELY

rlogin: connection closed

root@host>
```

All of the backup Routing Engines (**re1**) in the routing matrix reboot from the boot devices on which the software was just installed, which loads the new JUNOS Software. Each reboot operation can take between 5 and 10 minutes to complete.

When the backup Routing Engine on the TX Matrix Plus router (**sfc0-re1**) finishes rebooting, the management console attached to the backup Routing Engine on the TX Matrix Plus router displays the login prompt.

2. Log in to the TX Matrix Plus backup Routing Engine (**sfc0-re1**) and issue the **show version** command to verify the version of the software installed:

```
root@mysfc0_alt_re> show version
```

sfc0-re1:

```
-----
Hostname: mysfc0_alt_re
Model: txp
JUNOS Base OS boot [9.6R1.2]
JUNOS Base OS Software Suite [9.6R1.2]
JUNOS Kernel Software Suite [9.6R1.2]
JUNOS Crypto Software Suite [9.6R1.2]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6R1.2]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6R1.2]
JUNOS Online Documentation [9.6R1.2]
JUNOS Voice Services Container package [9.6R1.2]
JUNOS Border Gateway Function package [9.6R1.2]
JUNOS Services AACL Container package [9.6R1.2]
JUNOS Services LL-PDF Container package [9.6R1.2]
JUNOS Services Stateful Firewall [9.6R1.2]
JUNOS AppId Services [9.6R1.2]
JUNOS IDP Services [9.6R1.2]
JUNOS Routing Software Suite [9.6R1.2]
```

lcc0-re1:

```
-----
Hostname: mylcc0_alt_re
Model: t1600
JUNOS Base OS boot [9.6R1.2]
JUNOS Base OS Software Suite [9.6R1.2]
JUNOS Kernel Software Suite [9.6R1.2]
JUNOS Crypto Software Suite [9.6R1.2]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6R1.2]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6R1.2]
JUNOS Online Documentation [9.6R1.2]
JUNOS Voice Services Container package [9.6R1.2]
JUNOS Border Gateway Function package [9.6R1.2]
JUNOS Services AACL Container package [9.6R1.2]
JUNOS Services LL-PDF Container package [9.6R1.2]
JUNOS Services Stateful Firewall [9.6R1.2]
JUNOS AppId Services [9.6R1.2]
JUNOS IDP Services [9.6R1.2]
JUNOS Routing Software Suite [9.6R1.2]
```

lcc2-re1:

```
-----
Hostname: mylcc2_alt_re
Model: t1600
JUNOS Base OS boot [9.6R1.2]
JUNOS Base OS Software Suite [9.6R1.2]
JUNOS Kernel Software Suite [9.6R1.2]
JUNOS Crypto Software Suite [9.6R1.2]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6R1.2]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6R1.2]
JUNOS Online Documentation [9.6R1.2]
JUNOS Voice Services Container package [9.6R1.2]
JUNOS Border Gateway Function package [9.6R1.2]
JUNOS Services AACL Container package [9.6R1.2]
```

```
JUNOS Services LL-PDF Container package [9.6R1.2]
JUNOS Services Stateful Firewall [9.6R1.2]
JUNOS AppId Services [9.6R1.2]
JUNOS IDP Services [9.6R1.2]
JUNOS Routing Software Suite [9.6R1.2]
```

Switching Routing Engine Mastership to the Backup Routing Engines

The new JUNOS Software is now running on all the backup Routing Engines (**re1**) in the routing matrix. Next, you must install and load the same new software on the master Routing Engines (**re0**) of the routing matrix.

While the JUNOS Software is being upgraded (installed and then loaded), the Routing Engine on which you are performing the software upgrade does not route traffic. Therefore, to minimize disruption to network operation, you need to temporarily switch Routing Engine mastership over to the backup Routing Engines (**re1**), which are now running the new version of JUNOS Software.

To switch Routing Engine mastership over to the backup Routing Engines:

1. At a management console attached to the master Routing Engine (**re0**) on the TX Matrix Plus router, log in to the JUNOS CLI using a root administration account:

```
login: root
Password: ••••••••
```

```
— JUNOS 9.6R1.0 built 2009-07-13 08:52:23 UTC
```

2. At the shell prompt (%), start the JUNOS CLI:

```
% cli
root@mysfc0>
```

3. To transfer Routing Engine mastership to the backup Routing Engine (**re1**) for all routers in the routing matrix, issue the **request chassis routing-engine master** command with the **switch** and **all-chassis** options.



CAUTION: Switching Routing Engine mastership causes traffic to be disrupted.

At the warning message and prompt, type **yes** to allow the command to proceed:

```
root@mysfc0> request chassis routing-engine master switch all-chassis
```

```
warning: Traffic will be interrupted while the PFE is re-initialized
Toggle mastership between routing engines ? [yes,no] (no) yes
```

```
1cc0-re0:
```

```
-----
warning: Traffic will be interrupted while the PFE is re-initialized
Resolving mastership...
Complete. The other routing engine becomes the master.
```

1cc2-re0:

```
-----
warning: Traffic will be interrupted while the PFE is re-initialized
Resolving mastership...
Complete. The other routing engine becomes the master.
```

sfc0-re0:

```
-----
warning: Traffic will be interrupted while the PFE is re-initialized
Resolving mastership...
Complete. The other routing engine becomes the master.
```

4. To verify that the *default* backup Routing Engines (installed in slot 1) are now operating the *elected* master Routing Engines, issue the **show chassis routing-engine** command at the TX Matrix Plus router:

```
root@mysfc0> show chassis routing-engine
sfc0-re0:
```

Routing Engine status:

Slot 0:

Current state	Backup
Election priority	Master (default)
...output_truncated...	

Routing Engine status:

Slot 1:

Current state	Master
Election priority	Backup (default)
...output_truncated...	

1cc0-re0:

Routing Engine status:

Slot 0:

Current state	Backup
Election priority	Master (default)
...output_truncated...	

0.02 0.07 0.11

Routing Engine status:

Slot 1:

Current state	Master
Election priority	Backup (default)
...output_truncated...	

1cc2-re0:

Routing Engine status:

Slot 0:

Current state	Backup
Election priority	Master (default)
...output_truncated...	

Routing Engine status:

Slot 1:

Current state	Master
---------------	--------

```
Election priority          Backup (default)
...output_truncated...
```

The default backup Routing Engines (**re1**), which are now running the new version of JUNOS Software, are temporarily operating as the elected master Routing Engines. This enables you to avoid disruption to network operation while you install and load the new version of JUNOS Software on the default master Routing Engines (**re0**) .

Installing Software on the New Default Master Routing Engines

To install the new version of JUNOS Software on the new default master Routing Engines (**re0**) in the routing matrix, issue the **request system software add** operational command at the default master Routing Engine (**re0** in this procedure) of the TX Matrix Plus router, and specify the new JUNOS Software image file.



NOTE: The default backup Routing Engines (**re1**), which are now running the new version of JUNOS Software, are operating as the elected master Routing Engines. This enables you to avoid disruption to network operation while you install and load the new version of JUNOS Software on the default master Routing Engines (**re0**).

The following example of the **request system software add** command is truncated, but the command output is similar to the output displayed when you installed the new software on the default backup Routing Engines:

```
root@mysfc0> request system software add
/var/tmp/jinstall-9.6R1.1-domestic-signed.tgz

Pushing bundle to lcc0-re0
Pushing bundle to lcc2-re0
...output_truncated...
```

Loading Software on the New Default Master Routing Engines

To start running the newly installed JUNOS Software on all the default master Routing Engines in the routing matrix, you must reboot those Routing Engines. The Routing Engines reboot from the boot device on which the software was just installed. Each reboot operation can take between 5 and 10 minutes to complete.



NOTE: This is your last chance to abort the upgrade of the JUNOS Software on the default master Routing Engines. If you want to abort the software upgrade, do not reboot the default master Routing Engines. However, aborting the software upgrade procedure at this point this is not recommend, because the default backup routing Engines are already running the new JUNOS Software.

If you choose to abort the software upgrade, you can remove the new JUNOS Software package or bundle from the router by issuing the **request system software delete** operational command and specifying the **jinstall** command.

To start running the new JUNOS Software on the default master Routing Engines:

1. To reboot all the default master Routing Engines (re0), issue the **request system reboot** operational command at the TX Matrix Plus router.

At the warning message and prompt, type **yes** to allow the command to proceed:

```

root@mysfc0> request system reboot
Reboot the system ? [yes,no] (no) yes

Rebooting lcc0-re1
Rebooting lcc2-re1
Shutdown NOW!
Reboot consistency check bypassed - jinstall 9.6R1.2 will complete
installation upon reboot
[pid 23517]

*** FINAL System shutdown message from root@mysfc0> ***
System going down IMMEDIATELY

root@mysfc0> Jul 23 14:00:52 Waiting (max 60 seconds) for system process
`vn1ru' to stop...done
Waiting (max 60 seconds) for system process `bufdaemon' to stop...done
Waiting (max 60 seconds) for system process `syncer' to stop...
Syncing disks, vnodes remaining...1 1 1 0 1 1 1 1 0 0 0 0 0 0 0 0 0
done

syncing disks... All buffers synced.
Uptime: 26m26s
recorded reboot as normal shutdown
Rebooting...

```

All of the default master Routing Engines (re0) in the routing matrix reboot from the boot devices on which the software was just installed, which loads the new JUNOS Software. Each reboot operation can take between 5 and 10 minutes to complete.

```
...router_boot_sequence...
```

When the default master Routing Engine on the TX Matrix Plus router (**sfc0-re0**) finishes rebooting, the management console attached to the default master Routing Engine on the TX Matrix Plus router displays the login prompt.

```
var/pdb/profile_db initialized
```

```
Profile database initialized
Local package initialization:.
kern.securelevel: -1 -> 1
starting local daemons:.
Thu Jul 23 14:11:00 PDT 2009
```

```
login:
```

The new version of JUNOS software is loaded and running on the new default master Routing Engines

2. At a management console attached to the new default master Routing Engine (**re1**) on the TX Matrix Plus router, log in to the JUNOS CLI using a root administration account:

```
Thu Jul 23 14:11:00 PDT 2009
```

```
login: root
Password: ••••••••
```

```
--- JUNOS 9.6R1.2 built 2009-07-13 08:52:23 UTC
```

3. Issue the **show version** command to verify the version of the software installed:

```
root@mysfc0> show version
```

```
sfc0-re0:
```

```
-----
Hostname: mysfc0
Model: txp
JUNOS Base OS boot [9.6R1.2]
JUNOS Base OS Software Suite [9.6R1.2]
JUNOS Kernel Software Suite [9.6R1.2]
JUNOS Crypto Software Suite [9.6R1.2]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6R1.2]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6R1.2]
JUNOS Online Documentation [9.6R1.2]
JUNOS Voice Services Container package [9.6R1.2]
JUNOS Border Gateway Function package [9.6R1.2]
JUNOS Services AACL Container package [9.6R1.2]
JUNOS Services LL-PDF Container package [9.6R1.2]
JUNOS Services Stateful Firewall [9.6R1.2]
JUNOS AppId Services [9.6R1.2]
JUNOS IDP Services [9.6R1.2]
JUNOS Routing Software Suite [9.6R1.2]
```

```
lcc0-re0:
```

```
-----
Hostname: mylcc0
```

```

Model: t1600
JUNOS Base OS boot [9.6R1.2]
JUNOS Base OS Software Suite [9.6R1.2]
JUNOS Kernel Software Suite [9.6R1.2]
JUNOS Crypto Software Suite [9.6R1.2]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6R1.2]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6R1.2]
JUNOS Online Documentation [9.6R1.2]
JUNOS Voice Services Container package [9.6R1.2]
JUNOS Border Gateway Function package [9.6R1.2]
JUNOS Services AACL Container package [9.6R1.2]
JUNOS Services LL-PDF Container package [9.6R1.2]
JUNOS Services Stateful Firewall [9.6R1.2]
JUNOS AppId Services [9.6R1.2]
JUNOS IDP Services [9.6R1.2]
JUNOS Routing Software Suite [9.6R1.2]

```

lcc2-re0:

```

-----
Hostname: mylcc2
Model: t1600
JUNOS Base OS boot [9.6R1.2]
JUNOS Base OS Software Suite [9.6R1.2]
JUNOS Kernel Software Suite [9.6R1.2]
JUNOS Crypto Software Suite [9.6R1.2]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6R1.2]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6R1.2]
JUNOS Online Documentation [9.6R1.2]
JUNOS Voice Services Container package [9.6R1.2]
JUNOS Border Gateway Function package [9.6R1.2]
JUNOS Services AACL Container package [9.6R1.2]
JUNOS Services LL-PDF Container package [9.6R1.2]
JUNOS Services Stateful Firewall [9.6R1.2]
JUNOS AppId Services [9.6R1.2]
JUNOS IDP Services [9.6R1.2]
JUNOS Routing Software Suite [9.6R1.2]

```

Optional: Switching Routing Engine Mastership Back to the Default Master Routing Engines

The *default backup* Routing Engines (re1) are still operating as the *elected master* Routing Engines while the default master Routing Engines (re0) are operating as the elected backup Routing Engines.

To avoid confusion about which Routing Engine in any router is the master and which is the backup, you should switch the Routing Engine mastership assignment back to the original designations (with the re0 Routing Engines operating as the master Routing Engines and the re1 Routing Engines operating as the backup Routing Engines).



NOTE: Switching Routing Engine mastership causes traffic to be disrupted.

However, if you need to avoid another disruption to traffic while the Routing Engine mastership is switched on all the routers, you can skip this task.

To switch Routing Engine mastership back to the default master Routing Engine (re0) on all routers in the routing matrix:

1. Transfer Routing Engine mastership to the backup Routing Engine (re1) for all routers in the routing matrix. To do this, issue the **request chassis routing-engine master** command and specify the **switch** and **all-chassis** options.

At the warning message and prompt, type **yes** to allow the command to proceed:

```
root@mysfc0> request chassis routing-engine master switch all-chassis
warning: Traffic will be interrupted while the PFE is re-initialized
Toggle mastership between routing engines ? [yes,no] (no) yes
```

1cc0-re0:

```
-----
warning: Traffic will be interrupted while the PFE is re-initialized
Resolving mastership...
Complete. The local routing engine becomes the master.
```

1cc2-re0:

```
-----
warning: Traffic will be interrupted while the PFE is re-initialized
Resolving mastership...
Complete. The local routing engine becomes the master.
```

sfc0-re0:

```
-----
warning: Traffic will be interrupted while the PFE is re-initialized
Resolving mastership...
Complete. The local routing engine becomes the master.
```

2. Verify that Routing Engine mastership has indeed switched back to the default master Routing Engines (re0) in the routing matrix. To display Routing Engine mastership information, issue the **show chassis routing-engine** command at the TX Matrix Plus router:

```
root@mysfc0> show chassis routing-engine
sfc0-re0:
```

Routing Engine status:

Slot 0:

Current state	Master
Election priority	Master (default)
...output_truncated...	

Routing Engine status:

Slot 1:

Current state	Backup
Election priority	Backup (default)
...output_truncated...	

1cc0-re0:

Routing Engine status:

Slot 0:

Current state	Master
Election priority	Master (default)
...output_truncated...	

Routing Engine status:

Slot 1:

Current state	Backup
Election priority	Backup (default)
...output_truncated...	

lcc2-re0:

Routing Engine status:

Slot 0:

Current state	Master
Election priority	Master (default)
...output_truncated...	

Routing Engine status:

Slot 1:

Current state	Backup
Election priority	Backup (default)
...output_truncated...	

Restoring the Original Configuration

After you have installed the new JUNOS Software and are satisfied that it is running properly on all Routing Engines, restore the routing matrix to the original configuration (that existed before you deleted it at the beginning of this procedure) and then back up the configuration change.

To restore the previous configuration and then create a configuration backup, perform the following steps at the master Routing Engine of the TX Matrix Plus router:

1. Restore the routing matrix configuration that existed before you deleted it at the start of this procedure:

```
{master}
root@mysfc0-re0> configure
Entering configuration mode
```

```
{master}
root@mysfc0# rollback 1
load complete
```

2. Commit the configuration changes on all Routing Engines, which activates the original configuration, and then return to operational mode:

```
{master}
root@mysfc0-re0# commit synchronize and-quit
sfc0-re0:
configuration check succeeds
lcc0-re1:
commit complete
lcc0-re0:
commit complete
lcc2-re1:
commit complete
lcc2-re0:
```

```

commit complete
sfc0-re1:
commit complete
sfc0-re0:
commit complete

{master}
root@mysfc0-re0>

```

Backing Up the Routing Engines

Back up the currently running and active file system partitions on each Routing Engine to standby partitions that are not running. Specifically, the root file system (/) is backed up to /altroot, and /config is backed up to /altconfig. The root and /config file systems are on each Routing Engine's flash drive, and the /altroot and /altconfig file systems are on Routing Engine's hard drive.



CAUTION: After you run 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.

To back up the Routing Engines of the routing matrix:

1. To back up the master Routing Engines (re0) in the routing matrix, issue the `request system snapshot` command.

The following example shows sample output for backing up the master Routing Engines:

```

{master}
root@mysfc0-re0> request system snapshot
sfc0-re0:
-----
Verifying compatibility of destination media partitions...
Running newfs (220MB) on hard-disk media / partition (ad1s1a)...
Running newfs (24MB) on hard-disk media /config partition (ad1s1e)...
Copying '/dev/ad0s1a' to '/dev/ad1s1a' .. (this may take a few minutes)
Copying '/dev/ad0s1e' to '/dev/ad1s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config

1cc0-re0:
-----
Verifying compatibility of destination media partitions...
Running newfs (220MB) on hard-disk media / partition (ad1s1a)...
Running newfs (24MB) on hard-disk media /config partition (ad1s1e)...
Copying '/dev/ad0s1a' to '/dev/ad1s1a' .. (this may take a few minutes)
Copying '/dev/ad0s1e' to '/dev/ad1s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config

1cc2-re0:
-----
Verifying compatibility of destination media partitions...
Running newfs (220MB) on hard-disk media / partition (ad1s1a)...

```

```
Running newfs (24MB) on hard-disk media /config partition (ad1s1e)...
Copying '/dev/ad0s1a' to '/dev/ad1s1a' .. (this may take a few minutes)
Copying '/dev/ad0s1e' to '/dev/ad1s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

2. To transfer Routing Engine mastership back to the default backup Routing Engines, issue the `request chassis routing-engine master switch` command with the `switch` and `all-chassis` options.



CAUTION: Switching Routing Engine mastership causes traffic to be disrupted.

At the warning message and prompt, type **yes** to allow the operation to proceed

```
{master}
root@mysfc0-re0> request chassis routing-engine master switch all-chassis
```

```
Toggle mastership between routing engines ? [yes,no] (no) yes
```

```
1cc0-re0:
```

```
-----
Resolving mastership...
Complete. The other routing engine becomes the master.
```

```
1cc2-re0:
```

```
-----
Resolving mastership...
Complete. The other routing engine becomes the master.
```

```
sfc0-re0:
```

```
-----
Resolving mastership...
Complete. The other routing engine becomes the master.
```

3. To back up the backup Routing Engines (`re1`), issue the `request system snapshot` operational command at the backup Routing Engine on the TX Matrix Plus router.

The following examples shows sample command output for backing up the backup Routing Engines:

```
{master}
root@mysfc0-re1> request system snapshot
sfc0-re1:
```

```
-----
Verifying compatibility of destination media partitions...
Running newfs (220MB) on hard-disk media / partition (ad1s1a)...
Running newfs (24MB) on hard-disk media /config partition (ad1s1e)...
Copying '/dev/ad0s1a' to '/dev/ad1s1a' .. (this may take a few minutes)
Copying '/dev/ad0s1e' to '/dev/ad1s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

```
1cc0-re1:
```

```
-----
Verifying compatibility of destination media partitions...
Running newfs (223MB) on hard-disk media / partition (ad2s1a)...
```

```
Running newfs (24MB) on hard-disk media /config partition (ad2s1e)...
Copying '/dev/ad0s1a' to '/dev/ad2s1a' .. (this may take a few minutes)
Copying '/dev/ad0s1e' to '/dev/ad2s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

lcc2-re1:

```
-----
Verifying compatibility of destination media partitions...
Running newfs (220MB) on hard-disk media / partition (ad1s1a)...
Running newfs (24MB) on hard-disk media /config partition (ad1s1e)...
Copying '/dev/ad0s1a' to '/dev/ad1s1a' .. (this may take a few minutes)
Copying '/dev/ad0s1e' to '/dev/ad1s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Troubleshooting: Upgrade Failure Due to Insufficient Disk Space on page 85
 - Troubleshooting: Upgrade Failure Due to Insufficient DRAM on page 91
 - Troubleshooting: Upgrade Failure Due to Mixed Software on page 91

Troubleshooting: Software Upgrade Failure Due to Insufficient Free Disk Space on a Routing Matrix with a TX Matrix Plus Router

Problem In a routing matrix with a TX Matrix Plus router, a successful upgrade of the JUNOS Software requires sufficient free disk space on all Routing Engines of the TX Matrix Plus router and connected T1600 routers.

If the amount of free disk space on a Routing Engine is insufficient to install the JUNOS Software, you might receive a warning that the `/var` filesystem is low on free disk space, similar to the following message:

```
WARNING: The /var filesystem is low on free disk space.
WARNING: This package requires 1075136k free, but there
WARNING: is only 666502k available.
```

Cause The amount of free disk space necessary to upgrade a Routing Engine with a new version of the JUNOS Software can vary from one release to another. Check the *JUNOS Software Release Notes* for the software version you are installing.

To determine the amount of free disk space on each Routing Engine in the routing matrix, issue the `show system storage` command on the master Routing Engine and also on the backup Routing Engine on the TX Matrix Plus router. The command output is organized by chassis: `sfc0`, `lcc0`, `lcc1`, `lcc2`, and `lcc3`.



NOTE: When you issue the **show system storage** command on a TX Matrix Plus master Routing Engine, the command is broadcast to the master Routing Engines (only) on all T1600 routers in the routing matrix. Similarly, when you issue the **show system storage** command on a TX Matrix Plus backup Routing Engine, the command is broadcast to the backup Routing Engines (only) on all T1600 routers in the routing matrix.

The following example shows sample output for the **show system storage** command when it is issued on the master Routing Engine in slot 0 on the TX Matrix Plus router:

```
root@mysfc0> show system storage
sfc0-re0:
```

Filesystem	Size	Used	Avail	Capacity	Mounted on
/dev/ad0s1a	3.4G	179M	3.2G	5%	/
devfs	1.0K	1.0K	0B	100%	/dev
devfs	1.0K	1.0K	0B	100%	/dev/
/dev/md0	33M	33M	0B	100%	/packages/mnt/jbase
/dev/md1	217M	217M	0B	100%	
/packages/mnt/jkernel-9.6-20090624.0					
/dev/md2	66M	66M	0B	100%	
/packages/mnt/jpfe-T-9.6-20090624.0					
/dev/md3	4.1M	4.1M	0B	100%	
/packages/mnt/jdocs-9.6-20090624.0					
/dev/md4	57M	57M	0B	100%	
/packages/mnt/jroute-9.6-20090624.0					
/dev/md5	15M	15M	0B	100%	
/packages/mnt/jcrypto-9.6-20090624.0					
/dev/md6	34M	34M	0B	100%	
/packages/mnt/jpfe-common-9.6-20090624.0					
/dev/md7	2.0G	8.0K	1.8G	0%	/tmp
/dev/md8	2.0G	878K	1.8G	0%	/mfs
/dev/ad0s1e	383M	90K	379M	0%	/config
procfs	4.0K	4.0K	0B	100%	/proc
/dev/ad1s1f	24G	8.7G	13G	40%	/var

```
1cc0-re0:
```

Filesystem	Size	Used	Avail	Capacity	Mounted on
/dev/ad0s1a	3.4G	179M	2.9G	6%	/
devfs	1.0K	1.0K	0B	100%	/dev
devfs	1.0K	1.0K	0B	100%	/dev/
/dev/md0	33M	33M	0B	100%	/packages/mnt/jbase
/dev/md1	217M	217M	0B	100%	
/packages/mnt/jkernel-9.6-20090629.0					
/dev/md2	66M	66M	0B	100%	
/packages/mnt/jpfe-T-9.6-20090629.0					
/dev/md3	4.1M	4.1M	0B	100%	
/packages/mnt/jdocs-9.6-20090629.0					
/dev/md4	57M	57M	0B	100%	
/packages/mnt/jroute-9.6-20090629.0					
/dev/md5	15M	15M	0B	100%	
/packages/mnt/jcrypto-9.6-20090629.0					
/dev/md6	34M	34M	0B	100%	
/packages/mnt/jpfe-common-9.6-20090629.0					
/dev/md7	2.0G	8.0K	1.8G	0%	/tmp
/dev/md8	2.0G	344K	1.8G	0%	/mfs

/dev/ad0s1e	383M	14K	352M	0%	/config
procfs	4.0K	4.0K	0B	100%	/proc
/dev/ad1s1f	24G	3.7G	18G	17%	/var

lcc2-re0:

Filesystem	Size	Used	Avail	Capacity	Mounted on
/dev/ad0s1a	3.4G	178M	2.9G	6%	/
devfs	1.0K	1.0K	0B	100%	/dev
devfs	1.0K	1.0K	0B	100%	/dev/
/dev/md0	32M	32M	0B	100%	/packages/mnt/jbase
/dev/md1	217M	217M	0B	100%	
/packages/mnt/jkernel-9.6I20090626					
/dev/md2	66M	66M	0B	100%	
/packages/mnt/jpfe-T-9.6I20090626					
/dev/md3	4.1M	4.1M	0B	100%	
/packages/mnt/jdocs-9.6I20090626					
/dev/md4	57M	57M	0B	100%	
/packages/mnt/jroute-9.6I20090626					
/dev/md5	15M	15M	0B	100%	
/packages/mnt/jcrypto-9.6I20090626					
/dev/md6	34M	34M	0B	100%	
/packages/mnt/jpfe-common-9.6I20090626					
/dev/md7	2.0G	10.0K	1.8G	0%	/tmp
/dev/md8	2.0G	346K	1.8G	0%	/mfs
/dev/ad0s1e	383M	14K	352M	0%	/config
procfs	4.0K	4.0K	0B	100%	/proc
/dev/ad1s1f	24G	12G	9.4G	57%	/var

Solution To resolve the issue of insufficient free disk space on a Routing Engine to perform a JUNOS Software upgrade on that router, perform the following steps:

1. To free up space in the file system, you can log in to a particular Routing Engine and then issue the **request system storage cleanup** command. This command deletes rotating log files in **/var/log** that are not current, temporary files in **/var/tmp** that have not been modified within the last two days, and all crash files in **/var/crash**. The command output first lists the files proposed for deletion and then prompts you to proceed to delete these storage cleanup candidates.



NOTE: Although the command prompts you before deleting files, you can issue the command with the **dry-run** option, which causes the command to only list the cleanup candidates without offering to remove them.

The following example shows sample output for the **request system storage cleanup** command with the **dry-run** option included:

```
root@mysfc0> request system storage cleanup dry-run
```

```
Currently rotating log files, please wait.
This operation can take up to a minute.
```

```
List of files to delete:
```

Size	Date	Name
2B	Jun 16 12:46	/var/crash/bounds
80.4M	Jun 15 14:56	/var/crash/cores/kernel.0.090615.1455
80.4M	Jun 15 15:46	/var/crash/cores/kernel.1.090615.1546
80.4M	Jun 15 15:58	/var/crash/cores/kernel.2.090615.1558
80.4M	Jun 16 12:47	/var/crash/cores/kernel.3.090616.1246
30.6M	Jun 15 15:05	/var/crash/cores/vmcore.0.090615.1455.tgz
33.5M	Jun 15 16:33	/var/crash/cores/vmcore.1.090615.1546.tgz
35.9M	Jun 15 17:52	/var/crash/cores/vmcore.2.090615.1558.tgz
48.0M	Jun 16 13:47	/var/crash/cores/vmcore.3.090616.1246.tgz
504B	Jun 15 14:55	/var/crash/info.0
504B	Jun 15 15:46	/var/crash/info.1
505B	Jun 15 15:58	/var/crash/info.2
505B	Jun 16 12:46	/var/crash/info.3
258.1K	Jun 23 16:26	/var/log/chassisd.1.gz
259.2K	Jun 17 08:41	/var/log/chassisd.2.gz
39.6K	Jun 15 13:03	/var/log/dcd.0.gz
716B	Jun 15 13:43	/var/log/install.0.gz
1669B	Jun 13 18:05	/var/log/install.1.gz

...output_truncated...

4B	Jun 27 10:37	/var/tmp/idp_license_info
263.6M	Jun 11 16:25	/var/tmp/jbundle-9.6-20090611.1-domestic.tgz
263.8M	Jun 24 13:44	/var/tmp/jbundle-9.6-20090624.0-domestic.tgz
268.7M	Jun 11 15:14	/var/tmp/jinstall-9.6-20090611.1-domestic-signed.tgz
268.8M	Jun 23 23:04	/var/tmp/jinstall-9.6-20090624.0-domestic-signed.tgz
22.9K	Jun 22 10:33	/var/tmp/memprobe
1024.0K	Jun 16 16:16	/var/tmp/p737a_BIOS_0_A_D.ROM
1024.0K	Jun 16 16:17	/var/tmp/p737f_BIOS_0_0_S.ROM

Delete these files ? [yes,no] (no)

2. You can use the following file management operational commands to display information about the directories and files on a Routing Engine, but you must include the chassis and Routing Engine specifiers in the pathname:
 - To display detailed information about a file or directory contents, issue the **file list** command with the **detail** option.
 - To display the contents of a file, issue the **file show** command.
 - To compare the contents of two files, issue the **file compare** command.
3. In addition to or instead of using the **request system storage cleanup** command to delete unnecessary files, you can use the following file management commands to delete or move files on a Routing Engine, but you must include the chassis and Routing Engine specifiers in the pathname:
 - To delete files, issue the **file delete** command.
 - To copy files, issue the **file copy** command.
 - To rename files, issue the **file rename** command.

If you elect to use the **request system storage cleanup** command to automatically delete files, issue the command without the **dry-run** option, and then type **yes** when the CLI prompts to proceed with deleting the candidate files. The following truncated example shows sample output for the **request system storage cleanup** command:

```

root@mysfc0> request system storage cleanup
Currently rotating log files, please wait.
This operation can take up to a minute.

List of files to delete:

      Size Date      Name
      2B Jun 16 12:46 /var/crash/bounds
      80.4M Jun 15 14:56 /var/crash/cores/kernel.0.090615.1455
      80.4M Jun 15 15:46 /var/crash/cores/kernel.1.090615.1546

...output_truncated...

1024.0K Jun 16 16:16 /var/tmp/p737a_BIOS_0_A_D.ROM
1024.0K Jun 16 16:17 /var/tmp/p737f_BIOS_0_0_S.ROM
Delete these files ? [yes,no] (no) yes

```

4. After sufficient free disk space is available on the Routing Engines on which you are upgrading the JUNOS Software, attempt the upgrade procedure again.

Related Topics

- Routing Matrix with a TX Matrix Plus Router Solutions Page
- Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
- Roadmap for Configuring the Routing Matrix on page 15
- Example Configuration for the Routing Matrix on page 33
- Upgrading the JUNOS Software on the Routing Matrix on page 61

- Troubleshooting: Upgrade Failure Due to Insufficient DRAM on page 91
- Troubleshooting: Upgrade Failure Due to Mixed Software on page 91

Troubleshooting: Software Upgrade Failure Due to Insufficient Available DRAM on a Routing Matrix with a TX Matrix Plus Router

Problem In a routing matrix with a TX Matrix Plus router, a successful upgrade of the JUNOS Software requires sufficient available DRAM on all Routing Engines of the TX Matrix Plus router and connected T1600 routers.

Cause Failure of a JUNOS Software upgrade due to insufficient available DRAM on a Routing Engine occurs infrequently. When it does occur, there are various causes and therefore various solutions to the issue.

If you suspect that insufficient available DRAM is preventing a successful upgrade of the JUNOS Software, contact your technical support representative.

Solution If you suspect that insufficient available DRAM is preventing a successful upgrade of the JUNOS Software, contact your technical support representative.



NOTE: It can be useful to issue the `request support information` command before contacting technical support. The command runs several JUNOS operational commands to obtain detailed information about the system hardware and software. Because the amount of command output is quite large, we recommend that you save the output to a file.

In some cases of insufficient available DRAM, a different or upgraded Routing Engine with a larger amount of DRAM is required. However, most DRAM issues that prevent a successful upgrade can be resolved by removing any superfluous files from the `/root` and `/tmp` directories.

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61
 - Troubleshooting: Upgrade Failure Due to Insufficient Disk Space on page 85
 - Troubleshooting: Upgrade Failure Due to Mixed Software on page 91

Troubleshooting: Software Upgrade Failure Due to Mixed Software Versions on a Routing Matrix with a TX Matrix Plus Router

Problem In a routing matrix with a TX Matrix Plus router, a successful upgrade of the JUNOS Software requires that all Routing Engines of the TX Matrix Plus router and connected T1600 routers are running the same version of software.

If the software versions on the Routing Engines are not aligned, the software upgrade process will return an error.

Cause Different versions of the JUNOS Software can have incompatible message formats, particularly if graceful routing engine switchover (GRES) is enabled. Because the steps in the upgrade process include changing Routing Engine mastership, running the same version of software is recommended.

To display the hostname and version information about the software running on all routers in a routing matrix, issue the **show version** command with the **invoke-on all-routing-engines** option.

In the following example, a routing matrix consists of a TX Matrix Plus router (**sfc0**) and two connected T1600 routers (**lcc0** and **lcc2**), with all three routers containing redundant host subsystems. The sample output from the **show version** command shows that a different version of the JUNOS Software is installed on the TX Matrix Plus router's backup Routing Engine (**sfc0-re1**), while all the other Routing Engines have JUNOS Release **9.6R1.0** installed.

In the following sample output, the backup Routing Engine on the TX Matrix Plus router is running a version of JUNOS Software that is different from the software on the other Routing Engines:

```
root@mysfc0> show version invoke-on all-routing-engines
sfc0-re0:
-----
Hostname: mysfc0
Model: txp
JUNOS Base OS boot [9.6-20090713.0]
JUNOS Base OS Software Suite [9.6-20090713.0]
JUNOS Kernel Software Suite [9.6-20090713.0]
JUNOS Crypto Software Suite [9.6-20090713.0]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.0]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.0]
JUNOS Online Documentation [9.6-20090713.0]
JUNOS Voice Services Container package [9.6-20090713.0]
JUNOS Border Gateway Function package [9.6-20090713.0]
JUNOS Services ACL Container package [9.6-20090713.0]
JUNOS Services LL-PDF Container package [9.6-20090713.0]
JUNOS Services Stateful Firewall [9.6-20090713.0]
JUNOS AppId Services [9.6-20090713.0]
JUNOS IDP Services [9.6-20090713.0]
JUNOS Routing Software Suite [9.6-20090713.0]
```

```
lcc0-re0:
-----
Hostname: mylcc0
Model: t1600
JUNOS Base OS boot [9.6-20090713.0]
JUNOS Base OS Software Suite [9.6-20090713.0]
JUNOS Kernel Software Suite [9.6-20090713.0]
JUNOS Crypto Software Suite [9.6-20090713.0]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.0]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.0]
JUNOS Online Documentation [9.6-20090713.0]
JUNOS Voice Services Container package [9.6-20090713.0]
JUNOS Border Gateway Function package [9.6-20090713.0]
```

```
JUNOS Services AACL Container package [9.6-20090713.0]
JUNOS Services LL-PDF Container package [9.6-20090713.0]
JUNOS Services Stateful Firewall [9.6-20090713.0]
JUNOS AppId Services [9.6-20090713.0]
JUNOS IDP Services [9.6-20090713.0]
JUNOS Routing Software Suite [9.6-20090713.0]
```

lcc2-re0:

```
-----
Hostname: mylcc2
Model: t1600
JUNOS Base OS boot [9.6-20090713.0]
JUNOS Base OS Software Suite [9.6-20090713.0]
JUNOS Kernel Software Suite [9.6-20090713.0]
JUNOS Crypto Software Suite [9.6-20090713.0]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.0]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.0]
JUNOS Online Documentation [9.6-20090713.0]
JUNOS Voice Services Container package [9.6-20090713.0]
JUNOS Border Gateway Function package [9.6-20090713.0]
JUNOS Services AACL Container package [9.6-20090713.0]
JUNOS Services LL-PDF Container package [9.6-20090713.0]
JUNOS Services Stateful Firewall [9.6-20090713.0]
JUNOS AppId Services [9.6-20090713.0]
JUNOS IDP Services [9.6-20090713.0]
JUNOS Routing Software Suite [9.6-20090713.0]
```

sfc0-re1:

```
-----
Hostname: mysfc0_alt_re
Model: txp
JUNOS Base OS boot [1.1-1111111.1]
JUNOS Base OS Software Suite [1.1-1111111.1]
JUNOS Kernel Software Suite [1.1-1111111.1]
JUNOS Crypto Software Suite [1.1-1111111.1]
JUNOS Packet Forwarding Engine Support (M/T Common) [1.1-1111111.1]
JUNOS Packet Forwarding Engine Support (T-Series) [1.1-1111111.1]
JUNOS Online Documentation [1.1-1111111.1]
JUNOS Voice Services Container package [1.1-1111111.1]
JUNOS Border Gateway Function package [1.1-1111111.1]
JUNOS Services AACL Container package [1.1-1111111.1]
JUNOS Services LL-PDF Container package [1.1-1111111.1]
JUNOS Services Stateful Firewall [1.1-1111111.1]
JUNOS AppId Services [1.1-1111111.1]
JUNOS IDP Services [1.1-1111111.1]
JUNOS Routing Software Suite [1.1-1111111.1]
```

lcc0-re1:

```
-----
Hostname: mylcc0_alt_re
Model: t1600
JUNOS Base OS boot [9.6-20090713.0]
JUNOS Base OS Software Suite [9.6-20090713.0]
JUNOS Kernel Software Suite [9.6-20090713.0]
JUNOS Crypto Software Suite [9.6-20090713.0]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.0]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.0]
JUNOS Online Documentation [9.6-20090713.0]
JUNOS Voice Services Container package [9.6-20090713.0]
JUNOS Border Gateway Function package [9.6-20090713.0]
JUNOS Services AACL Container package [9.6-20090713.0]
```

```
JUNOS Services LL-PDF Container package [9.6-20090713.0]
JUNOS Services Stateful Firewall [9.6-20090713.0]
JUNOS AppId Services [9.6-20090713.0]
JUNOS IDP Services [9.6-20090713.0]
JUNOS Routing Software Suite [9.6-20090713.0]
```

1cc2-re1:

```
-----
Hostname: my1cc2_alt_re
Model: t1600
JUNOS Base OS boot [9.6-20090713.0]
JUNOS Base OS Software Suite [9.6-20090713.0]
JUNOS Kernel Software Suite [9.6-20090713.0]
JUNOS Crypto Software Suite [9.6-20090713.0]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.6-20090713.0]
JUNOS Packet Forwarding Engine Support (T-Series) [9.6-20090713.0]
JUNOS Online Documentation [9.6-20090713.0]
JUNOS Voice Services Container package [9.6-20090713.0]
JUNOS Border Gateway Function package [9.6-20090713.0]
JUNOS Services ACL Container package [9.6-20090713.0]
JUNOS Services LL-PDF Container package [9.6-20090713.0]
JUNOS Services Stateful Firewall [9.6-20090713.0]
JUNOS AppId Services [9.6-20090713.0]
JUNOS IDP Services [9.6-20090713.0]
JUNOS Routing Software Suite [9.6-20090713.0]
```

Solution To synchronize the JUNOS Software versions running on the Routing Engines in a routing matrix, perform the following steps on each Routing Engine that has an incorrect version of JUNOS Software installed:

1. Log in to the Routing Engine that has the incorrect version of JUNOS Software installed. To log in to a Routing Engine, issue the **request routing-engine login** command.

In the example scenario, the backup Routing Engine on the TX Matrix Plus router is running an incorrect version of software. To log in to that Routing Engine, you would enter the following command:

```
root@mysfc0> request routing-engine login other-routing-engine

--- JUNOS 1.1-1111111.1 built 1111-11-11 11:11:11 UTC
% cli
root@mysfc0_alt_re>
```



NOTE: Throughout the software upgrade procedure instructions, we assume that the master Routing Engines are in slot 0 and the backup Routing Engines are in slot 1, which is the recommended installation. Therefore, we could issue the **request routing-engine login** command with the **re1** option (instead of the **other-routing-engine** option) and obtain the same results.

2. If the appropriate software package or bundle is not already present on the Routing Engine that has the incorrect version of JUNOS Software installed, copy the software onto the Routing Engine.

For more information, see the *JUNOS Software Release Notes* and the *JUNOS Software Installation and Upgrade Guide*.

3. To install a software package or bundle on the Routing Engine that has the incorrect version of JUNOS Software installed, use the **request system software add** command.
4. When all the Routing Engines in the routing matrix are running the same version of the JUNOS Software, try the software upgrade procedure again.

Related Topics

- Routing Matrix with a TX Matrix Plus Router Solutions Page
- Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
- Roadmap for Configuring the Routing Matrix on page 15
- Example Configuration for the Routing Matrix on page 33
- Upgrading the JUNOS Software on the Routing Matrix on page 61
- Troubleshooting: Upgrade Failure Due to Insufficient Disk Space on page 85
- Troubleshooting: Upgrade Failure Due to Insufficient DRAM on page 91

Chapter 6

Managing Files and Processes

- Managing Files on Routing Engines in a Routing Matrix with a TX Matrix Plus Router on page 97
- Managing System Processes in a Routing Matrix with a TX Matrix Plus Router on page 100

Managing Files on Routing Engines in a Routing Matrix with a TX Matrix Plus Router

This topic summarizes key file management issues useful in operating a routing matrix with a TX Matrix Plus router:

- Displaying a List of Files on page 97
- Displaying the Contents of a File on page 98
- Copying Files on page 99
- Renaming Files on page 99
- Deleting Files on page 99

Displaying a List of Files

To display a list of files on a Routing Engine in a routing matrix with a TX Matrix Plus router, issue the **file list** operational management command.

- To display a list of files on a different chassis, include the chassis specifier (**lcc0**, **lcc1**, **lcc2**, or **lcc3**) in the directory or file pathname.
- To display a list of files on a different Routing Engine, include the Routing Engine specifier (**re0** or **re1**) in the directory or file pathname.

The following sample command displays the list of files in the **/var/tmp** directory on Routing Engine **re0** in the line-card chassis **lcc0**:

```
user@host> file list lcc0-re0:/var/tmp
lcc0-re0:
-----
/var/tmp/:
.gdbinit
.pccardd
Test/
chassisd*
chassisd.nathan*
```

```

check_time*
cores/
diagTestPrep*
diagtest*
diagtest.regress*
do_switchovers*
dump_test*
err.manoj.log
esw_clearstats*
esw_counter*
esw_debug*
esw_debug_ge*
esw_filt_test*
esw_filter_tnp_addr*
esw_getstats*
esw_phy*
esw_stats*

```

Displaying the Contents of a File

To display the contents of a file, issue the `file show` operational command. To specify a file that is not local to the Routing Engine from which the command is issued, include chassis and Routing Engine information in the filename.

The following sample command displays the contents of the `.gdbinit` file in the `/var/tmp` directory on the master Routing Engine of the T1600 router `lcc0`:

```

user@host> file show lcc0-re0:/var/tmp/.gdbinit
lcc0-re0:
-----
#####
# Settings
#####

set print pretty

#####
# Basic stuff
#####

define msgbuf
    printf "%s", msgbufp->msg_ptr
end
# hex dump of a block of memory
# usage: dump address length
define dump
    p $arg0, $arg1
    set $ch = $arg0
    set $j = 0
    set $n = $arg1
    while ($j < $n)
        #printf "%x %x ",&$ch[$j],$ch[$j]
        printf "%x ",$ch[$j]
        set $j = $j + 1
        if (!($j % 16))
            printf "\n"
        end
    end
end

```

```

    end
end

```

Copying Files

To copy files from one place to another on the same router or between different routers in the routing matrix, use the **file copy** operational management command. To specify a file that is not local to the Routing Engine from which the command is issued, include chassis and Routing Engine information in the filename.

The following sample command is used to copy the file **sample.txt** from the **/tmp** directory on **sfc0-re1** (the backup Routing Engine in the TX Matrix Plus router) to the **/var/tmp** directory on **lcc0-re1** (the backup Routing Engine in one of the connected T1600 routers):

```
user@host> file copy sfc0-re1:/tmp/sample.txt lcc0-re1:/var/tmp
```

Renaming Files

To rename a file, use the **file rename** operational management command. To specify a file that is not local to the Routing Engine from which the command is issued, include chassis and Routing Engine information in the filename.

The following sequence of sample commands lists the files in the **/var/tmp** directory on **sfc0-re0** (the master Routing Engine of the TX Matrix Plus router), renames one of the files (changing **dcd.core** to **dcd.core.990415**), and then lists the files in the **/var/tmp** directory again to show the newly named file:

```

user@host> file list sfc0-re0:/var/tmp
dcd.core
rpd.core
snmpd.core
user@host> file rename sfc0-re0:/var/tmp/dcd.core /var/tmp/dcd.core.990413

user@host> file list sfc0-re0:/var/tmp
dcd.core.990413
rpd.core
snmpd.core

```

Deleting Files

To delete a file, use the **file delete** operational management command. To specify a file that is not local to the Routing Engine from which the command is issued, include chassis and Routing Engine information in the filename.

The following sequence of sample command lists the files in the **/var/tmp** directory on **sfc0-re0** (the master Routing Engine on the TX Matrix Plus router), deletes the file **snmpd.core** from that directory, and then lists the files in the directory again:

```
user@host> file list lcc0-re0:/var/tmp
```

```
dcd.core
rpd.core
snmpd.core
user@host> file delete lcc0-re0:/var/tmp/snmpd.core
user@host> file list /var/tmp
dcd.core
rpd.core
```

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Managing System Processes in a Routing Matrix with a TX Matrix Plus Router

Some system processes in a routing matrix with a TX Matrix Plus run on the TX Matrix Plus router and some run on the T1600 routers. For example, the routing protocol process (rpd) runs exclusively on the TX Matrix Plus router. To restart the routing protocol process for the entire routing matrix, issue the **restart routing** command on the TX Matrix Plus router:

```
user@host> restart routing ?
Possible completions:
<[Enter]>      Execute this command
gracefully     Gracefully restart the process
immediately    Immediately restart (SIGKILL) the process
logical-system Name of logical system
soft           Soft reset (SIGHUP) the process
|             Pipe through a command
```

Other processes run on both the TX Matrix Plus router and the T1600 routers. To restart the chassis process that manages PICs, FPCs, and other hardware components, issue the **restart chassis-control** command on the TX Matrix Plus router and select the **all**, **all-lcc**, or **lcc lcc-number** option:

```
user@host> restart chassis-control ?
Possible completions:
<[Enter]>      Execute this command
all-chassis    Restart software process on all chassis
all-lcc        Restart software process on all LCC chassis
gracefully     Gracefully restart the process
immediately    Immediately restart (SIGKILL) the process
lcc            Restart software process on specific LCC (0..3)
sfc            Restart software process on SFC (0..0)
soft           Soft reset (SIGHUP) the process
|             Pipe through a command
```

To restart the SNMP process, issue the **restart snmp** command on the TX Matrix Plus router and select the **all**, **all-lcc**, or **lcc lcc-number** option:

```
user@host> restart snmp ?
```

Possible completions:

<[Enter]>	Execute this command
all-chassis	Restart software process on all chassis
all-lcc	Restart software process on all LCC chassis
gracefully	Gracefully restart the process
immediately	Immediately restart (SIGKILL) the process
lcc	Restart software process on specific LCC (0..3)
sfc	Restart software process on SFC (0..0)
soft	Soft reset (SIGHUP) the process
	Pipe through a command

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Chapter 7

Managing Hardware Components

- Displaying Chassis Physical Locations for a Routing Matrix with a TX Matrix Plus Router on page 103
- Displaying the Status of the Switching Fabric Topology of a Routing Matrix with a TX Matrix Plus Router on page 104
- Displaying the Status of Control Board Ethernet Switch Ports in a Routing Matrix with a TX Matrix Plus Router on page 107
- Enabling and Disabling Hardware Components of a Routing Matrix with a TX Matrix Plus Router on page 111
- Rebooting and Halting Hardware Components of the Routing Matrix with a TX Matrix Plus Router on page 114

Displaying Chassis Physical Locations for a Routing Matrix with a TX Matrix Plus Router

In a routing matrix with a TX Matrix Plus router, you can use the `show chassis location` operational command to display the physical locations of hardware components in the routing matrix:

- To display the router physical location descriptions defined in the `location` configuration statement at the `[edit system]` hierarchy level, issue the `show chassis location` operational command. To limit the command output to the `location` configuration settings for the TX Matrix Plus router (SFC) only, include the `sfc 0` option. To limit the command output to the `location` configuration settings for a particular T1600 router (LCC), include the `lcc lcc-number` option. For example:

```
user@host> show chassis location sfc 0
```

```
sfc0-re0:
```

```
-----  
country-code: US  
postal-code: 94404  
Building: Building 2, Floor: 2
```

- To display the numbers of the connected Flexible PIC Concentrators (FPCs) in both routing matrix “global numbering” mode (0 through 31, as used to specify an interface name in a routing matrix) and associated “local numbering” mode (LCC number and FPC slot number), include the `fpc` option. Example:

```
user@host> show chassis location fpc
```

Global FPC	LCC	Local FPC
1	0	1
2	0	2
3	0	3
16	2	6
17	2	1
19	2	3

- To display the T1600 router (or line-card chassis) number and FPC number associated with a particular interface name, include the `interface by-name interface-name` option. For example:

```
user@host> show chassis location interface by-name ge-16/3/7.1
```

```
Interface Name: ge-16/3/7.0, LCC: 2, FPC: 6
```

- To display the interface name associated with a particular FPC, include the `interface by-slot lcc lcc-number fpc fpc-slot-number` option. For example:

```
user@host> show chassis location interface by-slot lcc 2 fpc 6
```

```
Global FPC: 16
```

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Displaying the Status of the Switching Fabric Topology of a Routing Matrix with a TX Matrix Plus Router

In a routing matrix with a TX Matrix Plus router, each of the five control planes contains various types of Switch Interface Boards (SIBs) that forward packets from a source Packet Forwarding Engine to a destination Packet Forwarding Engine.

You can use the `show chassis fabric topology` operational command to check the status of the SIB connections between the TX Matrix Plus router and the connected T1600 routers:

- To display the fabric topology states for the entire routing matrix, issue the `show chassis fabric topology` command. To limit the command output to a particular TXP-F13 SIB, also include the SIB slot number as a command option.

- To display the fabric topology states for the TX Matrix Plus router, include the **sfc number** option. To further limit the command output to a particular TXP-F13 SIB, also include the SIB slot number as a command option.
- To display the fabric topology states for a particular T1600 router, include the **lcc lcc-number** option. To further limit the command output to a particular TXP-F13 SIB, also include the SIB slot number as a command option.

To display the fabric topology state for the TX Matrix Plus router only, issue the **show fabric topology** command with the **sfc** option. The command output displays the fabric link status for transmit-side links and receive-side links for each of the 16 TXP-F13 SIB slots in the TX Matrix Plus router. For every TXP-F13 SIB, the following information is listed:

- Transmit-side link status—The state of each link from the TXP-F13 SIB to the TXP-T1600 SIB on a directly connected T1600 router. A TXP-F13 SIB connects to 2 different T1600 routers, and each connection consists of 128 links.
- Receive-side link status—The state of each link to the TXP-F13 SIB from the TXP-T1600 SIB on a directly connected T1600 router.

Example:

```
user@host> show chassis fabric topology sfc 0 0
sfc0-re0:
```

```
-----
SIB:0
```

```
=====
```

```
Out-Links:
```

```
=====
```

```
SFC0_F13_SIB_36      -> LCC00_ST_SIB_L00      Status
```

```
=====
```

```
SF_3_00_FB_D(04,11) -> FPC0_T_SG(0,0,0)_FB_D(04,11)  OK
SF_3_00_FB_D(04,10) -> FPC0_T_SG(0,0,1)_FB_D(04,10)  OK
SF_3_00_FB_D(04,09) -> FPC0_T_SG(0,0,2)_FB_D(04,09)  OK
SF_3_00_FB_D(04,08) -> FPC0_T_SG(0,0,3)_FB_D(04,08)  OK
SF_3_00_FB_D(04,07) -> FPC0_T_SG(0,0,4)_FB_D(04,07)  OK
SF_3_00_FB_D(04,06) -> FPC0_T_SG(0,0,5)_FB_D(04,06)  OK
SF_3_00_FB_D(04,05) -> FPC0_T_SG(0,0,6)_FB_D(04,05)  OK
SF_3_00_FB_D(04,04) -> FPC0_T_SG(0,0,7)_FB_D(04,04)  OK
SF_3_01_FB_B(16,11) -> FPC4_T_SG(2,0,0)_FB_B(16,11)  OK
```

```
...output_truncated...
```

```
SF_3_15_FB_C(09,08) -> FPC3_B_SG(1,3,7)_FB_C(09,08)  OK
PG(0,0)_FB_D(05,11) -> PC(0,0)_FB_D(05,11)          OK
PG(1,0)_FB_C(11,11) -> PC(1,0)_FB_C(11,11)          OK
PG(2,0)_FB_B(17,11) -> PC(2,0)_FB_B(17,11)          OK
PG(3,0)_FB_A(23,11) -> PC(3,0)_FB_A(23,11)          OK
```

```
In-Links:
```

```
=====
```

```
LCC00_ST_SIB_L00      -> SFC0_F13_SIB_36      Status
```

```
=====
```

```
FPC0_T_SG(0,0,0)_FB_D(01,11) -> SF_1_00_FB_D(01,11)  LOS
FPC0_T_SG(0,0,1)_FB_D(01,10) -> SF_1_00_FB_D(01,10)  LOS
```

```

FPC0_T_SG(0,0,2)_FB_D(01,09) -> SF_1_00_FB_D(01,09)  LOS
FPC0_T_SG(0,0,3)_FB_D(01,08) -> SF_1_00_FB_D(01,08)  LOS
FPC0_T_SG(0,0,4)_FB_D(01,07) -> SF_1_00_FB_D(01,07)  LOS
FPC0_T_SG(0,0,5)_FB_D(01,06) -> SF_1_00_FB_D(01,06)  LOS
FPC0_T_SG(0,0,6)_FB_D(01,05) -> SF_1_00_FB_D(01,05)  LOS
FPC0_T_SG(0,0,7)_FB_D(01,04) -> SF_1_00_FB_D(01,04)  LOS
FPC4_T_SG(2,0,0)_FB_B(13,11) -> SF_1_01_FB_B(13,11)  LOS

```

...output_truncated...

```

FPC3_B_SG(1,3,7)_FB_C(06,08) -> SF_1_15_FB_C(06,08)  LOS
PG(0,0)_FB_D(02,11)           -> PC(0,0)_FB_D(02,11)  LOS
PG(1,0)_FB_C(08,11)           -> PC(1,0)_FB_C(08,11)  LOS
PG(2,0)_FB_B(14,11)           -> PC(2,0)_FB_B(14,11)  LOS
PG(3,0)_FB_A(20,11)           -> PC(3,0)_FB_A(20,11)  LOS

```

Out-Links:

```

=====
SFC0_F13_SIB_36      -> LCC01_ST_SIB_L00                      Status
=====
SF_3_00_FB_D(28,11) -> FPC0_T_SG(0,0,0)_FB_D(28,11)  OK
SF_3_00_FB_D(28,10) -> FPC0_T_SG(0,0,1)_FB_D(28,10)  OK
SF_3_00_FB_D(28,09) -> FPC0_T_SG(0,0,2)_FB_D(28,09)  OK
SF_3_00_FB_D(28,08) -> FPC0_T_SG(0,0,3)_FB_D(28,08)  OK
SF_3_00_FB_D(28,07) -> FPC0_T_SG(0,0,4)_FB_D(28,07)  OK
SF_3_00_FB_D(28,06) -> FPC0_T_SG(0,0,5)_FB_D(28,06)  OK
SF_3_00_FB_D(28,05) -> FPC0_T_SG(0,0,6)_FB_D(28,05)  OK
SF_3_00_FB_D(28,04) -> FPC0_T_SG(0,0,7)_FB_D(28,04)  OK
SF_3_01_FB_B(40,11) -> FPC4_T_SG(2,0,0)_FB_B(40,11)  OK

```

...output_truncated...

```

SF_3_15_FB_C(33,08) -> FPC3_B_SG(1,3,7)_FB_C(33,08)  OK
PG(4,0)_FB_D(29,11) -> PC(4,0)_FB_D(29,11)  OK
PG(5,0)_FB_C(35,11) -> PC(5,0)_FB_C(35,11)  OK
PG(6,0)_FB_B(41,11) -> PC(6,0)_FB_B(41,11)  OK
PG(7,0)_FB_A(47,11) -> PC(7,0)_FB_A(47,11)  OK

```

In-Links:

```

=====
LCC01_ST_SIB_L00      -> SFC0_F13_SIB_36                      Status
=====
FPC0_T_SG(0,0,0)_FB_D(25,11) -> SF_1_00_FB_D(25,11)  LOS
FPC0_T_SG(0,0,1)_FB_D(25,10) -> SF_1_00_FB_D(25,10)  LOS
FPC0_T_SG(0,0,2)_FB_D(25,09) -> SF_1_00_FB_D(25,09)  LOS
FPC0_T_SG(0,0,3)_FB_D(25,08) -> SF_1_00_FB_D(25,08)  LOS
FPC0_T_SG(0,0,4)_FB_D(25,07) -> SF_1_00_FB_D(25,07)  LOS
FPC0_T_SG(0,0,5)_FB_D(25,06) -> SF_1_00_FB_D(25,06)  LOS
FPC0_T_SG(0,0,6)_FB_D(25,05) -> SF_1_00_FB_D(25,05)  LOS
FPC0_T_SG(0,0,7)_FB_D(25,04) -> SF_1_00_FB_D(25,04)  LOS
FPC4_T_SG(2,0,0)_FB_B(37,11) -> SF_1_01_FB_B(37,11)  LOS

```

...output_truncated...

```

FPC3_B_SG(1,3,7)_FB_C(30,08) -> SF_1_15_FB_C(30,08)  LOS
PG(4,0)_FB_D(26,11)           -> PC(4,0)_FB_D(26,11)  LOS
PG(5,0)_FB_C(32,11)           -> PC(5,0)_FB_C(32,11)  LOS
PG(6,0)_FB_B(38,11)           -> PC(6,0)_FB_B(38,11)  LOS
PG(7,0)_FB_A(44,11)           -> PC(7,0)_FB_A(44,11)  LOS

```

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Displaying the Status of Control Board Ethernet Switch Ports in a Routing Matrix with a TX Matrix Plus Router

The following sections describe the Ethernet switches on the Control Boards in a routing matrix with a TX Matrix Plus router and how you can use the `show chassis ethernet-switch` operational command to display information about the ports on Control Board Ethernet switches:

- Ethernet Switches on Control Boards in the Routing Matrix on page 107
- Status of Ports on Control Board Ethernet Switches on page 109
- Error Statistics for Ports on Control Board Ethernet Switches on page 109
- Traffic Statistics for Ports on Control Board Ethernet Switches on page 110

Ethernet Switches on Control Boards in the Routing Matrix

In a routing matrix with a TX Matrix Plus router, the SFC and all connected LCCs contain redundant host subsystems. For each host subsystem in a T1600 router in a routing matrix, the Control Board (LCC-CB) contains a Gigabit Ethernet switch. For each host subsystem in a TX Matrix Plus router, the Control Board (TXP-CB) contains a 10-Gigabit Ethernet switch and a Gigabit Ethernet switch, which are connected.

These switches support the following connectivity between Routing Engines in the routing matrix:

- Connectivity Between SFC and LCC Routing Engines on page 107
- Connectivity Between SFC Master and Backup Routing Engines on page 108
- Connectivity Between LCC Master and Backup Routing Engines on page 108

Connectivity Between SFC and LCC Routing Engines

In a routing matrix with a TX Matrix Plus router, the SFC master Routing Engine and every LCC master Routing Engine are connected, enabling the multi-chassis system to operate as a single routing system. The SFC backup Routing Engine and every LCC backup Routing Engine are likewise connected.

This connectivity entails the 10-Gigabit and Gigabit Ethernet switches on the TXP-CB and the Gigabit Ethernet switch on the LCC-CB:

1. The 10-Gigabit Ethernet port on the SFC Routing Engine (automatically configured at the **ixgbe0** internal Ethernet interface) connects to the 10-Gigabit Ethernet switch on the *local Control Board*.
2. On the SFC local Control Board, the 10-Gigabit Ethernet switch connects to the Gigabit Ethernet switch.
3. The Gigabit Ethernet switch on the SFC Control Board connects to the Gigabit Ethernet switch of every LCC Control Board.
4. On every LCC Routing Engine, the Gigabit Ethernet port (automatically configured at the **bcm0** internal Ethernet interface) connects to the Gigabit Ethernet switch on the *local Control Board*.

Connectivity Between SFC Master and Backup Routing Engines

In a routing matrix with a TX Matrix Plus router, the independent control planes of an SFC are connected by two physical links between the two 10-Gigabit Ethernet ports on their respective Routing Engines.

This connectivity entails the 10-Gigabit Ethernet switches on the local and remote TXP-CBs:

- The primary link to the remote Routing Engine is at the **ixgbe0** interface on the SFC Routing Engine. The **ixgbe0** interface—in addition to connecting the SFC Routing Engine to the Routing Engines of the connected LCCs—also connects the SFC Routing Engine to the 10-Gigabit Ethernet port accessed by the **ixgbe1** interface on the *remote Routing Engine*.
- The alternate link to the remote Routing Engine is the 10-Gigabit Ethernet port at the **ixgbe1** interface on the Routing Engine. This second port connects the Routing Engine to the 10-Gigabit Ethernet switch on the *remote Control Board*, which in turn connects to the 10-Gigabit Ethernet port at the **ixgbe0** interface on the *remote Routing Engine*.

If one of the two links between the host subsystems fails, both Routing Engines can use the other link for IP communication.

Connectivity Between LCC Master and Backup Routing Engines

In a routing matrix with a TX Matrix Plus router, the independent control planes of an LCC are connected by two physical links between the two Gigabit Ethernet ports on their respective Routing Engines.

This connectivity entails the Gigabit Ethernet switches on the local and remote LCC-CBs:

- The primary link to the remote Routing Engine is at the **bcm0** interface on the LCC Routing Engine. The **bcm0** interface—in addition to connecting the LCC Routing Engine to the SFC Routing Engine—also connects the LCC Routing Engine to the Gigabit Ethernet port accessed by the **em1** interface on the *remote Routing Engine*.

- The alternate link to the remote Routing Engine is at the Gigabit Ethernet port at the **em1** interface on the local Routing Engine. This second port connects the local Routing Engine to the Gigabit Ethernet switch on the *remote Control Board*, which in turn connects to the Gigabit Ethernet port at the **bcm0** interface on the *remote Routing Engine*.

If one of the two links between the host subsystems fails, both Routing Engines can use the other link for IP communication.

Status of Ports on Control Board Ethernet Switches

To display port status information for the ports on the Control Board Ethernet switches in the routing matrix, use the following form of the **show chassis ethernet-switch** operational command:

```
show chassis ethernet-switch <sfc sfc-number | lcc lcc-number>
```

You can control the scope of the command output as follows:

- To display the status of the switch ports on the Control Boards in every router in the routing matrix, issue the **show chassis ethernet-switch** command without any command options.
- To limit the output to the status of the ports on the 10-Gigabit Ethernet switch and the Gigabit Ethernet switch on the two Control Boards in the SFC, issue the **show chassis ethernet-switch** command with the **sfc sfc-number** option, where *sfc-number* is 0.
- To limit the output to the status of the ports on the Gigabit Ethernet switch in the two Control Boards in a specific LCC, issue the **show chassis ethernet-switch** operational command with the **lcc lcc-number** option, where *lcc-number* is a value from 0 through 3.

Error Statistics for Ports on Control Board Ethernet Switches

To display the numbers and types of errors accumulated on the connected ports of Control Board Ethernet switches in the routing matrix, use the following form of the **show chassis ethernet-switch** operational command:

```
show chassis ethernet-switch errors <sfc sfc-number | lcc lcc-number>
<switch switch-type-number> <switch-port-number>
```

You can control the scope of the command output as follows:

- To display error information for ports on Control Board Ethernet switches in every router in the routing matrix, issue the **show chassis ethernet-switch errors** command (applying only the **errors** command option).
- To limit the command output, you add any combination of the following command options:
 - Chassis—To filter the output on a specific chassis, you can include either the **sfc sfc-number** option (where *sfc-number* is 0) or the **lcc lcc-number** option (where *lcc-number* is a value from 0 through 3).

- SFC Control Board Ethernet Switch Type—If the command output includes information for Control Board Ethernet switches in an SFC (that is, if you applied either the **errors** option alone or the **errors sfc sfc-number** option), you can filter the SFC portion of the output on a specific switch type. To limit SFC Control Board Ethernet switch port error information, include the **switch switch-type-number** option. The **switch-type-number** value can be 0 (the 10-Gigabit Ethernet switch type) or 1 (the Gigabit Ethernet switch type). The **switch-type-number 2** is reserved for future use.
- Control Board Ethernet Switch Port Number—To filter the output on a specific switch port number, include the **switch-port-number** option, where **switch-port-number** is a value from 0 through 27.

Traffic Statistics for Ports on Control Board Ethernet Switches

To display the traffic statistics accumulated on the connected ports of Control Board Ethernet switches in the routing matrix, use the following form of the **show chassis ethernet-switch** operational command:

```
show chassis ethernet-switch statistics <sfc sfc-number | lcc lcc-number>
<switch switch-type-number> <switch-port-number>
```

You can control the scope of the command output as follows:

- To display traffic statistics for Ethernet ports on Control Board switches in every router in the routing matrix, issue the **show chassis ethernet-switch statistics** command (applying only the **statistics** command option).
- To limit the command output, you can add any combination of the following command options:
 - Chassis—To filter the output on a specific chassis, you can include either the **sfc sfc-number** option (where **sfc-number** is 0) or the **lcc lcc-number** option (where **lcc-number** is a value from 0 through 3).
 - SFC Control Board Ethernet Switch Type—If the command output includes information for Control Board Ethernet switches in an SFC (that is, if you applied either the **statistics** option alone or the **statistics sfc sfc-number** option), you can filter the SFC portion of the output on a specific switch type. To limit SFC Control Board Ethernet switch port error information, include the **switch switch-type-number** option. The **switch-type-number** value can be 0 (the 10-Gigabit Ethernet switch type) or 1 (the Gigabit Ethernet switch type). The **switch-type-number 2** is reserved for future use.
 - Control Board Ethernet Switch Port Number—To filter the output on a specific switch port number, include the **switch-port-number** option, where **switch-port-number** is a value from 0 through 27.

Related Topics

- Routing Matrix with a TX Matrix Plus Router Solutions Page
- Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
- Roadmap for Configuring the Routing Matrix on page 15

- Example Configuration for the Routing Matrix on page 33
- Upgrading the JUNOS Software on the Routing Matrix on page 61

Enabling and Disabling Hardware Components of a Routing Matrix with a TX Matrix Plus Router

You can temporarily disable certain hardware components (such as FPCs, PICs, and SIBs) that belong to the TX Matrix Plus router and T1600 routers in the routing matrix. To do so, issue the appropriate `request chassis` command and include the `sfc` or `lcc` option as needed.



NOTE: If you issue a chassis-related command that references FPCs, we recommend that you use the FPC hardware slot number (0 through 7) of the specific T1600 router and specify its corresponding LCC number.

```

user@host> request chassis ?
Possible completions:
cb          Change Control Board status
cip         Change Connector Interface Panel status
fpc         Change Flexible PIC Concentrator status
fpm         Change craft interface status
lcc         Change LCC status
pic         Change Physical Interface Card status
routing-engine Change Routing Engine status
scg         Change SONET Clock Generator status
sib         Change Switch Interface Board status
spmb        Change Switch Processor Mezzanine Board status
user@host> request chassis cb ?
Possible completions:
lcc         Change Control Board status (0..3)
offline     Take CB offline
online      Bring CB online
sfc         Change Control Board status (0..0)
slot        CB slot number (0..1)
user@host> request chassis cip ?
Possible completions:
offline     Take CIP offline
online      Bring CIP online
slot        CIP slot number (0..1)
user@host> request chassis fpc ?
Possible completions:
lcc         Change Flexible PIC Concentrator status of specific LCC
offline     Take FPC offline
online      Bring FPC online
restart     Restart FPC
slot        FPC slot number (0..31)
user@host> request chassis fpm ?
Possible completions:
lcc         Change craft interface status (0..3)
resync      Resynchronize craft interface
sfc         Change craft interface status (0..0)
user@host> request chassis lcc ?

```

```

Possible completions:
  offline          Take LCC offline
  online           Bring LCC online
  slot            LCC Slot (0..3)
user@host> request chassis pic ?
Possible completions:
  fpc-slot        Slot number of FPC that houses PIC (0..31)
  lcc             Change Physical Interface Card status of specific LCC
  offline         Take PIC offline
  online          Bring PIC online
  pic-slot        PIC slot number (0..3)
user@host> request chassis routing-engine ?
Possible completions:
  master          Set Routing Engine mastership
user@host> request chassis scg ?
Possible completions:
  lcc             Change SONET Clock Generator status (0..3)
  offline         Take SCG offline
  online          Bring SCG online
  slot            SCG slot number (0..1)
user@host> request chassis sib ?
Possible completions:
  all-lcc         Change Switch Interface Board status
  f13             Change SIB F13 status
  f2s            Change SIB F2S status
  lcc            Change Switch Interface Board status (0..3)
  offline         Take SIB offline
  online          Bring SIB online
  slot            SIB slot number (0..15)
user@host> request chassis spmb ?
Possible completions:
  lcc            Control operation of an SPMB (0..3)
  restart        Restart SPMB
  sfc            Control operation of an SPMB (0..0)
  slot            SPMB slot number (0..1)

```

The routing matrix extends the concept of taking specific hardware components offline or online to include an entire T1600 router in a routing matrix. To enable or disable a T1600 router in a routing matrix, issue the **request chassis lcc slot lcc-number** (offline | online) command:

```

user@host> request chassis lcc ?
Possible completions:
  offline          Take LCC offline
  online           Bring LCC online
  slot            LCC Slot (0..3)

```

Although you can enter the routing matrix-based slot number when you issue the **request chassis fpc** command, output from **show chassis** commands always references the FPC hardware slot number (0 through 7) of the specific T1600 router and its corresponding LCC number. As a result, we recommend that you include the FPC hardware slot number when you issue **request chassis** or **show chassis** commands, as shown in the following example.

First, issue the **request chassis fpc** command with the routing matrix-based FPC slot number of 19:

```
user@host> request chassis fpc offline slot 19
lcc2-re0:
```

Offline initiated, use "show chassis fpc" to verify

However, when you issue the **show chassis fpc** command to check the result, the output displays the change using node-centric terminology: FPC slot number 3 on T1600 router LCC2 (the equivalent of routing matrix slot 19).

```
user@host> show chassis fpc
lcc0-re0:
```

Slot	State	Temp (C)	CPU Total	Utilization (%) Interrupt	Memory DRAM (MB)	Utilization (%) Heap	Utilization (%) Buffer
0	Empty						
1	Online	31	2	0	256	7	44
2	Online	28	1	0	256	7	44
3	Online	31	2	0	256	14	44
4	Empty						
5	Empty						
6	Empty						
7	Empty						

```
lcc2-re0:
```

Slot	State	Temp (C)	CPU Total	Utilization (%) Interrupt	Memory DRAM (MB)	Utilization (%) Heap	Utilization (%) Buffer
0	Online	31	2	0	256	14	44
1	Online	30	2	0	256	7	44
2	Empty						
3	Offline	--- Offlined by cli command ---					
4	Empty						
5	Empty						
6	Empty						
7	Empty						

To bring the same FPC back online, use the slot number and LCC number from the previous command output:

```
user@host> request chassis fpc online lcc 2 slot 3
lcc2-re0:
```

Online initiated, use "show chassis fpc" to verify

Once you bring the FPC back online, reissue the **show chassis fpc** command to see that the FPC slot and LCC number you used in the last command now matches the command output:

```
user@host> show chassis fpc
lcc0-re0:
```

Slot	State	Temp (C)	CPU Total	Utilization (%) Interrupt	Memory DRAM (MB)	Utilization (%) Heap	Utilization (%) Buffer
0	Empty						
1	Online	31	1	0	256	7	44
2	Online	28	1	0	256	7	44
3	Online	31	3	0	256	14	44

```

4 Empty
5 Empty
6 Empty
7 Empty

```

lcc2-re0:

Slot	State	Temp	CPU Utilization (%)		Memory	Utilization (%)	
		(C)	Total	Interrupt	DRAM (MB)	Heap	Buffer
0	Online	31	3	0	256	14	44
1	Online	30	1	0	256	7	44
2	Empty						
3	Present	0	0	0	0	0	0
4	Empty						
5	Empty						
6	Empty						
7	Empty						

For more information about converting FPC hardware slot numbers on a T1600 router to the global FPC numbers used in a routing matrix and vice versa, see “Global FPC Numbering for Interfaces in a Routing Matrix with a TX Matrix Plus Router” on page 5 and “Displaying Chassis Physical Locations for a Routing Matrix with a TX Matrix Plus Router” on page 103.

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Rebooting and Halting Hardware Components of the Routing Matrix with a TX Matrix Plus Router

In a routing matrix with a TX Matrix Plus router, you can control which hardware component is rebooted or halted. If you reboot or halt the TX Matrix Plus router, by default you also reboot or halt the master Routing Engines on all T1600 routers. To reboot a specific component, issue the `request system reboot` command with the `all-lcc`, `lcc`, or `sfc` option.

```

user@host> request system reboot ?
Possible completions:
<[Enter]>      Execute this command
all-chassis    Reboot all chassis
all-lcc        Reboot all LCC chassis
at            Time at which to perform the operation
in            Number of minutes to delay before operation
lcc           Reboot specific LCC (0..3)
media         Boot media for next boot
message       Message to display to all users
other-routing-engine Reboot the other Routing Engine
sfc           Reboot SFC (0..0)
|            Pipe through a command
user@host> request system reboot

```

```
Reboot the system ? [yes,no] (no) yes
Rebooting lcc0-re0
Rebooting lcc1-re0
```

Similarly, to halt a specific component in a routing matrix, issue the **request system halt** command with the **all-lcc**, **lcc**, or **sfc** option.



CAUTION: Before entering this command, you must have access to the TX Matrix Plus console port and the console ports of all of the LCCs in order to bring up the Routing Engines.

```
user@host> request system halt ?
Possible completions:
<[Enter]>      Execute this command
all-chassis    Halt all chassis
all-lcc        Halt all LCC chassis
at             Time at which to perform the operation
both-routing-engines  Halt both Routing Engines
in            Number of minutes to delay before operation
lcc           Halt specific LCC (0..3)
media         Boot media for next boot
message       Message to display to all users
other-routing-engine  Halt other Routing Engine
sfc           Halt SFC (0..0)
|            Pipe through a command
```

Issuing the **request system halt both-routing-engines** command on a TX Matrix Plus router halts both Routing Engines in the TX Matrix Plus router and both Routing Engines in all T1600 routers in the routing matrix. To reboot a Routing Engine that has been halted, you must connect through the console. For more information about system commands, see the *JUNOS System Basics and Services Command Reference*.

- Related Topics**
- Routing Matrix with a TX Matrix Plus Router Solutions Page
 - Overview of a Routing Matrix with a TX Matrix Plus Router on page 3
 - Roadmap for Configuring the Routing Matrix on page 15
 - Example Configuration for the Routing Matrix on page 33
 - Upgrading the JUNOS Software on the Routing Matrix on page 61

Part 4

Index

- Index on page 119

Index

B

bcm0 interface
T1600 router in a routing matrix.....7, 8, 107

C

configuration groups
 applying inheritance.....20
 configuring19
 disabling inheritance.....21
 displaying inheritance.....20
 for Routing Engines in a routing matrix
 with a TX Matrix Plus router.....21
 overview.....18
configuring a routing matrix with a TX Matrix Plus
router
 configuration roadmap.....15
 configuration tasks
 1. Connecting to the routing matrix.....15
 2. Using configuration groups and
 inheritance.....15
 3. Using global FPC numbers in interface
 names.....15
 4. Configuring protocols and other
 features.....15
 5. Configuring chassis-specific features.....15
 6. Committing the configuration.....15
connecting to a Routing Engine
 routing matrix with a TX Matrix Plus router.....16
Control Board Ethernet switch ports
 routing matrix with a TX Matrix Plus router
 displaying switch port error counts.....107
 displaying switch port status.....107
 displaying switch port traffic statistics.....107

E

em0 interface
 T1600 router in a routing matrix.....7, 8, 16
 TX Matrix Plus router.....7, 8, 16
em1 interface
 T1600 router in a routing matrix.....7, 8, 107

example configurations

 merging a full example.....31
 merging a snippet.....31
 routing matrix with a TX Matrix Plus router.....31

F

fabric topology
 routing matrix with a TX Matrix Plus router
 displaying status information.....104
FPC numbering, global
 routing matrix with a TX Matrix Plus router.....5
full example configuration
 merging.....31

G

global FPC numbering
 routing matrix with a TX Matrix Plus router.....5
graceful Routing Engine switchover *See* GRES
GRES
 routing matrix with a TX Matrix Plus router.....8

I

inheritance of configuration statements.....18
interface names
 routing matrix with a TX Matrix Plus router
 using global FPC numbering.....5
internal Ethernet interfaces
 T1600 router in a routing matrix.....7, 8, 107
 TX Matrix Plus router.....7, 8, 107
ixgbe0 interface
 TX Matrix Plus router.....7, 8, 107
ixgbe1 interface
 TX Matrix Plus router.....7, 8, 107

L

LCC-CB in a T1600 router in a routing matrix
 host subsystem Control Board
 Ethernet switch ports.....107

M

- management Ethernet interface
 - T1600 router in a routing matrix.....7, 8, 16
 - TX Matrix Plus router.....7, 8, 16

N

- nonstop active routing *See* NSR
- NSR
 - T1600 router in a routing matrix.....3

R

- redundant host subsystems
 - best practice for managing Routing Engines.....16
 - routing matrix with a TX Matrix Plus router.....8
- redundant Routing Engines *See* redundant host subsystems
- Routing Engines
 - routing matrix with a TX Matrix Plus router
 - inheritance of configuration statements.....21
 - internal Ethernet interfaces.....7, 8, 107
 - management Ethernet interfaces.....7, 8, 16
 - serial interface.....16
- Routing Engines,
 - TX Matrix Plus router
 - connecting to the master.....16
- routing matrix with a TX Matrix Plus router
 - configuration example.....33
 - configuration roadmap.....15
 - configuration tasks
 1. Connecting to the routing matrix.....15
 2. Using configuration groups and inheritance.....15
 3. Using global FPC numbers in interface names.....15
 4. Configuring protocols and other features.....15
 5. Configuring chassis-specific features.....15
 6. Committing the configuration.....15
 - configuring optional chassis-specific features
 - alarm if an LCC does not come online.....26
 - LCC to stay offline.....26
 - PIC-specific features.....25
 - displaying hardware information
 - chassis physical locations.....103
 - Control Board Ethernet switch ports.....107
 - FPC global and local numbering.....103
 - interface name for a local FPC number.....103
 - local FPC number for an interface
 - name.....103
 - status of SIB connections.....104
 - FPC numbering in interface names.....5
 - GRES.....8
 - inheritance of configuration statements
 - Routing Engines.....21

- NSR.....3
- overview.....3
- Routing Engine configuration groups.....21
- system requirements.....11
- upgrading the software
 - problem: insufficient available DRAM.....91
 - problem: insufficient free disk space.....85
 - problem: mixed software versions.....91
 - procedure.....61
 - verifying the configuration.....40

S

- snippet of an example configuration
 - merging.....31
- system requirements
 - routing matrix with a TX Matrix Plus router.....11

T

- T1600 router in a routing matrix
 - configuring optional chassis-specific features
 - alarm if LCC does not come online.....26
 - LCC to stay offline.....26
 - PIC-specific features.....25
 - internal Ethernet interfaces.....7, 8, 107
 - management Ethernet interface.....7, 8, 16
 - NSR.....3
 - replacing fxp0 with em0.....7
 - system requirements.....11
- TX Matrix Plus router
 - internal Ethernet interfaces.....7, 8, 107
 - management Ethernet interface.....7, 8, 16
- TXP-CB in a TX Matrix Plus router
 - host subsystem Control Board
 - Ethernet switch ports.....107
- TXP-F13 SIB (TX Matrix Plus router)
 - displaying the status of connections to TXP-T1600
 - SIBs in an LCC.....104
- TXP-T1600 SIB (T1600 router in a routing matrix)
 - displaying the status of connections to TXP-F13
 - SIBs on the SFC.....104

U

- upgrading the software
 - routing matrix with a TX Matrix Plus router
 - problem: insufficient available DRAM.....91
 - problem: insufficient free disk space.....85
 - problem: mixed software versions.....91
 - procedure.....61