



Junos[®] OS

Installation and Upgrade Guide



Modified: 2017-08-11

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. and/or its affiliates in the United States and other countries. All other trademarks may be property of their respective owners.

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

Junos® OS Installation and Upgrade Guide

Copyright © 2017 Juniper Networks, Inc. All rights reserved.

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

	About the Documentation	xxi
	Documentation and Release Notes	xxi
	Supported Platforms	xxi
	Using the Examples in This Manual	xxii
	Merging a Full Example	xxii
	Merging a Snippet	xxiii
	Documentation Conventions	xxiii
	Documentation Feedback	xxv
	Requesting Technical Support	xxvi
	Self-Help Online Tools and Resources	xxvi
	Opening a Case with JTAC	xxvi
Part 1	Junos Software and Hardware Overview	
Chapter 1	Software Overview	3
	Junos OS Overview	3
	One Operating System	4
	One Modular Software Architecture	4
	Junos OS Editions	5
	FIPS 140-2 Security Compliance	5
	Junos OS Installation Packages Prefixes	6
	Software Naming Convention	7
	Software Naming Convention for SRX Series Devices	8
	Software Package Information Security	9
	Junos OS Release Numbers	10
	Installation Media	11
	Installation Bundles	12
	Installation Modules	12
	Configuration Files	13
	Configuration File Selection Sequence	13
	Remote Storage of Configuration Files	15
	Understanding Software Infrastructure and Processes on EX Series	
	Switches	15
	Routing Engine and Packet Forwarding Engine	15
	Junos OS Processes	16
	Understanding Junos OS with Upgraded FreeBSD	17
	Understanding Changes in Operations and Management for Junos OS with	
	Upgraded FreeBSD	19
	Understanding Junos OS with Upgraded FreeBSD Package Names	21
	Understanding Junos OS with Upgraded FreeBSD Snapshots	23
	Understanding Junos OS with Upgraded FreeBSD Disk Volumes	25

Chapter 2	Hardware Overview	27
	Hardware Architecture Overview	27
	Hardware Overview (ACX Series, M Series, MX Series, T Series, and TX Matrix Routers)	28
	System Memory	29
	Storage Media	30
	Hardware Overview of SRX Series Services Gateways	31
	SRX Series Device Overview	31
	System Memory	32
	Storage Media	32
	Routing Engines and Storage Media Names (ACX Series, M Series, MX Series, PTX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers)	33
	Storage Media Names for SRX Series Devices	36
	Boot Sequence on M Series, MX Series, T Series, TX Matrix, TX Matrix Plus, ACX Series, and PTX Series Devices with Routing Engines	36
	Boot Sequence on SRX Series Devices	38
Part 2	Installing Junos Software	
Chapter 3	Installation Overview	41
	Installation Type Overview	41
	Standard Installation	41
	Category Change Installation	41
	Recovery Installation	42
	Installation Categories on the ACX Series, M Series, MX Series, T Series, TX Matrix, and TX Matrix Plus Routers	42
	Installation Categories on SRX Series Devices	43
	Understanding Software Installation on EX Series Switches	44
	Overview of the Software Installation Process	44
	Software Package Security	44
	Installing Software on a Virtual Chassis	45
	Installing Software on Switches with Redundant Routing Engines	45
	Installing Software Using Automatic Software Download	45
	Autoinstalling a Configuration File on an EX2200 or EX3300 Switch from a Disk-on-Key USB Memory Stick	46
	Installing Software on an EX2300 or EX3400 Switch	46
	Troubleshooting Software Installation	46
Chapter 4	Performing a Standard or Change Category Installation	47
	Checking the Current Configuration and Candidate Software Compatibility	47
	Determining the Junos OS Version	48
	Downloading Software	48
	Downloading Software with a Browser	49
	Downloading Software Using the Command-Line Interface	49
	Downloading Software Packages from Juniper Networks	51
	Understanding Download Manager for SRX Series Devices	51
	Overview	52
	Using Download Manager to Upgrade Junos OS	52
	Handling Errors	53

	Considerations	53
	Understanding the Console Port	54
	Accessing a Junos OS Device the First Time	55
	Backing Up the Existing Installation on Routers	56
	Backing Up the Current Installation on SRX Series Devices	57
	Backing Up the Current Installation on SRX5800, SRX5600, and SRX5400 Devices	57
	Backing Up the Current Installation on SRX300, SRX320, SRX340, SRX345, and SRX550M Devices	58
	Configuring External CompactFlash for SRX650 Devices	58
	Installing the Software Package on a Router with a Single Routing Engine	59
	Installing the Software Package on a Router with Redundant Routing Engines	60
	Preparing the Router for the Installation	61
	Installing Software on the Backup Routing Engine	62
	Installing Software on the Remaining Routing Engine	63
	Finalizing the Installation	65
	Repartitioning Routing Engine System Storage to Increase the Swap Partition	66
	Installing Software on EX Series Switches (J-Web Procedure)	66
	Installing Software Upgrades from a Server	67
	Installing Software Upgrades by Uploading Files	68
	Registering the EX Series Switch with the J-Web Interface	68
	Connecting and Configuring an EX Series Switch (CLI Procedure)	69
	Connecting and Configuring an EX9200 Switch (CLI Procedure)	72
	Preparing the USB Flash Drive to Upgrade Junos OS on SRX Series Devices	75
	Installing Junos OS on SRX Series Devices Using a USB Flash Drive	77
	Upgrading the Boot Loader on SRX Series Devices	78
	Installing Junos OS on SRX Series Devices from the Boot Loader Using a TFTP Server	79
	Installing Junos OS on SRX Series Devices from the Boot Loader Using a USB Storage Device	82
Chapter 5	Configuring Zero Touch Provisioning	85
	Understanding Zero Touch Provisioning	85
	Configuring Zero Touch Provisioning	89
Chapter 6	Configuring Automatic Installation of Configuration Files	95
	Autoinstallation Overview	95
	Automatic Installation of Configuration Files	96
	Supported Autoinstallation Interfaces and Protocols	96
	Typical Autoinstallation Process on a New Device	97
	Example: Configuring Autoinstallation on SRX Series Devices	100
	Configuring Autoinstallation on JNU Satellite Devices	103
	Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group	105
	Supported Autoinstallation Interfaces and Protocols	105
	Typical Autoinstallation Process on a New Router	106
	Autoinstallation of Satellite Devices in a Junos Node Unifier Group	107
	Verifying Autoinstallation on JNU Satellite Devices	108

Chapter 7	Configuring Dual-Root Partitions for High Availability	111
	Understanding Resilient Dual-Root Partitions on Switches	111
	Resilient Dual-Root Partition Scheme (Junos OS Release 10.4R3 and Later)	112
	Automatic Fixing of Corrupted Primary Root Partition with the Automatic Snapshot Feature	112
	Earlier Partition Scheme (Junos OS Release 10.4R2 and Earlier)	113
	Understanding Upgrading or Downgrading Between Resilient Dual-Root Partition Releases and Earlier Releases	114
	Dual-Root Partitioning Scheme on SRX Series Devices	114
	Boot Media and Boot Partition on SRX Series Devices	115
	Important Features of the Dual-Root Partitioning Scheme	116
	Understanding Automatic Recovery of the Primary Junos OS Image with Dual-Root Partitioning	117
	Understanding How the Primary Junos OS Image with Dual-Root Partitioning Recovers Devices	118
	Understanding How Junos OS Release 10.0 or Later Upgrades with Dual-Root Partitioning	120
	Example: Installing Junos OS on SRX Series Devices Using the Partition Option	120
	Reinstalling the Single-Root Partition on SRX Series Devices	124
Chapter 8	Upgrading Software	127
	Upgrading Software Packages	128
	Overview of Upgrading to 64-bit Junos OS	131
	Upgrading Redundant Routing Engines from 32-bit to 64-bit Junos OS	132
	Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using One Slot	134
	Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using Two Slots	134
	Upgrading Routers Using Unified ISSU	136
	Understanding Nonstop Software Upgrade on EX Series Switches	136
	Requirements for Performing an NSSU	138
	How an NSSU Works	139
	EX3300, EX3400, EX4200, EX4300, EX4500, and Mixed Virtual Chassis	139
	EX6200 and EX8200 Switches	140
	EX8200 Virtual Chassis	141
	NSSU Limitations	142
	NSSU and Junos OS Release Support	142
	Overview of NSSU Configuration and Operation	143
	Upgrading Software by Using Automatic Software Download	144
	Verifying That Automatic Software Download Is Working Correctly	145
	Verifying Junos OS and Boot Loader Software Versions on an EX Series Switch	146
	Verifying the Number of Partitions and File System Mountings	146
	Verifying the Loader Software Version	147
	Verifying Which Root Partition Is Active	148

Verifying the Junos OS Version in Each Root Partition	148
Upgrading the Loader Software on the Line Cards in a Standalone EX8200 Switch or an EX8200 Virtual Chassis	149
Upgrading Junos OS with Upgraded FreeBSD	153
Determining Which Package or Packages to Install	154
Installing Junos OS with Upgraded FreeBSD Over Plain Junos OS	156
Installing Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Different Release	158
To Install Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Later Release	159
Understanding Junos OS Upgrades for SRX Series Devices	160
Understanding Junos OS Upgrades	160
Junos OS Upgrade Methods on the SRX Series Devices	160
Preparing Your SRX Series Device for Junos OS Upgrades	162
Secondary Storage Devices Available on SRX Series Devices	162
Verifying Available Disk Space on SRX Series Devices	163
Cleaning Up the System File Storage Space	164
Understanding Junos OS with Upgraded FreeBSD for SRX5400, SRX5600, and SRX5800 Devices	165
Understanding Changes in Operations and Management for Junos OS with Upgraded FreeBSD for SRX5400, SRX5600, and SRX5800 devices	166
Understanding Junos OS with Upgraded FreeBSD Package Names	168
Upgrading Junos OS with Upgraded FreeBSD	169
Installing Junos OS with Upgraded FreeBSD Over Plain Junos OS	170
Understanding Junos OS with Upgraded FreeBSD Snapshots	175
Understanding Junos OS with Upgraded FreeBSD Disk Volumes for SRX5400, SRX5600, and SRX5800 devices	176
Downloading Software Packages from Juniper Networks	177
Example: Installing Junos OS Upgrade Packages on SRX Series Devices	178
Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server	181
Understanding BIOS Upgrades on SRX Series Devices	182
Understanding Manual BIOS Upgrade Using the Junos CLI	182
Understanding Auto BIOS Upgrade Methods on SRX Series Devices	183
Disabling Auto BIOS Upgrade on SRX Series Devices	184
Reverting the Junos OS Software Image Back to the Previous Version	184
Overview of CoS Upgrade Requirements (Junos OS Release 11.1 or 11.2 to a Later Release)	187
Before You Begin Installing or Upgrading the Firmware	189
Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC (PTX-5-100G-WDM)	190
Upgrading Firmware on the 5-Port 100-Gigabit DWDM OTN PIC (PTX-5-100G-WDM)	191
Installing Firmware on the 100-Gigabit DWDM OTN MIC (MIC3-100G-DWDM)	193
Upgrading Firmware on the 100-Gigabit DWDM OTN MIC (MIC3-100G-DWDM)	194

Chapter 9	Booting a Device Using a System Snapshot	197
	Understanding System Snapshot on EX Series Switches	197
	Creating a Snapshot and Using It to Boot an EX Series Switch	198
	Creating a Snapshot on a USB Flash Drive and Using It to Boot the Switch	198
	Verifying That a System Snapshot Was Created on an EX Series Switch	199
	Booting an EX Series Switch Using a Software Package Stored on a USB Flash Drive	200
	Example: Creating a Snapshot and Using It to Boot an SRX Series Device	201
Chapter 10	Performing a Recovery Installation	205
	Creating an Emergency Boot Device	205
	Understanding Integrity Check and Autorecovery of Configuration, Licenses, and Disk Information on SRX Series Devices	207
	Overview	207
	How Autorecovery Works	207
	How to Use Autorecovery	208
	Data That Is Backed Up in an Autorecovery	208
	Troubleshooting Alarms	208
	Considerations	209
	Creating an Emergency Boot Device for RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX Routing Engines	209
	Performing a Recovery Installation	211
	Creating a New Configuration on a Single Routing Engine	212
	Logging In to the Router Console	212
	Configuring Administration User Accounts	213
	Adding the Management Console to the Network	213
	Committing Changes	215
	Creating a New Configuration with Redundant Routing Engines	217
	Configuring Administration User Accounts	217
	Setting Up Routing Engine Configuration Groups	218
	Completing the Management Console Configuration	220
	Committing and Synchronizing Changes	220
	Saving a Rescue Configuration File	222
	Restoring a Saved Configuration	223
	Copy Saved Files to the Router	223
	Loading and Committing the Configuration File	224
	Reverting to the Default Factory Configuration by Using the request system zeroize Command	225
	Reverting to the Rescue Configuration	225
Chapter 11	Reinstalling Software	227
	Checklist for Reinstalling Junos OS	227
	Log the Software Version Information	229
	Log the Hardware Version Information	230
	Log the Chassis Environment Information	231
	Log the System Boot-Message Information	231
	Log the Active Configuration	233
	Log the Interfaces on the Router	233

	Log the BGP, IS-IS, and OSPF Adjacency Information	234
	Log the System Storage Information	235
	Back Up the Currently Running and Active File System	236
	Reinstall Junos OS	237
	Reconfigure Junos OS	237
	Configure Host Names, Domain Names, and IP Addresses	237
	Protecting Network Security by Configuring the Root Password	239
	Check Network Connectivity	240
	Copy Backup Configurations to the Router	241
	Configure Host Names, Domain Names, and IP Addresses	241
	Protecting Network Security by Configuring the Root Password	243
	Check Network Connectivity	245
	Copy Backup Configurations to the Router	245
	After You Reinstall Junos OS	245
	Compare Information Logged Before and After the Reinstall	245
	Back Up the New Software	246
	Compare Information Logged Before and After the Reinstall	246
	Back Up the New Software	247
Chapter 12	Downgrading Software	249
	Downgrading Junos OS from Upgraded FreeBSD	249
	Downgrading from Junos OS with Upgraded FreeBSD to Junos OS	249
	Installing Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Different Release	250
Chapter 13	Rebooting or Halting Software Processes on a Device	253
	Restarting and Halting SRX Series Devices	253
	Rebooting SRX Series Devices	253
	Halting SRX Series Devices	255
	Bringing Chassis Components Online and Offline on SRX Series Devices	257
	Restarting the Chassis on SRX Series Devices	258
	Rebooting or Halting the EX Series Switch (J-Web Procedure)	258
Chapter 14	RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Overview	261
	RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX with VM Host Support	261
	Architecture of RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines	262
	Salient Features of the RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines	263
	Platform Virtualization	263
	Hardware Assisted Paravirtualized Guest Junos OS	264
	Guest Junos OS to Serve as the Administrative Framework	264
	Storage Partitioning and Redundancy	264
	NTP and Time Zone	265
	Autorecovery	265
	Handling Reboot and Power Off	266
	Boot Process Overview	267
	Booting for the First Time	267
	Boot Sequence	267
	Understanding Console Port	267
	Understanding Hostnames Synchronization	268

VM Host Installation	268
VM Host Upgrade	269
VM Host Rollback	270
VM Host Snapshot	272
Disk Recovery Using the VM Host Snapshot	272
Copying VM Host Installation Package to the PXE Boot Server	272
Upgrading the SSD Firmware on RE-S-X6-64G and RE-MX2K-X8-64G Routing Engines	275
Disabling Autorecovery on RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines	277
VM Host Operations and Management	277

Part 3

Chapter 15

Installing and Managing Software Licenses

Software License Overview	281
Junos OS Feature Licenses	281
License Enforcement	282
Junos OS Feature License Keys	283
Release-Tied License Keys and Upgrade Licenses on MX Series Routers ..	283
Licensable Ports on MX5, MX10, and MX40 Routers	284
Port Activation on MX104 Routers	285
Software Feature Licenses	286
Software Features That Require Licenses on M Series, MX Series, and T Series Routers	287
Software Features That Require Licenses on M Series Routers Only	290
Software Features That Require Licenses on MX Series Routers Only	291
Software Feature Licenses for SRX Series Devices	297
Software Features That Require Licenses on EX Series Switches	297
Software Features That Require Licenses on the QFX Series	298
Software Features That Require Licenses on MX Series Routers Only	301
Understanding Software Licenses for EX Series Switches	307
Purchasing a Software Feature License	307
Features Requiring a License on EX2200 Switches	308
Features Requiring a License on EX2300 Switches	309
Features Requiring a License on EX3300 Switches	309
Features Requiring a License on EX3400 Switches	311
Features Requiring a License on EX4300 Switches	311
Features Requiring a License on EX4600 Switches	313
Features Requiring a License on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches	314
License Warning Messages	315

Chapter 16	Installing and Managing Licenses	317
	Adding New Licenses (CLI Procedure)	317
	Installing a License Using a Configuration Statement	317
	Installing Licenses Using the CLI Directly	318
	Installing Licenses Using a Configuration File	319
	Installing a License Using an Operational Command	321
	Adding a License to a Device with a Single Routing Engine	321
	Adding a License to a Device with Dual Routing Engines	321
	Deleting License Keys (CLI)	322
	Using the Operational Command to Delete Licenses	322
	Using a Configuration Command to Delete Licenses	323
	Saving License Keys (CLI)	324
	Verifying Junos OS License Installation (CLI)	324
	Displaying Installed Licenses	325
	Displaying License Usage	326
Part 4	Troubleshooting Information	
Chapter 17	Troubleshooting Software Installation	329
	Troubleshooting Software Installation	329
	Recovering from a Failed Software Upgrade on an EX Series Switch	329
	Rebooting from the Inactive Partition	330
	Freeing Disk Space for Software Installation	331
	Installation from the Boot Loader Generates 'cannot open package' Error	331
	Troubleshooting a Switch That Has Booted from the Backup Junos OS Image	332
	Disk Space Management for Junos OS Installation	333
	Verifying PIC Combinations	334
Part 5	Configuration Statements and Operational Commands	
Chapter 18	Configuration Statements	339
	auto-configuration	340
	auto-configuration (System)	341
	auto-image-upgrade	343
	auto-snapshot	344
	autoinstallation	345
	autoinstallation (JNU Satellite Devices)	346
	bootp	347
	commit	348
	configuration-servers	349
	delete-after-commit (JNU Satellites)	350
	file (App Engine Virtual Machine Management Service)	351
	flag (App Engine Virtual Machine Management Service)	353
	interfaces (Autoinstallation)	354
	level (App Engine Virtual Machine Management Service)	355
	license	356
	traceoptions (App Engine Virtual Machine Management Service)	358

	traceoptions (System License)	359
	vmhost	361
	vmhost management-if disable	362
	vmhost management-if link-mode	363
	vmhost management-if speed	364
	usb	365
	vmhost	366
Chapter 19	Operational Commands	367
	clear system login lockout	369
	request system autorecovery state	370
	request system download abort	372
	request system download clear	373
	request system download pause	374
	request system download resume	375
	request system download start	376
	request system firmware upgrade	378
	request system halt	380
	request system license add	382
	request system license delete	383
	request system license save	384
	request system license update	385
	request system partition compact-flash	386
	request system power-off	387
	request system reboot	389
	request system reboot (EX Series)	395
	request system reboot (Junos OS with Upgraded FreeBSD)	398
	request system recover	401
	request system scripts add	403
	request system scripts delete	404
	request system scripts rollback	405
	request system snapshot	406
	request system snapshot (Junos OS with Upgraded FreeBSD)	414
	request system snapshot (Maintenance)	416
	request system software abort in-service-upgrade (ICU)	419
	request system software add	420
	request system software add (Maintenance)	432
	request system software configuration-backup	433
	request system software configuration-restore	434
	request system software delete	435
	request system software rollback	439
	request system software rollback (SRX Series)	444
	request system software validate	445
	request system software validate on (Junos OS with Upgraded FreeBSD)	449
	request system storage cleanup	452
	request system storage cleanup (SRX Series)	463
	request system zeroize	466
	request vmhost halt	472
	request vmhost reboot	474

	show chassis usb storage	476
	show system autoinstallation status	477
	show system autorecovery state	479
	show system boot-messages	481
	show system auto-snapshot	488
	show system download	490
	show system license	492
	show system license (View)	501
	show system login lockout	504
	show system snapshot	505
	show system snapshot (Junos OS with Upgraded FreeBSD)	508
	show system snapshot (Upgraded FreeBSD)	510
	show system snapshot media	512
	show system storage partitions (EX Series Switches Only)	513
	show system storage partitions (View SRX Series)	515
	show version	517
Chapter 20	VM Host Software Administrative Commands	519
	request vmhost cleanup	520
	request vmhost file-copy	522
	request vmhost halt	523
	request vmhost hard-disk-test	525
	request vmhost power-off	527
	request vmhost power-on	529
	request vmhost reboot	530
	request vmhost snapshot	532
	request vmhost software abort	534
	request vmhost software add	535
	request vmhost software in-service-upgrade	538
	request vmhost software rollback	540
	request vmhost zeroize	543
Chapter 21	VM Host Software Monitoring Commands	545
	show vmhost bridge	546
	show vmhost crash	548
	show vmhost hard-disk-test	549
	show vmhost hardware	551
	show vmhost information	553
	show vmhost logs	555
	show vmhost management-if	557
	show vmhost netstat	558
	show vmhost processes	560
	show vmhost resource-usage	562
	show vmhost snapshot	565
	show vmhost status	567
	show vmhost uptime	569
	show vmhost version	571

List of Figures

Part 1	Junos Software and Hardware Overview	
Chapter 1	Software Overview	3
	Figure 1: Configuration Selection Sequence	14
Chapter 2	Hardware Overview	27
	Figure 2: Routing Engines	29
	Figure 3: SRX240 Device Front Panel	31
	Figure 4: SRX650 Device System Routing Engine	31
	Figure 5: SRX345 Device Front Panel	31
	Figure 6: SRX1500 Device Front Panel	31
	Figure 7: SRX5800 Device Routing Engine	32
	Figure 8: SRX4200 Services Gateway Front Panel	32
Part 2	Installing Junos Software	
Chapter 4	Performing a Standard or Change Category Installation	47
	Figure 9: Connecting to the Console Port on a Junos OS Device	54
Chapter 8	Upgrading Software	127
	Figure 10: Upgrading to 64-bit Junos OS with Redundant Routing Engines	132
	Figure 11: Upgrading a Single Routing Engine to 64-bit Junos OS Using Two Slots	135
Chapter 14	RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Overview	261
	Figure 12: Architecture of RE-MX-X6, RE-MX-X8 and RE-PTX-X8 Routing Engines	263
	Figure 13: SSD Partitioning	265

List of Tables

	About the Documentation	xxi
	Table 1: Notice Icons	xxiv
	Table 2: Text and Syntax Conventions	xxiv
Part 1	Junos Software and Hardware Overview	
Chapter 1	Software Overview	3
	Table 3: Installation Package Prefixes	6
	Table 4: Junos OS Processes	16
	Table 5: Upgraded FreeBSD Kernel Support by Hardware Platform	18
	Table 6: New and Changed Commands and Statements for Junos OS with Upgraded FreeBSD	19
	Table 7: Deprecated Commands and Statements for Junos OS with Upgraded FreeBSD	20
Chapter 2	Hardware Overview	27
	Table 8: Routing Engines and Storage Media Names (ACX Series, M Series, MX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers)	33
	Table 9: Storage Media Names	36
Part 2	Installing Junos Software	
Chapter 4	Performing a Standard or Change Category Installation	47
	Table 10: show system download Output Fields	53
	Table 11: Install Remote Summary	67
	Table 12: Upload Package Summary	68
	Table 13: Environment Variables Settings	80
Chapter 6	Configuring Automatic Installation of Configuration Files	95
	Table 14: Interfaces and Protocols for IP Address Acquisition During Autoinstallation	96
Chapter 7	Configuring Dual-Root Partitions for High Availability	111
	Table 15: Resilient Dual-Root Partition Scheme	112
	Table 16: Earlier Partition Scheme	114
	Table 17: Storage Media on SRX Series Devices	115
Chapter 8	Upgrading Software	127
	Table 18: Platform and Release Support for NSSU	142
	Table 19: Upgrade Path to Junos OS with the Upgraded FreeBSD	155
	Table 20: Secondary Storage Devices for SRX Series Devices	163
	Table 21: Upgraded FreeBSD Kernel Support by Hardware Platform	165

	Table 22: New and Changed Commands and Statements for Junos OS with Upgraded FreeBSD	167
	Table 23: Deprecated Commands and Statements for Junos OS with Upgraded FreeBSD	167
	Table 24: Upgrade Path to Junos OS with the Upgraded FreeBSD	169
	Table 25: Install Package Summary	181
	Table 26: CLI Commands for Manual BIOS Upgrade	182
Chapter 10	Performing a Recovery Installation	205
	Table 27: Autorecovery Alarms	208
Chapter 11	Reinstalling Software	227
	Table 28: Checklist for Reinstalling Junos OS	227
Chapter 14	RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Overview	261
	Table 29: Hardware Specifications of the RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX, Routing Engines	262
Part 3	Installing and Managing Software Licenses	
Chapter 15	Software License Overview	281
	Table 30: Upgrade Licenses for Enhancing Port Capacity	285
	Table 31: Port Activation License Model for MX104 Routers	286
	Table 32: Junos OS Feature License Model Number for M Series, MX Series, and T Series Routers	287
	Table 33: Junos OS Feature License Model Number for M Series Routers	290
	Table 34: Junos OS Feature License Model Number for MX Series Routers	291
	Table 35: Standard Junos OS Feature Licenses and Model Numbers for QFX Series Devices	299
	Table 36: Junos OS Feature License Model Number for MX Series Routers	301
	Table 37: Junos OS EFL Part Number on EX2200 Switches	308
	Table 38: Junos OS EFL Part Number on EX2300 Switches	309
	Table 39: Junos OS EFL Part Number on EX3300 Switches	310
	Table 40: Junos OS AFL Part Number on EX3300 Switches	310
	Table 41: Junos OS EFL Part Number on EX3400 Switches	311
	Table 42: Junos OS EFL Part Number on EX4300 Switches	312
	Table 43: Junos OS AFL Part Number on EX4300 Switches	312
	Table 44: Junos OS AFL Part Number on EX4600 Switches	313
	Table 45: Junos OS AFL Part Number on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches	314
Part 5	Configuration Statements and Operational Commands	
Chapter 19	Operational Commands	367
	Table 46: request system storage cleanup Output Fields	455
	Table 47: request system storage cleanup Output Fields	463
	Table 48: show system autoinstallation status Output Fields	478
	Table 49: show system autorecovery state Output Fields	479
	Table 50: show system auto-snapshot status Output Fields	488
	Table 51: show system download Output Fields	490
	Table 52: show system license Output Fields	493

Table 53: show system license Output Fields	501
Table 54: show system login lockout	504
Table 55: show system snapshot Output Fields	506
Table 56: show system storage partitions Output Fields	513

About the Documentation

- [Documentation and Release Notes on page xxi](#)
- [Supported Platforms on page xxi](#)
- [Using the Examples in This Manual on page xxii](#)
- [Documentation Conventions on page xxiii](#)
- [Documentation Feedback on page xxv](#)
- [Requesting Technical Support on page xxvi](#)

Documentation and Release Notes

To obtain the most current version of all Juniper Networks[®] technical documentation, see the product documentation page on the Juniper Networks website at <http://www.juniper.net/techpubs/>.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at <http://www.juniper.net/books>.

Supported Platforms

For the features described in this document, the following platforms are supported:

- [ACX Series](#)
- [M Series](#)
- [MX Series](#)
- [T Series](#)
- [EX Series](#)
- [PTX Series](#)
- [QFabric System](#)
- [QFX Series](#)
- [SRX Series](#)

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

For more information about the **load** command, see [CLI Explorer](#).

Documentation Conventions

[Table 1](#) on [page xxiv](#) defines notice icons used in this guide.

Table 1: Notice Icons

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

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

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the configure command: user@host> configure
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> show chassis alarms No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none"> Introduces or emphasizes important new terms. Identifies guide names. Identifies RFC and Internet draft titles. 	<ul style="list-style-type: none"> A policy <i>term</i> is a named structure that defines match conditions and actions. <i>Junos OS CLI User Guide</i> RFC 1997, <i>BGP Communities Attribute</i>
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i>

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

Convention	Description	Examples
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none">To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level.The console port is labeled CONSOLE.
< > (angle brackets)	Encloses optional keywords or variables.	stub <default-metric <i>metric</i>>;
(pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	broadcast multicast (<i>string1</i> <i>string2</i> <i>string3</i>)
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	rsvp { # Required for dynamic MPLS only
[] (square brackets)	Encloses a variable for which you can substitute one or more values.	community name members [<i>community-ids</i>]
Indentation and braces ({ })	Identifies a level in the configuration hierarchy.	<pre>[edit] routing-options { static { route default { nexthop <i>address</i>; retain; } } }</pre>
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		
Bold text like this	Represents graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none">In the Logical Interfaces box, select All Interfaces.To cancel the configuration, click Cancel.
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select Protocols>Ospf .

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page of the Juniper Networks TechLibrary site at <http://www.juniper.net/techpubs/index.html>, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at <http://www.juniper.net/techpubs/feedback/>.

- E-mail—Send your comments to techpubs-comments@juniper.net. Include the document or topic name, URL or page number, and software version (if applicable).

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or Partner Support Service support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

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

Self-Help Online Tools and Resources

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

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>
- Download the latest versions of software and review release notes: <http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <http://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum: <http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

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

Opening a Case with JTAC

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

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

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

PART 1

Junos Software and Hardware Overview

- [Software Overview on page 3](#)
- [Hardware Overview on page 27](#)

CHAPTER 1

Software Overview

- [Junos OS Overview on page 3](#)
- [Junos OS Editions on page 5](#)
- [FIPS 140-2 Security Compliance on page 5](#)
- [Junos OS Installation Packages Prefixes on page 6](#)
- [Software Naming Convention on page 7](#)
- [Software Naming Convention for SRX Series Devices on page 8](#)
- [Software Package Information Security on page 9](#)
- [Junos OS Release Numbers on page 10](#)
- [Installation Media on page 11](#)
- [Installation Bundles on page 12](#)
- [Installation Modules on page 12](#)
- [Configuration Files on page 13](#)
- [Understanding Software Infrastructure and Processes on EX Series Switches on page 15](#)
- [Understanding Junos OS with Upgraded FreeBSD on page 17](#)
- [Understanding Changes in Operations and Management for Junos OS with Upgraded FreeBSD on page 19](#)
- [Understanding Junos OS with Upgraded FreeBSD Package Names on page 21](#)
- [Understanding Junos OS with Upgraded FreeBSD Snapshots on page 23](#)
- [Understanding Junos OS with Upgraded FreeBSD Disk Volumes on page 25](#)

Junos OS Overview

Juniper Networks provides high-performance network devices that create a responsive and trusted environment for accelerating the deployment of services and applications over a single network. The Junos[®] operating system (Junos OS) is the foundation of these high-performance networks.

Starting with Junos OS Release 15.1, certain hardware platforms run Junos OS based on an upgraded FreeBSD kernel instead of older versions of FreeBSD. Basing Junos OS on the newer kernel (referred to as Junos OS with upgraded FreeBSD) provides Junos OS

with sophisticated processing, efficiency, and security features which do not have to be reproduced in Junos OS.

Starting in Junos OS Release 16.1, virtualized Routing Engines are supported that not only provide increased control plane scalability and performance but also provide virtualization capabilities to the Junos OS infrastructure. These virtualized Routing Engines, or VM hosts are the Routing Engines RE-MX-X6, RE-MX-X8, and RE-PTX-X8.



NOTE: VM hosts only run on a virtualized Junos OS with Upgraded FreeBSD.

Unlike other complex, monolithic software architectures, Junos OS incorporates key design and developmental differences to deliver increased network availability, operational efficiency, and flexibility. The key advantages to this approach are:

- [One Operating System on page 4](#)
- [One Modular Software Architecture on page 4](#)

One Operating System

Unlike other network operating systems that share a common name but splinter into many different programs, Junos OS is a single, cohesive operating system that is shared across all network devices and product lines. This allows Juniper Networks engineers to develop software features once and share these features across all product lines simultaneously. Because features are common to a single source, they generally are implemented the same way for all product lines, thus reducing the training required to learn different tools and methods for each product. Because all Juniper Networks products use the same code base, interoperability between products is not an issue.

One Modular Software Architecture

Although individual modules of Junos OS communicate through well-defined interfaces, each module runs in its own protected memory space, preventing one module from disrupting another. This separation enables the independent restart of each module as necessary. This is in contrast to monolithic operating systems where a malfunction in one module can ripple to other modules and cause a full system crash or restart. This modular architecture then provides for high performance, high availability, security, and device scalability not found in other operating systems.

The Junos OS is preinstalled on your Juniper Networks device when you receive it from the factory. Thus, when you first power on the device, all software starts automatically. You simply need to configure the software so that the device can participate in the network.

You can upgrade the device software as new features are added or software problems are fixed. You normally obtain new software by downloading the software installation packages from the Juniper Networks Support Web page onto your device or onto another system on your local network. You then install the software upgrade onto the device.

Juniper Networks routing platforms run only binaries supplied by Juniper Networks, and currently do not support third-party binaries. Each Junos OS image includes a digitally

signed manifest of executables that are registered with the system only if the signature can be validated. Junos OS will not execute any binary without a registered signature. This feature protects the system against unauthorized software and activity that might compromise the integrity of your device.

- Related Documentation**
- [Junos OS Editions on page 5](#)
 - [Junos OS Installation Packages Prefixes on page 6](#)

Junos OS Editions

In releases earlier than Junos OS Release 15.1, software packages came in several major software package categories or editions, such as domestic, worldwide, or Federal Information Processing Standard (FIPS). However, starting in Junos OS Release 15.1, most of these editions were dropped. Only one edition remains, Limited. And FIPS, instead of being a separate edition, is an option you select on installation.

In support of this change, the Juniper Networks downloads page has a simplified Type/OS pulldown menu. Terms such as *Domestic*, *Continuity*, *MacSec*, *Junos US/Canada*, and *Worldwide* do not appear in the menu in favor of just *Junos* and *Junos SR*. (Where needed, the additional descriptors are still used in the table presented of available software.)

Junos OS with a null (empty) edition field is the standard image for Junos OS. The only edition remaining, Limited (for limited encryption), does not have data-plane encryption and is intended for countries in the Eurasian Customs Union (EACU). These countries have import restrictions on software containing data-plane encryption.

- Related Documentation**
- [Understanding Junos OS with Upgraded FreeBSD Package Names on page 21](#)

FIPS 140-2 Security Compliance



NOTE: Hardware platforms running Junos OS with the upgraded FreeBSD kernel employ a new naming scheme for software packages that does not recognize different major software package categories, such as domestic, world-wide, or Federal Information Processing Standard (FIPS). For more information, see [“Understanding Junos OS with Upgraded FreeBSD Package Names” on page 21](#).

For advanced network security, a special version of Junos OS, called Junos-FIPS 140-2, is available. Junos-FIPS 140-2 provides customers with software tools to configure a network of Juniper Networks devices in a FIPS environment. FIPS support includes:

- Upgrade package to convert Junos OS to Junos-FIPS 140-2
- Revised installation and configuration procedures
- Enforced security for remote access

- FIPS user roles (Crypto Officer, User, and Maintenance)
- FIPS-specific system logging and error messages
- IPsec configuration for Routing Engine–to–Routing Engine communication
- Enhanced password creation and encryption

Junos-FIPS has special installation and configuration requirements. Installation procedures include downloading the FIPS software package from www.juniper.net. For detailed guidelines on how installation and configuration procedures differ between Junos OS and Junos-FIPS 140-2, see the [Secure Configuration Guide for Common Criteria and Junos-FIPS](#).



NOTE: Junos-FIPS has special password requirements. FIPS passwords must be between 10 and 20 characters in length. Passwords must use at least three of the five defined character sets (uppercase letters, lowercase letters, digits, punctuation marks, and other special characters). If Junos-FIPS is installed on the device, you cannot configure passwords unless they meet this standard.

Related Documentation

- [Understanding Junos OS with Upgraded FreeBSD Package Names on page 21](#)

Junos OS Installation Packages Prefixes

The first part of the installation package filename is a combination of a standard prefix and product designation. [Table 3 on page 6](#) lists a variety of Junos OS package name prefixes.

Starting in Junos OS Release 15.1, certain hardware platforms run a Junos OS based on an upgraded FreeBSD kernel (hereafter called Junos OS with upgraded FreeBSD). [Table 3 on page 6](#) also indicates the prefixes used for the different platforms running Junos OS with upgraded FreeBSD.

Table 3: Installation Package Prefixes

Installation Package Prefix	Description
jinstall*	Junos OS for M Series, MX Series, T Series, TX Matrix, and TX Matrix Plus routers.
jinstall64*	64-bit Junos OS for the JCS1200 Route Reflector, TX Matrix Plus routers with 3D SIBs, and PTX Series Packet Transport Routers.
jinstall-ex*	Junos OS for the EX Series Ethernet Switch portfolio.

Table 3: Installation Package Prefixes (*continued*)

Installation Package Prefix	Description
jinstall-host-qfx*	<p>Junos OS for QFX platforms are yocto based; this prefix indicates the image includes the host and Junos OS.</p> <p>jinstall-host-qfx-5-.tgz is the package name for Junos OS on the QFX500.</p> <p>For specific examples of package names for Junos OS with upgraded FreeBSD, see “Understanding Junos OS with Upgraded FreeBSD Package Names” on page 21.</p>
jinstall-ppc*	Junos OS for the ACX Series, MX80, and MX104 routers.
junos-install*	<p>Junos OS with upgraded FreeBSD for EX Series and MX Series routers that support Junos OS with upgraded FreeBSD.</p> <p>For specific examples of package names for Junos OS with upgraded FreeBSD, see “Understanding Junos OS with Upgraded FreeBSD Package Names” on page 21.</p>
junos-srx1k3k*	Junos OS for SRX1400, SRX3400 and SRX3600.
junos-srx5000*	Junos OS for SRX5400, SRX5600, and SRX5800.
junos-srxentedge*	Junos OS for SRX1500.
junos-srxmr*	Junos OS for SRX4100 and SRX4200.
junos-srxsme*	Junos OS for SRX300, SRX320, SRX340, SRX345, and SRX550M.
junos-vmhost-install*	Junos OS with upgraded FreeBSD on VM Host.

Software Naming Convention

All Junos OS conforms to the following naming convention:

package-release-edition-cfxxx-signed.comp

For example:

jinstall-9.2R1.8-domestic-signed.tgz

where:

- **package** is the name of the Junos OS package. For 64-bit Junos OS, the package name is **package64**.
- **cfxxx** designates the CompactFlash card size to use with the software. This value is optional.

- **signed** means that the software includes a digital signature for verification purposes. This value is not used with all software packages.

All SRX Series packages conform to the following naming convention:

`junos-product-release-edition`

For example:

`junos-srxentedge-15.1X49-D30.3-domestic.tgz` (for SRX1500)
`junos-vsrx-15.1X49-D30.3-domestic.tgz` (for vSRX)
`junos-srx5000-15.1X49-D30.3-domestic.tgz` (for SRX5400, SRX5600, SRX5800)
`junos-srxsme-15.1X49-D30.3-domestic.tgz` (for SRX550M)
`junos-srxsmr-15.1X49-D65.5-domestic.tgz` (for SRX4100 and SRX4200)

where:

- **product** means SRX Series product line.
- **edition** means Junos OS for customers in the United States and Canada, and for all other customers with a valid encryption agreement. This edition includes high-encryption capabilities such as IPsec and SSH for data leaving the device.



NOTE: Starting with Junos OS Release 15.1, certain hardware platforms run Junos OS based on an upgraded FreeBSD kernel (hereafter called Junos OS with upgraded FreeBSD). Junos OS with upgraded FreeBSD has a new naming convention. For more information on this new naming convention, see [“Understanding Junos OS with Upgraded FreeBSD Package Names” on page 21](#).

Related Documentation

- [Software Naming Convention for SRX Series Devices on page 8](#)
- [Junos OS Release Numbers on page 10](#)
- [FIPS 140-2 Security Compliance on page 5](#)
- [Junos OS Editions on page 5](#)

Software Naming Convention for SRX Series Devices

Typically, you upgrade your device software by downloading a software image to your device from another system on your local network. Using the J-Web user interface or the CLI to upgrade, the device downloads the software image, decompresses the image, and installs the decompressed software. Finally, you reboot the device, at which time it boots from the upgraded software. Junos OS is delivered in signed packages that contain digital signatures to ensure official Juniper Networks software.

An upgrade software package name for an SRX Series device is in the following format:

package-name-m.nZx-distribution.tgz

- **package-name**—Name of the package; for example, junos-srxsme.
- **m.n**—Junos OS release, with m representing the major release number and n representing the minor release number; for example, 10.0.
- **Z**—Type of Junos OS release; for example, R indicates released software, and B indicates beta-level software.

For more information, see “Junos OS Release Numbers” on page 10.



NOTE: Starting with Junos OS Release 12.1X44-D10, SRX Series devices follow a special naming convention for Junos OS releases. For more information, refer to the Knowledge Base article KB30092 at <http://kb.juniper.net/InfoCenter/index?page=home>.

- **x.y**—Junos OS build number and spin number; for example, 1.8.
- **distribution**—Area for which the Junos OS package is provided. It is domestic for the United States and Canada, and it is export for worldwide distribution.

The following package name is an example of an SRX Series device upgrade Junos OS package:

junos-srxentedge-15.1X49-D30.3-domestic.tgz

Related Documentation

- [FIPS 140-2 Security Compliance on page 5](#)
- [Junos OS Release Numbers on page 10](#)
- [Downloading Software Packages from Juniper Networks on page 177](#)
- [Understanding Junos OS Upgrades for SRX Series Devices on page 160](#)

Software Package Information Security

Junos OS software is delivered in signed packages that contain digital signatures, Secure Hash Algorithm (SHA-1), and Message Digest 5 (MD5) checksums. A package is installed only if the checksum within it matches the hash recorded in its corresponding file. Which checksum is used depends on the software version:

- Digital signatures are used when you upgrade or downgrade between Junos OS Release 7.0 and a later version.
- The SHA-1 checksum is used when you upgrade or downgrade between Junos OS Release 6.4 and a later version.
- The MD5 checksum is used when you upgrade or downgrade between Junos OS Release 6.3 or earlier and a later version.

Related Documentation

- [Installation Type Overview on page 41](#)
- [Software Naming Convention for SRX Series Devices on page 8](#)

Junos OS Release Numbers

The Junos OS release number represents a particular revision of the software that runs on a Juniper Networks routing platform, for example, Junos OS Release 14.1, 14.2, or 15.1. Each Junos OS release has certain new features that complement the software processes that support Internet routing protocols, control the device's interfaces and the device chassis itself, and allow device system management. On the Juniper Networks Support webpage, you download Junos OS for a particular Junos OS release number.

The following example shows how the software release number is formatted:

m.nZb.s

For example:

14.2R3.2

Where:

- *m* is the main release number of the product.
- *n* is the minor release number of the product.
- *Z* is the type of software release. The following release types are used:
 - *R*—FRS/maintenance release software
 - *F*—Feature velocity release software



NOTE: Feature velocity releases are only in Junos OS Release 15.1.

- *B*—Beta release software
- *I*—Internal release software: private software release for verifying fixes
- *S*—Service release software: released to customers to solve a specific problem—this release will be maintained along with the life span of the underlying release
- *X*—Special (eXception) release software: releases that follow a numbering system that differs from the standard Junos OS release numbering

Starting with Junos OS Release 12.1X44-D10, SRX Series devices follow a special naming convention for Junos OS releases. For more information, refer to the Knowledge Base article KB30092 at

<http://kb.juniper.net/InfoCenter/index?page=home>.

- *b* is the build number of the product.
 - if *b=1*: software is the FRS release.
 - if *b>1*: software is a maintenance release.
- *s* is the spin number of the product.

For service release software, the release number is added at the end. For example, 14.2R3-S4.4. Here S4 represents the 4th service release on top of 14.2R3 and is the 4th respin.



NOTE: Prior to Junos OS Release 11.4, the software release number format for service releases was same as other releases. For example, 10.4S4.2 represented the 4th service release and 2nd respin of 10.4.



NOTE: Starting with Junos OS Release 15.1, certain hardware platforms run a Junos OS based on an upgraded FreeBSD kernel (hereafter called Junos OS with upgraded FreeBSD). Junos OS with upgraded FreeBSD has a new naming convention. For more information on this new naming convention, see [“Understanding Junos OS with Upgraded FreeBSD Package Names” on page 21](#).

Related Documentation

- [Junos OS Installation Packages Prefixes on page 6](#)
- [Software Naming Convention for SRX Series Devices on page 8](#)
- [Junos OS Editions on page 5](#)
- [Understanding Junos OS with Upgraded FreeBSD Package Names on page 21](#)

Installation Media

The installation media is used to recover a device from a software failure. The installation media repartitions the media and completely reinstalls Junos OS. No information from previous installations is retained during this installation. Thus, an initial configuration is required before the device can be put back into service. For more information about creating an initial configuration, see the *Getting Started Guide* for your device.



NOTE: Once you have rebuilt a device using the installation media, access to the device is restricted to the console port until the management port is configured during the initial configuration.

The following installation media files are available for download:

Installation Media	Description
floppy1-<release>*	Junos OS for the M40 router when you use the LS-120 external drive.
floppy2-<release>*	
install-media*	Junos OS for the ACX Series, M Series, MX Series, T Series, PTX Series Packet Transport Routers, TX Matrix, and TX Matrix Plus routers.



NOTE: SRX300, SRX320, SRX340, SRX345, and SRX550M devices are upgraded from loader prompt using USB or TFTP. For more details, see [“Installing Junos OS on SRX Series Devices Using a USB Flash Drive” on page 77](#) and [“Installing Junos OS on SRX Series Devices from the Boot Loader Using a TFTP Server” on page 79](#).

Installation Bundles

The installation bundle can be used to downgrade or upgrade Junos OS between minor revisions (from Release 9.1 to Release 9.2, for example). When used, the installation bundle modifies only the files required for the upgrade or downgrade between versions.



NOTE: You should only use the installation bundle under direction of a Juniper Networks support representative.

The following installation bundle files are available for download:

Installation Bundle	Description
jbundle*	Junos OS for the ACX series, M Series, MX Series, T Series, PTX Series Packet Transport Routers, TX Matrix, and TX Matrix Plus routers.



NOTE: Starting with Junos OS Release 15.1, certain hardware platforms run a Junos OS based on an upgraded FreeBSD kernel (hereafter called Junos OS with upgraded FreeBSD). Junos OS with upgraded FreeBSD has a new naming convention. For more information on this new naming convention, see [“Understanding Junos OS with Upgraded FreeBSD Package Names” on page 21](#).

Installation Modules

Installation modules are used to upgrade individual software modules in Junos OS. For example, you can upgrade only the Routing Engine software by installing the **jroute*** installation module.



NOTE: You should only use installation module files under the direction of a Juniper Networks support representative.

The following installation module files are available for download:

Installation Module	Description
---------------------	-------------

jkernel*	The kernel and network tools package. This package contains the basic operating system files.
jbase*	The base package for Junos OS. This package contains additions to the operating system.
jroute*	The Routing Engine package. This package contains the Routing Engine software.
jpfe*	The Packet Forwarding Engine package. This package contains the PFE software.
jdocs*	The documentation package. This package contains the documentation set for the software.
jcrypto*	The encryption package. This package contains the domestic version of the security software.
jweb*	The J-Web package. This package contains the graphical user interface software for M Series, MX Series, T Series, TX Matrix, and TX Matrix Plus routers.



NOTE: Starting with Junos OS Release 15.1, certain hardware platforms run a Junos OS based on an upgraded FreeBSD kernel (hereafter called Junos OS with upgraded FreeBSD). Junos OS with upgraded FreeBSD has a new naming convention. For more information on this new naming convention, see [“Understanding Junos OS with Upgraded FreeBSD Package Names”](#) on page 21.

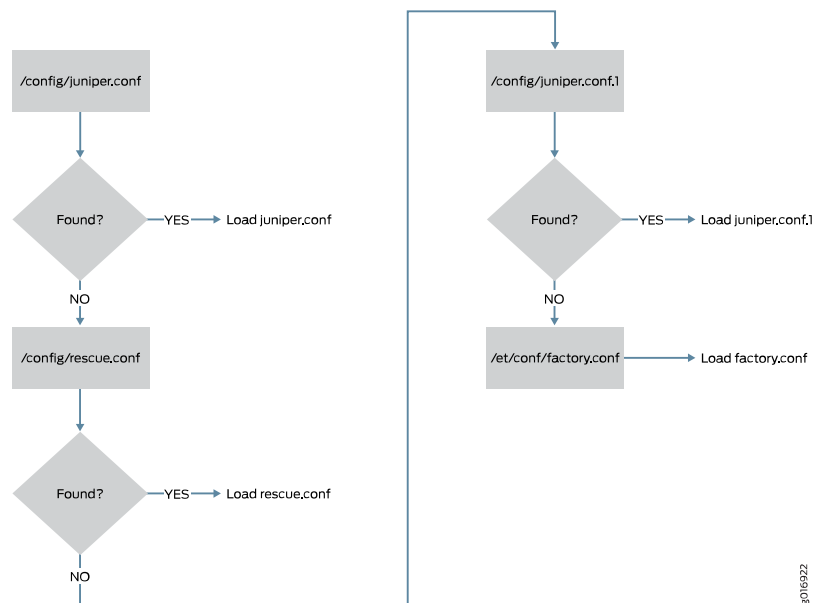
Configuration Files

All configuration settings for the device are handled in the configuration files on the device. These files are saved in the **/config** directory on the device.

Configuration File Selection Sequence

During the boot process, the device is configured based on a predefined configuration file. The device selects the configuration file based on the sequence shown in [Figure 1 on page 14](#).

Figure 1: Configuration Selection Sequence



g016922

The configuration file selection process occurs in the following sequence:

1. **/config/juniper.conf**—Active configuration file.
2. **/config/rescue.conf**—Rescue configuration file. This file is created by the router or switch administrator.
3. **/config/juniper.conf.1**—First rollback configuration.
4. **/etc/config/factory.conf**—Default factory configuration file.

The **factory.conf** file is the initial device configuration file shipped with the system. All configuration settings are returned to the factory default, and access to the device is restricted to the console. For more information about setting up your device from the factory default configuration, see the specific hardware guide for your device.

For SRX Series Services Gateways running Junos OS Release 10.0 or later, the current operational Junos OS configuration is stored in a file named **juniper.conf** and the last five committed configurations are stored in the files **juniper.conf.1** through **juniper.conf.5**. The rescue configuration is stored in a file named **rescue.conf**. These files are located in the **/config** directory available on the flash drive of the SRX Series Services Gateway.

To list the configuration files, use the **file list /config** operational mode command.

```

user@host>file list / config
/config:
.snap/
idp-dfa-status.db
juniper.conf+.gz
juniper.conf.1.gz
juniper.conf.2.gz
juniper.conf.3.gz
juniper.conf.4.gz

```

```

juniper.conf.5.gz
juniper.conf.gz
juniper.conf.md5*
jwxd_initialized
license/
license-status.db
rescue.conf.gz
usage.db
usage.db.1344499761

```

Remote Storage of Configuration Files

Configuration files can be stored off the device. This can be helpful if the device encounters a software failure or other problem that forces you to restore the device's software. Once the software is restored, you can then reload the saved configuration file. For more information about restoring Junos OS, see [“Loading and Committing the Configuration File” on page 224](#).

When the configuration file is stored off the device, you can encrypt the configuration files using the Data Encryption Standard (DES) encryption algorithm.

Related Documentation

- [Boot Sequence on SRX Series Devices on page 38](#)

Understanding Software Infrastructure and Processes on EX Series Switches

Each switch runs the Juniper Networks Junos operating system (Junos OS) for Juniper Networks EX Series Ethernet Switches on its general-purpose processors. Junos OS includes processes for Internet Protocol (IP) routing and for managing interfaces, networks, and the chassis.

Junos OS runs on the Routing Engine. The Routing Engine kernel coordinates communication among the Junos OS processes and provides a link to the Packet Forwarding Engine.

With the J-Web interface and the command-line interface (CLI) to Junos OS, you configure switching features and routing protocols and set the properties of network interfaces on your switch. After activating a software configuration, use either the J-Web or CLI user interface to monitor the switch, manage operations, and diagnose protocol and network connectivity problems.

- [Routing Engine and Packet Forwarding Engine on page 15](#)
- [Junos OS Processes on page 16](#)

Routing Engine and Packet Forwarding Engine

A switch has two primary software processing components:

- **Packet Forwarding Engine**—Processes packets; applies filters, routing policies, and other features; and forwards packets to the next hop along the route to their final destination.
- **Routing Engine**—Provides three main functions:

- Creates the packet forwarding switch fabric for the switch, providing route lookup, filtering, and switching on incoming data packets, then directing outbound packets to the appropriate interface for transmission to the network
- Maintains the routing tables used by the switch and controls the routing protocols that run on the switch.
- Provides control and monitoring functions for the switch, including controlling power and monitoring system status.

Junos OS Processes

The Junos OS running on the Routing Engine and Packet Forwarding Engine consists of multiple processes that are responsible for individual functions.

The separation of functions provides operational stability, because each process accesses its own protected memory space. In addition, because each process is a separate software package, you can selectively upgrade all or part of the Junos OS, for added flexibility.

[Table 4 on page 16](#) describes the primary Junos OS processes.

Table 4: Junos OS Processes

Process	Name	Description
Chassis process	chassisd	<p>Detects hardware on the system that is used to configure network interfaces.</p> <p>Monitors the physical status of hardware components and field-replaceable units (FRUs), detecting when environment sensors such as temperature sensors are triggered.</p> <p>Relays signals and interrupts—for example, when devices are taken offline, so that the system can close sessions and shut down gracefully.</p>
Ethernet switching process	eswd	<p>Handles Layer 2 switching functionality such as MAC address learning, Spanning Tree Protocol and access port security. The process is also responsible for managing Ethernet switching interfaces, VLANs, and VLAN interfaces.</p> <p>Manages Ethernet switching interfaces, VLANs, and VLAN interfaces.</p> <p>NOTE: The process is not applicable for EX2300 and EX3400 switches</p>
Forwarding process	pfem	<p>Defines how routing protocols operate on the switch. The overall performance of the switch is largely determined by the effectiveness of the forwarding process.</p>
Interface process	dcd	<p>Configures and monitors network interfaces by defining physical characteristics such as link encapsulation, hold times, and keepalive timers.</p>

Table 4: Junos OS Processes (*continued*)

Process	Name	Description
Management process	mgd	<p>Provides communication between the other processes and an interface to the configuration database.</p> <p>Populates the configuration database with configuration information and retrieves the information when queried by other processes to ensure that the system operates as configured.</p> <p>Interacts with the other processes when commands are issued through one of the user interfaces on the switch.</p> <p>If a process terminates or fails to start when called, the management process attempts to restart it a limited number of times to prevent thrashing and logs any failure information for further investigation.</p>
Routing protocol process	rpd	Defines how routing protocols such as RIP, OSPF, and BGP operate on the device, including selecting routes and maintaining forwarding tables.

Related Documentation • [EX Series Ethernet Switches](#)

Understanding Junos OS with Upgraded FreeBSD

Introduced in Junos OS Release 15.1, Junos OS with upgraded FreeBSD is based on an upgraded FreeBSD kernel instead of older versions of FreeBSD. Basing Junos OS on the newer kernel provides Junos OS with sophisticated processing, efficiency, and security features which do not then have to be reproduced in Junos OS. Initially, Junos OS with upgraded FreeBSD was available running on bare metal in selected platforms (see [Table 5 on page 18](#)). Starting in Junos OS Release 16.1, it is possible to run Junos OS with upgraded FreeBSD as a guest virtual machine (VM) on a Linux VM host. The PTX5000 and MX240, MX480, MX960, MX2010, and MX2020 routers run on the virtualized Junos OS with upgraded FreeBSD environment. See [“RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX with VM Host Support” on page 261](#) and [“VM Host Installation” on page 268](#) for more information.

Whether running on bare metal or as a guest VM, Junos OS with upgraded FreeBSD provides a clean-slate implementation of Junos OS on top of a pristine (minimally modified) and current version of the FreeBSD OS.

[Table 5 on page 18](#) lists the platforms currently running Junos OS with upgraded FreeBSD.



NOTE: Before upgrading to Junos OS Release 15.1 or later on these platforms, see the installation and upgrade procedures in the following topics:

- Bare metal: [“Upgrading Junos OS with Upgraded FreeBSD” on page 153](#)
- Guest VM: [“VM Host Installation” on page 268](#)

Table 5: Upgraded FreeBSD Kernel Support by Hardware Platform

Platforms	CPU Type	Release Introduced
MX240, MX480, MX960, MX2010, MX2020	Intel	15.1
	VM host	16.1
EX9200	Intel	15.1
QFX5200	Intel	15.1X53-D30
QFX10000 switches	Intel	15.1X53-D60
PTX5000	VM host	16.1

The major processing changes are as follows:

- Interactions between Junos OS and the upgraded FreeBSD kernel use well-established interfaces because Junos OS is now layered on a minimally modified and current version of FreeBSD.
- Symmetric multiprocessing (SMP) is enabled by default.
- FreeBSD provides a consistent runtime environment for all Junos OS platforms.

There are also major changes in file structures and software packages. These changes are as follows:

- New packages use XML description files instead of scripts.
- Hybrid packages are used to install legacy or replacement build images in the general form **junos-upgrade-x.tgz**, where *x* is a variable such as **mx-x86-64-15.1-20150114** (the whole package name is **junos-upgrade-mx-x86-64-15.1-20150114.tgz**).



NOTE: To upgrade VM hosts, use the **junos-vmhost-install-x.tgz** image upgrade. For more details, see [“VM Host Installation” on page 268](#).

- Multiple package sets (a collection of installed packages) are stored on the router at the same time. Sets can be either active (the currently used set), pending (the set that should be used at the next reboot), or previous (a formerly active set). Non-recovery snapshots (but not recoverable image snapshots) are available for the package sets to preserve package content lists.

There is now a separate Operations, Administration, and Maintenance (OAM) volume (**oam**) distinct from the Junos OS volume (**junos**). This provides support for downgrades from replacement build images (that is, those using the upgraded FreeBSD kernel) to the legacy Junos OS with a different kernel. The OAM volume allows you to recover the Junos OS volume using recovery snapshots. See [“Understanding Junos OS with Upgraded FreeBSD Snapshots” on page 23](#).

Release History Table

Release	Description
16.1	Starting in Junos OS Release 16.1, it is possible to run Junos OS with upgraded FreeBSD as a guest virtual machine (VM) on a Linux VM host.

Related Documentation

- [Upgrading Junos OS with Upgraded FreeBSD on page 153](#)
- [Downgrading Junos OS from Upgraded FreeBSD on page 249](#)
- [RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX with VM Host Support on page 261](#)
- [VM Host Installation on page 268](#)
- [Understanding Changes in Operations and Management for Junos OS with Upgraded FreeBSD on page 19](#)

Understanding Changes in Operations and Management for Junos OS with Upgraded FreeBSD

The upgraded FreeBSD kernel requires changes to several commands and statements and their related parameters. The new and changed actions are summarized in [Table 6 on page 19](#).

To see the changed actions for VM hosts, see “[VM Host Operations and Management](#)” on page 277.

For details on the changes listed in [Table 6 on page 19](#), see the topics covering the specific command or statement.

Table 6: New and Changed Commands and Statements for Junos OS with Upgraded FreeBSD

Command or Statement	Release Introduced	Change
<code>request system snapshot delete <i>snapshot</i></code>	15.1	New action
<code>request system snapshot recovery</code>	15.1	New action
<code>request system snapshot load <i>snapshot</i></code>	15.1	New action
<code>request system recover <i>volume</i></code>	15.1	New action: <i>volume</i> is either <code>/junos-volume</code> or <code>/oam-volume</code>
<code>request system snapshot</code>	15.1	Changed action
<code>show system snapshot</code>	15.1	Changed action
<code>request system reboot {junos network oam usb}</code>	15.1	Changed action with new media options

The new FreeBSD kernel also requires that several commands and statements are now deprecated. In some cases, these commands and statements generate an error, and, in

other cases, the result is appropriate for the new kernel. The deprecated commands and statements are summarized in [Table 7 on page 20](#). For details, see the topics covering the specific command or statement.

Table 7: Deprecated Commands and Statements for Junos OS with Upgraded FreeBSD

Deprecated Command or Configuration Statement	Release Deprecated
Deprecated Command	
<code>request system partition abort</code>	15.1
<code>request system partition compact-flash</code>	15.1
<code>request system partition hard-disk</code>	15.1
<code>request system snapshot <config-partition></code>	15.1
<code>request system snapshot <root-partition></code>	15.1
<code>request system snapshot <slice></code>	15.1
<code>request system software delete-backup</code>	15.1
<code>request system software rollback <force></code>	15.1
<code>show system processes providers</code>	15.1
<code>show system snapshot <slice></code>	15.1
Deprecated Configuration Statement	
<code>set system mirror-flash-on-disk</code>	15.1

- Related Documentation**
- [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 414](#)
 - [show system snapshot \(Junos OS with Upgraded FreeBSD\) on page 508](#)
 - [request system reboot \(Junos OS with Upgraded FreeBSD\) on page 398](#)

Understanding Junos OS with Upgraded FreeBSD Package Names

Starting in Junos OS Release 15.1, certain hardware platforms run a Junos OS based on an upgraded FreeBSD kernel (hereafter called Junos OS with upgraded FreeBSD). In releases earlier than Junos OS Release 15.1, software packages came in several major software package categories, such as domestic, worldwide, or Federal Information Processing Standard (FIPS). However, Junos OS with upgraded FreeBSD has a new naming convention: There is only one category, and FIPS, instead of being a separate category, is an option you select on installation. This topic describes the simplified naming convention for Junos OS with upgraded FreeBSD.



NOTE: Be aware that not all platforms follow this naming convention. The QFX Series devices continue to use the `jinstall` prefix for Junos OS with upgraded FreeBSD. For information on package names for Junos OS with upgraded FreeBSD for the EX Series devices, see *Understanding Junos OS with Upgraded FreeBSD Package Names for EX2300 and EX3400 Switches*.

The components of the new package naming conventions are as follows:

- **Prefix**—This is **junos-install**. This prefix takes the place of the prefix **jinstall** and the bundle **jbundle**. We still use the term *bundle* in the new package-naming convention. Starting in Junos OS Release 15.1F3, PTX5000 and MX240, MX480, MX960, MX2010, and MX2020 routers run Junos OS with upgraded FreeBSD as a guest virtual machine (VM) on a Linux VM host. The installation package for these routers has the **junos-vmhost-install** prefix.
- **Media keyword**—Added to the prefix, a media keyword is only used when the image is not for use with the **request system software add** command. Values for the **media** keyword include **usb** for images installed from a USB drive or **net** for images installed over a network; for example, the entire prefix of your package might be **junos-install-usb-**.
- **Platform**—This field indicates the major product group, such as **mx** or **qfx**.
- **Architecture**—This field indicates the CPU architecture of the platforms. Values include **x86** for Intel and **arm** for Advanced RISC Machines CPUs.
- **Application Binary Interface (ABI)**—This field indicates the “word length” of the CPU architecture. Values include **32** for 32-bit architectures and **64** for 64-bit architectures.
- **Release**—This field indicates the release number, such as **15.1R1.9**.
- **Edition**—The edition field is null (empty) for the standard (domestic) images. For jurisdictions with limits on dataplane encryption, this field is set to **limited**. Starting in Junos OS Release 17.2R1, the MX240, MX480, MX960, MX2010, and MX2020 routers with the Routing Engines RE-S-X6-64G-LT and RE-MX2K-X8-64G-LT support only Junos Limited image. The installation package for these routers has the **junos-vmhost-install** prefix. The Routing Engines are restricted to boot only the Junos Limited image. The Routing Engines fail to boot if you try to install or upgrade the router with an image other than the Junos Limited image.

As before, all images are in tarred and gzipped (.tgz) format.



NOTE: There are no longer “export” worldwide images or separate FIPS images. The keyword “signed” no longer appears because all Junos OS images are signed for validation.

Examples of valid Junos OS with upgraded FreeBSD package names include the following:

- **junos-install-mx-x86-32-15.1R1.9.tgz**—Image for a supported MX Series platform outside the jurisdictions with limits on dataplane encryption.
- **junos-install-mx-x86-32-15.1R1.9-limited.tgz**—Image for a supported MX Series platform used inside a jurisdiction with limits on dataplane encryption.
- **junos-install-usb-mx-x86-32-15.1R1.9.tgz**—Image stored on and installed from a USB drive for a supported MX Series platform outside the jurisdictions with limits on dataplane encryption.
- **jinstall-host-qfx-5e-version.tgz**—Image for QFX5110 and QFX5200.
- **jinstall-host-qfx-10-f-version.tgz**—Image for QFX10002.
- **jinstall-host-qfx-10-m-version.tgz**—Image for QFX10008 and QFX10016.

Because an upgrade to Junos OS with upgraded FreeBSD from a release earlier than Junos OS 15.1 restructures the disk file system, you can lose many configuration and log files that you might want to keep. Items that are essential can be preserved by moving or copying them to the `/var/preserve` directory.

Release History Table

Release	Description
17.2R1	Starting in Junos OS Release 17.2R1, the MX240, MX480, MX960, MX2010, and MX2020 routers with the Routing Engines RE-S-X6-64G-LT and RE-MX2K-X8-64G-LT support only Junos Limited image.
15.1F3	Starting in Junos OS Release 15.1F3, PTX5000 and MX240, MX480, MX960, MX2010, and MX2020 routers run Junos OS with upgraded FreeBSD as a guest virtual machine (VM) on a Linux VM host. The installation package for these routers has the junos-vmhost-install prefix.
15.1	Starting in Junos OS Release 15.1, certain hardware platforms run a Junos OS based on an upgraded FreeBSD kernel (hereafter called Junos OS with upgraded FreeBSD).

Related Documentation

- [Understanding Junos OS with Upgraded FreeBSD on page 17](#)
- [Upgrading Junos OS with Upgraded FreeBSD on page 153](#)

Understanding Junos OS with Upgraded FreeBSD Snapshots

Starting in Junos OS Release 15.1, certain hardware platforms have two types of snapshots. These platforms run a Junos OS based on an upgraded FreeBSD kernel instead of older versions of FreeBSD. The two types of snapshots have different content, locations, and purposes, so it is important that they are created and maintained properly. One major change is the distinction between recovery snapshots and non-recovery snapshots. The hardware platforms listed in the table in [“Understanding Junos OS with Upgraded FreeBSD” on page 17](#) have these two different types of snapshots.

One major change between Junos OS and Junos OS with Upgraded FreeBSD is the distinction between recovery snapshots and non-recovery snapshots.

The major characteristics of the recovery snapshots are as follows:

- Recovery snapshots are full copies of the packages and configuration taken at the time the snapshot command is issued.
- Recovery snapshots reside on the OAM volume or USB medium.

The major characteristics of the non-recovery snapshots are as follows:

- Non-recovery snapshots are snapshots residing on the Junos OS volume that refer to the current running set of packages and a copy of the configuration at the time the snapshot command is issued.
- Non-recovery snapshots do not need to copy the whole Junos OS installation and so are very fast.
- Non-recovery snapshots can be requested as the boot image for the next reboot.

Recovery snapshots are full copies of the packages and configuration taken at the time the snapshot command is issued. Recovery snapshots reside on the OAM volume or USB medium. Recovery snapshots take some time to complete because of the level of detail captured. Recovery snapshots can be used to recover the Junos OS volume. There is only ever one recovery snapshot on the system.

On the other hand, non-recovery snapshots are snapshots residing on the Junos OS volume that refer to the current running set of packages and a copy of the configuration at the time the snapshot command is issued. Non-recovery snapshots do not need to copy the whole Junos OS installation and so are very fast. They also consume little space, except for the **config.tgz** file. Non-recovery snapshots can be requested as the boot image for the next reboot. You can rename non-recovery snapshots and retain more than one. You rename the non-recovery snapshots with the same procedure used to rename any other file on the system.



NOTE: We recommend that you generate both a non-recovery and a recovery snapshot after you successfully upgrade to Junos OS with upgraded FreeBSD. These snapshots should be refreshed periodically.

Package sets relate to non-recovery and recovery snapshots. The **/active**, **/pending**, and **/previous** sets are all package sets. A non-recovery snapshot is also a package set in a sense, with the addition of a copy of the configuration at the time that the non-recovery snapshot is taken.

Packages that are no longer referenced by any package set or non-recovery snapshot are automatically deleted. We recommend deleting any old non-recovery snapshots after an upgrade so that old packages can be deleted and space recovered.

Some helpful commands for non-recovery snapshots are:

- **request system snapshot**—Use this command to create a non-recovery snapshot.
- **show system snapshot**—Use this command to list all the available non-recovery snapshots.
- **request system snapshot delete**—Use this command to delete a non-recovery snapshot.

A helpful command for recovery snapshots is:

- **request system snapshot recovery**—Use this command to create a recovery snapshot. You can use other parameters to determine the details of the recovery snapshot created. There is only ever one recovery snapshot on the system.

Release History Table

Release	Description
15.1	Starting in Junos OS Release 15.1, certain hardware platforms have two types of snapshots.

Related Documentation

- [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 414](#)
- [show system snapshot \(Junos OS with Upgraded FreeBSD\) on page 508](#)
- [request system reboot \(Junos OS with Upgraded FreeBSD\) on page 398](#)
- [request system software validate on \(Junos OS with Upgraded FreeBSD\) on page 449](#)
- [Understanding Junos OS with Upgraded FreeBSD Package Names on page 21](#)
- [Understanding Junos OS with Upgraded FreeBSD Package Names for EX2300 and EX3400 Switches](#)
- [Understanding Junos OS with Upgraded FreeBSD Disk Volumes on page 25](#)

Understanding Junos OS with Upgraded FreeBSD Disk Volumes

Starting in Junos OS Release 15.1, certain hardware platforms have a new disk naming convention. These platforms run a Junos OS based on an upgraded FreeBSD kernel instead of older versions of FreeBSD.

The hardware platforms listed in the table in [“Understanding Junos OS with Upgraded FreeBSD” on page 17](#) have two volumes. Technically, these volumes are **dev/gpt/oam** and **dev/gpt/junos**, but the short forms (**/junos** and **/oam**) are used in this topic. Because the **/junos** and **/oam** volumes have very different purposes, their content is different. Essentially, the **/junos** volume is used for running device software and holds configuration information and logs, whereas the **/oam** volume is used for backup copies of everything needed in the event that the **/junos** volume fails.

The **/junos** volume contains a directory named **/packages/db** that has all the components present on the device, such as **os-kernel-123**, **os-kernel-456**, and so on. A sibling directory named **/package-sets** is also present. Package sets are an important concept in Junos OS with upgraded FreeBSD.

The **/package-sets** directory contains a package listing that gathers all the components of the running Junos OS into an XML format in the **/active** subdirectory. So **os-kernel-123** could be a component in the **/package-sets/active** subdirectory, but then **os-kernel-456** could not be in the same XML package. Package sets do not contain the kernel software itself (for example), but tell the device where to find the kernel component needed for the software package. The same kernel can be present in several package listings, but only one package can be active and running on the device at any given time.

There are several directories on the **/junos** volume where a particular software package listing can be found:

- **/previous**—The package set in this directory contains the list of all the components that ran on the device before the last upgrade.
- **/active**—The package set in this directory contains the list of all the software components currently running on the device.
- **/pending**—The package set in this directory contains the list of all the software components on the device that will run after the next reboot.



NOTE: After a successful reboot, the package set in the **/pending** directory becomes the active package set, and the package set in the **/active** directory becomes the previous set.

The **/junos** volume also contains non-recovery snapshots taken with the **request system snapshot** command. These types of snapshots are new to Junos OS with upgraded FreeBSD and cannot be used for recovery of a failed system. Non-recovery snapshots are a special type of package set that includes a copy of the configuration. There can be many non-recovery snapshots on the device, and the files can be renamed. Multiple non-recovery snapshots, essentially lists of software components and configuration files,

can be helpful when major software or configuration changes are occurring and establishment of a known stable system baseline is required.

The compact flash drive is the **/oam** volume and stores recovery snapshot backup information. In case of failure of the main drive (that is, the **/junos** volume), the **/oam** volume can be used to boot the system. In order to perform this reboot, the **/oam** volume needs to have all of the information required to provide the system with a running configuration. This information is provided by the recovery snapshot, created with the **request system snapshot recovery** command. Although it can take a while to perform, the recovery snapshot establishes an **.izo** or **.iso** image of the running Junos OS.

In the case of a total failure of the **/junos** volume, the system can be booted from the **/oam** volume. The recovery snapshot can then restore the repaired system.

Release History Table

Release	Description
15.1	Starting in Junos OS Release 15.1, certain hardware platforms have a new disk naming convention. These platforms run a Junos OS based on an upgraded FreeBSD kernel instead of older versions of FreeBSD.

Related Documentation

- [Understanding Junos OS with Upgraded FreeBSD on page 17](#)
- [Upgrading Junos OS with Upgraded FreeBSD on page 153](#)
- [request system recover on page 401](#)

CHAPTER 2

Hardware Overview

- [Hardware Architecture Overview on page 27](#)
- [Hardware Overview \(ACX Series, M Series, MX Series, T Series, and TX Matrix Routers\) on page 28](#)
- [Hardware Overview of SRX Series Services Gateways on page 31](#)
- [Routing Engines and Storage Media Names \(ACX Series, M Series, MX Series, PTX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers\) on page 33](#)
- [Storage Media Names for SRX Series Devices on page 36](#)
- [Boot Sequence on M Series, MX Series, T Series, TX Matrix, TX Matrix Plus, ACX Series, and PTX Series Devices with Routing Engines on page 36](#)
- [Boot Sequence on SRX Series Devices on page 38](#)

Hardware Architecture Overview

Juniper Networks routing platforms are made up of two basic routing components:

- **Routing Engine**—The Routing Engine controls the routing updates and system management.
- **Packet Forwarding Engine (PFE)**—The Packet Forwarding Engine performs Layer 2 and Layer 3 packet switching, route lookups, and packet forwarding.

From a system administration perspective, you install the software onto the Routing Engine and during the installation, the appropriate software is forwarded to other components as necessary. Most Routing Engines include a CompactFlash card that stores Junos OS. On M Series Multiservice Edge Routers; MX240, MX480, and MX960 3D Universal Edge Routers; T Series Core Routers; and TX Matrix routers, the system also includes a hard disk or solid-state drive (SSD) that acts as a backup boot drive. PTX Series Packet Transport Routers and the TX Matrix Plus router include a solid-state drive as a backup boot drive.



NOTE: The MX80 router is a single-board router with a built-in Routing Engine and single Packet Forwarding Engine. On an MX80 router, Junos OS is stored on dual, internal NAND flash devices. These devices provide the same functionality as a CompactFlash card and hard disk or solid-state drive (SSD).



NOTE: The ACX Series router is a single board router with a built-in Routing Engine and one Packet Forwarding Engine. The ACX router supports dual-root partitioning, which means that the primary and backup Junos OS images are kept in two independently bootable root partitions. If the primary partition becomes corrupted, the system remains fully functional by booting from the backup Junos OS image located in the other root partition.

On routing platforms with dual Routing Engines, each Routing Engine is independent with regard to upgrading the software. To install new software on both Routing Engines, you need to install the new software on each Routing Engine. On platforms with dual Routing Engines configured for high availability, you can use the unified in-service software upgrade procedure to upgrade the software. For more information about this procedure, see the [High Availability Feature Guide for Routing Devices](#).

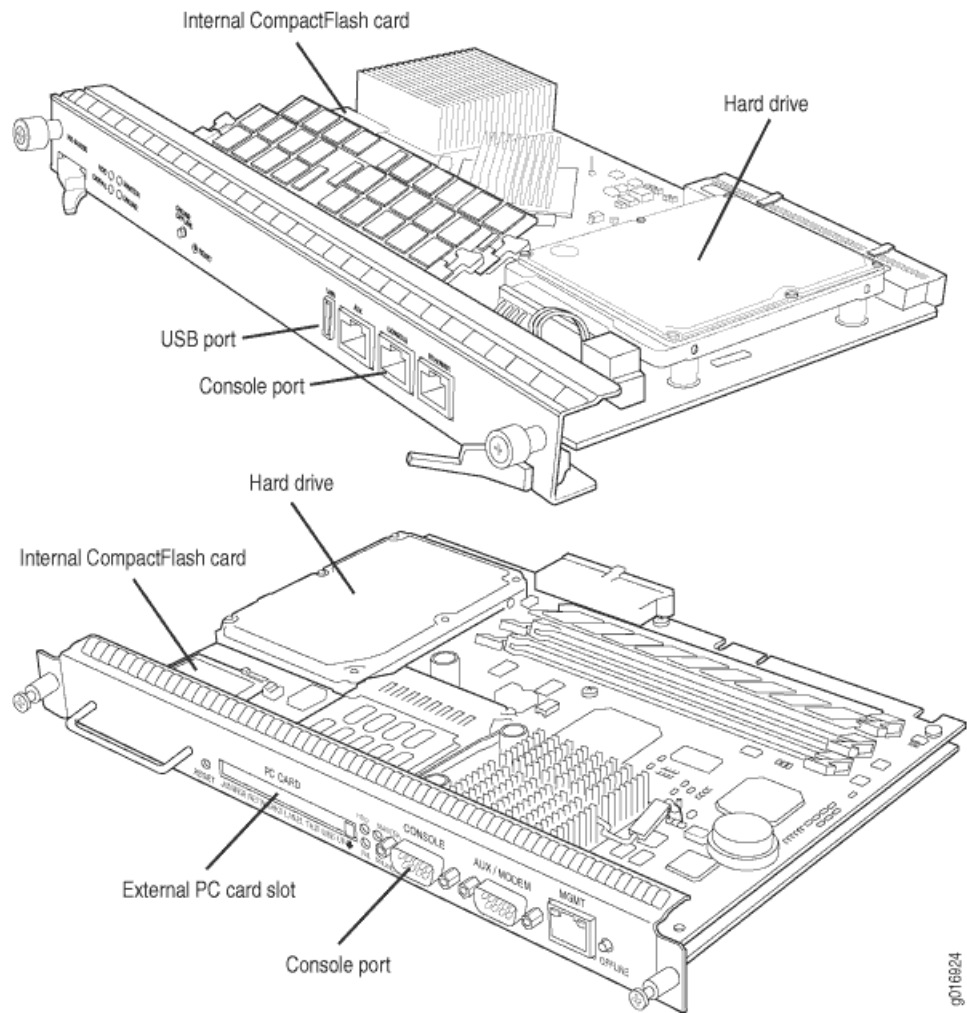
Related Documentation

- [Dual-Root Partitioning ACX Series Routers Overview](#)

[Hardware Overview \(ACX Series, M Series, MX Series, T Series, and TX Matrix Routers\)](#)

Figure 2 on page 29 shows examples of Routing Engines.

Figure 2: Routing Engines



The ACX Series, M Series, MX Series, PTX Series, T Series, TX Matrix, and TX Matrix Plus routers include the following:

- [System Memory on page 29](#)
- [Storage Media on page 30](#)

System Memory

Starting with Junos OS Release 9.0, all routing platforms require a minimum of 512 MB of system memory on each Routing Engine. All M7i and M10i routers delivered before December 7, 2007, had 256 MB of memory. These routers require a system memory upgrade before you install Junos OS Release 9.0 or a later release. To determine the amount of memory currently installed on your system, use the **show chassis routing-engine** command in the command-line interface (CLI).

For more information about upgrading your M7i or M10i router, see the Customer Support Center JTAC Technical Bulletin PSN-2007-10-001:

<https://www.juniper.net/alerts/viewalert.jsp?txtAlertNumber=PSN-2007-10-001&actionBtn=Search>.

ACX2000 routers are shipped with 2 GB of memory and ACX1000 routers with 1 GB of memory.

Storage Media

Except for the ACX Series, MX80 routers, and MX104 routers, the M Series, MX Series, PTX Series, T Series, TX Matrix, and TX Matrix Plus routers use the following media storage devices:

- CompactFlash card—The CompactFlash card is typically the primary storage device for most routers.



NOTE: M7i and M10i routers using RE-400 are not delivered from the factory with the CompactFlash card installed. In this case, the hard disk is the primary and only boot device. The M7i and M10i routers with RE-400 can be upgraded to include the CompactFlash card.

- Hard disk or solid-state drive—For most routers, a hard disk or solid-state drive is the secondary boot device. When the CompactFlash card is not installed on the router, the hard disk or the solid-state drive becomes the primary boot device. The hard disk or solid-state drive is also used to store system log files and diagnostic dump files.
- Emergency boot device—Depending on the router, the emergency boot device can be a PC card, a USB storage device, or an LS-120 floppy disk.

On MX80 routers, the internal NAND flash devices (first *da0*, then *da1*) act as the primary and secondary boot devices.

On ACX Series routers, the internal NAND flash devices (first *da0s1*, then *da0s2*) act as the primary and secondary boot devices.

Emergency boot devices can be used to revive a routing platform that has a damaged Junos OS. When an emergency boot device is attached to the router, the router attempts to boot from that device before it boots from the CompactFlash card, solid-state drive (SSD), or hard disk.

On an ACX Series router, the emergency boot device is a USB storage device.

On MX104 routers, the internal NAND flash device (*da0*) mounted on the internal eUSB card acts as the primary boot and storage device. On MX104 routers, the emergency boot device is a USB storage device that is plugged into one of the USB ports in the front plate.

When booting from an emergency boot device, the router requests a boot acknowledgment on the console interface. If you enter yes, the emergency boot device repartitions the primary boot device and reloads Junos OS onto the primary boot device. After the loading is complete, the routing platform requests that you remove the

emergency boot device and reboot the system. After the reboot is complete, you must perform an initial configuration of the router before it can be used on your network.



NOTE: For routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines, a set of two 64-GB SSDs are available for storage and redundancy. For more information see Storage Partitioning and Redundancy topic in “Salient Features of the RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines” on page 263 section.

Hardware Overview of SRX Series Services Gateways

SRX Series Device Overview

Figure 3 on page 31 shows an example of SRX240 device.

Figure 3: SRX240 Device Front Panel

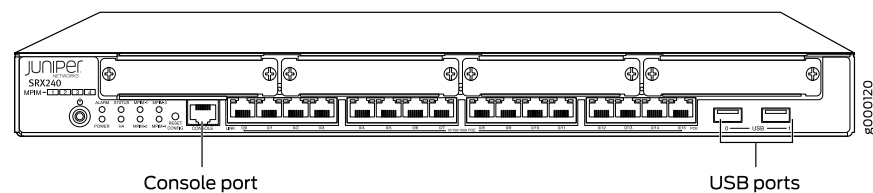


Figure 4 on page 31 shows an example of SRX650 device.

Figure 4: SRX650 Device System Routing Engine

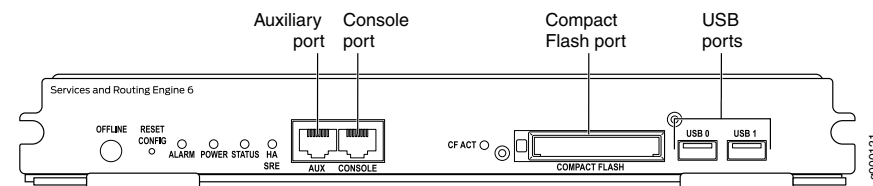


Figure 5 on page 31 shows the front panel of an SRX345 device.

Figure 5: SRX345 Device Front Panel

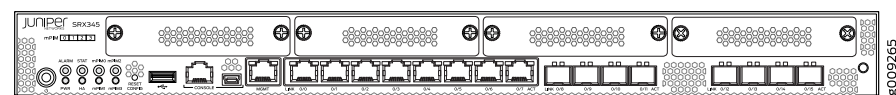


Figure 6 on page 31 shows an example of an SRX1500 device.

Figure 6: SRX1500 Device Front Panel

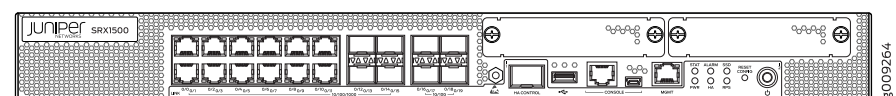


Figure 7 on page 32 shows an example of an SRX5800 device Routing Engine.

Figure 7: SRX5800 Device Routing Engine

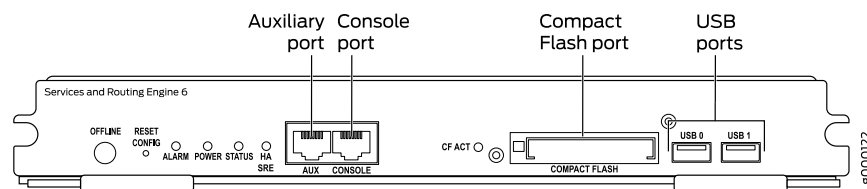
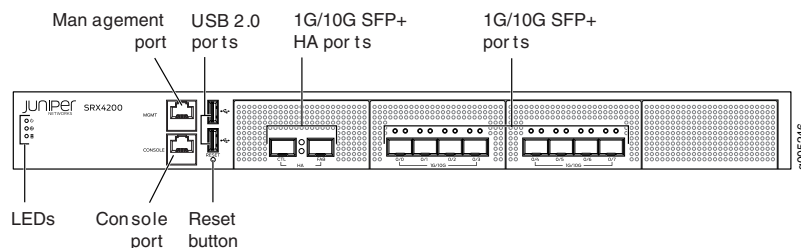


Figure 8 on page 32 shows an example of an SRX4200 device.

Figure 8: SRX4200 Services Gateway Front Panel



System Memory

The amount of free disk space necessary to upgrade a device with a new version of Junos OS can vary from one release to another for different SRX Series devices. Check the Junos OS software version you are installing to determine the free disk space requirements.

To determine the amount of free disk space on the device, issue the **show system storage detail** command. The command output displays statistics about the amount of free disk space in the device file systems.

Storage Media

The SRX100, SRX210, SRX240, Services Gateway can boot from the following storage media (in the order of priority):

- Internal NAND Flash (default; always present)
- USB storage key (alternate)

The SRX550 and SRX650 Services Gateway can boot from the following storage media (in the order of priority):

- CompactFlash (default; always present)
- External CompactFlash card (alternate) (SRX650 only)
- USB storage key (alternate)

The SRX300, SRX320, SRX340, 345 Services Gateway can boot from the following storage media (in the order of priority):

- Internal NAND flash device mounted on the internal eUSB card (default; always present)

- USB storage key (alternate)

The SRX550M Services Gateway can boot from the following storage media (in the order of priority):

- CompactFlash (default; always present)
- USB storage key (alternate)

SRX1500 device use the following media storage devices:

- Internal eSATA flash disk (default; always present)
- SSD

SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, SRX5800 devices use the following media storage devices:

- The CompactFlash card in the Routing Engine
- The hard disk in the Routing Engine



NOTE: You can also use a Junos OS image stored on a USB flash drive that you insert into the Routing Engine faceplate.

The SRX4100 and SRX4200 devices include the following storage media:

- Internal eSATA flash disk (default; always present)
- SSD

Related Documentation

- [Boot Sequence on SRX Series Devices on page 38](#)
- [Verifying PIC Combinations on page 334](#)

Routing Engines and Storage Media Names (ACX Series, M Series, MX Series, PTX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers)

[Table 8 on page 33](#) specifies the storage media names by Routing Engine. The storage media device names are displayed when the router boots.

Table 8: Routing Engines and Storage Media Names (ACX Series, M Series, MX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers)

Routing Engine	CompactFlash Card	Hard Disk	Solid-State Drive	Removable Media Emergency Boot Device
RE-400-768 (RE5)	ad0	ad1	No	ad3
RE-600-2048 (RE3)	ad0	ad1	No	ad3

Table 8: Routing Engines and Storage Media Names (ACX Series, M Series, MX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers) (continued)

Routing Engine	CompactFlash Card	Hard Disk	Solid-State Drive	Removable Media Emergency Boot Device
RE-850-1536 (RE-850)	ad0	ad1	No	ad3
RE-A-1000-2048 (RE-A-1000)	ad0	ad2	No	da0
RE-A-1800x2 (RE-A-1800)	ad0	No	Yes SSD1: ad1 SSD2: ad2	da0
RE-S-1300-2048 (RE-S-1300)	ad0	ad2	No	da0
RE-S-1800x2 RE-S-1800x4 (RE-S-1800)	ad0	No	Yes SSD1: ad1 SSD2: ad2	da0
RE-B-1800X1-4G-S	ad0	No	Yes SSD1: ad1	da0
RE-1600-2048 (RE4)	ad0	ad1	No	ad3 and ad4
RE-A-2000-4096 (RE-A-2000)	ad0	ad2	No	da0
RE-S-2000-4096 (RE-S-2000)	ad0	ad2	No	da0
RE-MX-104	No	da0	No	da1 and da2
RE-DUO-C2600-16G (RE-DUO-2600)	ad0	No	ad1	da0
RE-DUO-C1800-8G- (RE-DUO-1800)	ad0	No	ad1	da0
RE-DUO-C1800-16G	ad0	No	ad1	da0
RE-JCS1200-1x2330	da0	da1	No	da2

Table 8: Routing Engines and Storage Media Names (ACX Series, M Series, MX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers) (*continued*)

Routing Engine	CompactFlash Card	Hard Disk	Solid-State Drive	Removable Media Emergency Boot Device
RE-PTX-X8-64G	No	No	SSD1 SSD2	USB
RE-S-X6-64G	No	No	SSD1 SSD2	USB
REMX2K-X8-64G	No	No	SSD1 SSD2	USB



NOTE: On MX80 routers, the Routing Engine is a built-in device and has no model number. The dual internal NAND flash devices are da0 and da1. The USB storage device is da2.



NOTE: On ACX Series routers, the Routing Engine is a built-in device which does not have a model number. The dual internal NAND flash devices are da0s1 and da0s2. The USB storage device is da0s2a. Use the `show chassis hardware models` command to obtain the field-replaceable unit (FRU) model number—for example, ACX2000BASE-DC for the ACX2000 router.

To view the storage media currently available on your system, use the CLI **show system storage** command. For more information about this command, see the *CLI User Guide*.

Related Documentation

- *Supported Routing Engines by Router*
- *Routing Engine Specifications*
- *RE-S-1300 Routing Engine Description*
- *RE-S-2000 Routing Engine Description*
- *RE-S-1800 Routing Engine Description for MX Series*
- *JCS1200 Routing Engine Description*

Storage Media Names for SRX Series Devices

Table 9 on page 36 specifies the storage media names used by the SRX Series devices. The storage media device names are displayed as the device boots.

Table 9: Storage Media Names

Device	Internal CompactFlash Card	USB Storage Media Devices
SRX Series device	da0	da1

To view the storage media currently available on your system, use the CLI **show system storage** command.

Related Documentation

- [Hardware Overview of SRX Series Services Gateways on page 31](#)
- [Boot Sequence on SRX Series Devices on page 38](#)

Boot Sequence on M Series, MX Series, T Series, TX Matrix, TX Matrix Plus, ACX Series, and PTX Series Devices with Routing Engines



NOTE: For information about which Routing Engines are supported by each device, see http://www.juniper.net/techpubs/en_US/release-independent/junos/topics/reference/general/routing-engine-m-mx-t-series-support-by-chassis.html.

The M Series, MX Series (except for the MX80 routers and the MX104 routers), T Series, and TX Matrix routers with a Routing Engine that has a hard disk attempt to boot from the storage media in the following order:

1. Removable media emergency boot device, such as a PC Card (if present)
2. CompactFlash card (if present)
3. Hard disk

The M Series and MX Series with a Routing Engine that has a solid-state drive (SSD) attempt to boot from the storage media in the following order:

1. USB media emergency boot device (if present)
2. CompactFlash card
3. Solid-state drive (SSD) in the SSD slot 1 or SSD slot 2 (if present)

MX80 routers attempt to boot from the storage media in the following order:

1. USB media emergency boot device
2. Dual, internal NAND flash device (first da0, then da1)

MX104 routers attempt to boot from the storage media in the following order:

1. USB storage media device
2. Internal NAND flash device (**da0**)

The T Series routers with a Routing Engine that has a solid-state drive (SSD), and TX Matrix Plus routers attempt to boot from the storage media in the following order:

1. USB media emergency boot device
2. CompactFlash card (if present)
3. Solid-state drive (SSD) in the Disk 1 slot (if present)



NOTE: The Disk 2 slot is not currently supported.

4. Storage media available on the LAN

The ACX Series routers attempt to boot from the storage media in the following order:

1. USB storage media device
2. Dual, internal NAND flash device (first da0s1, then da0s2)

The PTX Series Packet Transport Routers attempt to boot from the storage media in the following order:

1. USB media emergency boot device
2. CompactFlash card
3. Solid-state drive (SSD) in the Disk 1 slot (if present)
4. Storage media available on the LAN



NOTE: Do not insert an emergency boot device during normal operations. The router does not operate normally when it is booted from an emergency boot device.

If the router boots from an alternate boot device, Junos OS displays a message indicating this when you log in to the router. For example, the following message shows that the software booted from the hard disk (**/dev/ad1s1a**):

```
login: username
Password: password
Last login: date on terminal
```

```
--- Junos 8.0 R1 built date
```

```
---
```

```
--- NOTICE: System is running on alternate media device (/dev/ad2s1a).
```

This situation results when the router detects a problem with the primary boot device—usually the CompactFlash card—that prevents it from booting, and consequently boots from the alternate boot device (the hard disk drive). When this happens, the primary boot device is removed from the list of candidate boot devices. The problem is usually a serious hardware error. We recommend you contact the Juniper Networks Technical Assistance Center (JTAC).



NOTE: On MX104 routers, if the router boots from an alternate boot device, Junos OS does not display any message indicating this when you log in to the router.

When the router boots from the alternate boot device, the software and configuration are only as current as the most recent **request system snapshot** command. However, if the **mirror-flash-on-disk** command was enabled, then the hard disk drive contains a synchronized, mirror image of the compact flash drive and therefore the current software and configuration.

Related Documentation

- [Routing Engine Specifications](#)

Boot Sequence on SRX Series Devices

On SRX Series devices, the device attempts to boot from the storage media in the following order:

- Internal CompactFlash card
- Internal eSATA flash disk (for SRX1500, SRX4100, and SRX4200 devices)
- USB storage media device

Related Documentation

- [Hardware Overview of SRX Series Services Gateways on page 31](#)
- [Storage Media Names for SRX Series Devices on page 36](#)

PART 2

Installing Junos Software

- [Installation Overview on page 41](#)
- [Performing a Standard or Change Category Installation on page 47](#)
- [Configuring Zero Touch Provisioning on page 85](#)
- [Configuring Automatic Installation of Configuration Files on page 95](#)
- [Configuring Dual-Root Partitions for High Availability on page 111](#)
- [Upgrading Software on page 127](#)
- [Booting a Device Using a System Snapshot on page 197](#)
- [Performing a Recovery Installation on page 205](#)
- [Reinstalling Software on page 227](#)
- [Downgrading Software on page 249](#)
- [Rebooting or Halting Software Processes on a Device on page 253](#)

CHAPTER 3

Installation Overview

- [Installation Type Overview on page 41](#)
- [Installation Categories on the ACX Series, M Series, MX Series, T Series, TX Matrix, and TX Matrix Plus Routers on page 42](#)
- [Installation Categories on SRX Series Devices on page 43](#)
- [Understanding Software Installation on EX Series Switches on page 44](#)

Installation Type Overview

The three types of installations used to upgrade or downgrade your routing platform are standard installation, category change, and recovery. The standard installation is the standard method of upgrading and downgrading the software. Use a category change installation when you are moving from one software category to another; for example, if you are changing the device from using the standard Junos OS to the Junos-FIPS category. Perform a recovery installation when the software on the device is damaged or otherwise unable to accommodate a software upgrade or downgrade.

Standard Installation

A standard installation is the typical method used to upgrade or downgrade software on the server. This method uses the installation package that matches the installation package already installed on the system. For example, you might upgrade an M120 router running Junos OS installed using the **jinstall*** installation package. If you upgrade the router from Release 9.0R2.10 to Release 9.1R1.8, you use the **jinstall-9.1R1.8-domestic-signed.tgz** installation package. For information on the different installation packages available, see [“Junos OS Installation Packages Prefixes” on page 6](#).

Category Change Installation

The category change installation process is used to move from one category of Junos OS to another on the same router; for example, moving from a Junos OS standard installation on an M Series, MX Series, or T Series router to a Junos-FIPS installation. When moving from one installation category to another, you need to be aware of the restrictions regarding this change.



NOTE: Juniper Networks does not support using the `request system software rollback` command to restore a different installation category on the device. When installing a different Junos OS category on a device, once the installation is complete, you should execute a `request system snapshot` command to delete the backup installation from the system.

Recovery Installation

A recovery installation is performed to repair a device with damaged software or a condition that prevents the upgrade, downgrade, or change in installation category of the software.

For example, you may need to perform a recovery installation to change a device's software category from Junos-FIPS to standard Junos OS.

Related Documentation

- [Junos OS Installation Packages Prefixes on page 6](#)
- [Software Naming Convention for SRX Series Devices on page 8](#)

Installation Categories on the ACX Series, M Series, MX Series, T Series, TX Matrix, and TX Matrix Plus Routers

The following installation categories are available with the ACX Series, M Series, MX Series, T Series, TX Matrix, and TX Matrix Plus routers:

- Standard Junos OS, domestic—`jinstall-<release>-domestic-signed.tgz`

This software includes high-encryption capabilities for data leaving the router. Because of U.S. government export restrictions, this software can only be installed on systems within the United States and Canada. For all other customers, a valid encryption agreement is required to use this software edition. Furthermore, no router can be shipped out of the United States or Canada without the domestic edition first being overwritten by the export edition. There are no current system-enforced restrictions when you install this software category.

- Standard Junos OS, export—`jinstall-<release>-export-signed.tgz`

This software does not include high-encryption capabilities. It can be installed on any system worldwide. There are no current system-enforced restrictions when you install this software category.

- Junos-FIPS—`junos-juniper-<release>-domestic-signed.tgz` and `junos-juniper-<release>-fips-signed.tgz`

The Junos-FIPS OS base provides customers with the software tools to configure the router for use within a Federal Information Processing Standards (FIPS) environment. Once you have installed this software category onto a router, you cannot install a different software category on the router using the `request system software add` command. When attempting to install a different Junos OS category package on the router, you receive the following warning message:

WARNING: Package `jinstall-<release>-<edition>-signed` is not compatible with this system.

WARNING: Please install a supported package (`junos-juniper-*.tgz`).

To return to a standard Junos OS category installation, you must perform a system recovery installation of the software. All configuration files, logs, and other data files on the server are overwritten during a recovery installation.

For more information about Junos-FIPS OS base, see [“FIPS 140-2 Security Compliance” on page 5](#).



NOTE: When you install a Junos OS installation package, the previous installation is maintained as a backup installation. You should issue a **request system software snapshot** command to overwrite the backup files any time you change software categories on a router. This is mandatory if the router is to be shipped outside of the United States or Canada after the Export edition of Junos OS has been installed. There are no current system-enforced restrictions when you install this software category.

Installation Categories on SRX Series Devices

The following installation categories are available with the SRX Series devices:

- Junos OS, domestic—`junos-srxsme-<release>-domestic.tgz` for SRX Series devices. .

This software includes high-encryption capabilities for data leaving the router. Because of U.S. government export restrictions, this software can only be installed on systems within the United States and Canada. For all other customers, a valid encryption agreement is required to use this software edition. Furthermore, no router can be shipped out of the United States or Canada without the domestic edition first being overwritten by the export edition. There are no current system-enforced restrictions when you install this software category.

- Junos OS, export—`junos-srxsme-<release>-export.tgz` for SRX Series devices.

This software does not include high-encryption capabilities. It can be installed on any system worldwide. There are no current system-enforced restrictions when you install this software category.

Related Documentation

- [Installation Type Overview on page 41](#)
- [Software Package Information Security on page 9](#)
- [Software Naming Convention for SRX Series Devices on page 8](#)

Understanding Software Installation on EX Series Switches

A Juniper Networks EX Series Ethernet Switch is delivered with the Juniper Networks Junos operating system (Junos OS) preinstalled. As new features and software fixes become available, you must upgrade your software to use them. You can also downgrade Junos OS to a previous release.

This topic covers:

- [Overview of the Software Installation Process on page 44](#)
- [Software Package Security on page 44](#)
- [Installing Software on a Virtual Chassis on page 45](#)
- [Installing Software on Switches with Redundant Routing Engines on page 45](#)
- [Installing Software Using Automatic Software Download on page 45](#)
- [Autoinstalling a Configuration File on an EX2200 or EX3300 Switch from a Disk-on-Key USB Memory Stick on page 46](#)
- [Installing Software on an EX2300 or EX3400 Switch on page 46](#)
- [Troubleshooting Software Installation on page 46](#)

Overview of the Software Installation Process

An EX Series switch is delivered with a domestic version of Junos OS preinstalled. When you connect power to the switch, it starts (boots) from the installed software.

You upgrade Junos OS on an EX Series switch by copying a software package to your switch or another system on your local network, then use either the J-Web interface or the command-line interface (CLI) to install the new software package on the switch. Finally, you reboot the switch; it boots from the upgraded software. After a successful upgrade, you should back up the new current configuration to a secondary device. You should follow this procedure regardless of whether you are installing a domestic or controlled Junos OS package.

During a successful upgrade, the upgrade package removes all files from `/var/tmp` and completely reinstalls the existing software. It retains configuration files, and similar information, such as secure shell and host keys, from the previous version. The previous software package is preserved in a separate disk partition, and you can manually revert back to it if necessary. If the software installation fails for any reason, such as loss of power during the installation process, the system returns to the originally active installation when you reboot.

Software Package Security

All Junos OS releases are delivered in signed packages that contain digital signatures to ensure official Juniper Networks software. For more information about signed software packages, see the [Junos OS Installation and Upgrade Guide](#).

Installing Software on a Virtual Chassis

You can connect individual EX Series switches together to form one unit and manage the unit as a single device, called a Virtual Chassis. The Virtual Chassis operates as a single network entity composed of member switches. Each member switch in a Virtual Chassis must be running the same version of Junos OS. See the *EX Series Virtual Chassis Software Features Overview* for a list of switches that can be used in a Virtual Chassis.

For ease of management, a Virtual Chassis provides flexible methods to upgrade software releases. You can deploy a new software release to all member switches of a Virtual Chassis or to only a particular member switch.

You can also upgrade the software on an EX4200, EX4500, mixed EX4200 and EX4500, and EX8200 Virtual Chassis using nonstop software upgrade (NSSU). NSSU takes advantage of graceful Routing Engine switchover (GRES) and nonstop active routing (NSR) to ensure no disruption to the control plane during the upgrade. You can minimize disruption to network traffic by defining link aggregation groups (LAGs) such that the member links of each LAG reside on different line cards (on EX8200 Virtual Chassis) or on different members (on EX4200, EX4500, mixed EX4200 and EX4500 Virtual Chassis). During an NSSU, the line cards and Virtual Chassis members are upgraded one at a time, so that traffic continues to flow through the other line cards or members while that line card or member is being upgraded.

Installing Software on Switches with Redundant Routing Engines

You can install software on a switch with redundant Routing Engines in one of two ways:

- Perform an NSSU—An NSSU upgrades both Routing Engines with a single command and with a minimum of network disruption. An NSSU takes advantage of GRES and NSR to ensure no disruption to the control plane. You can minimize disruption to network traffic by defining LAGs such that the member links of each LAG reside on different line cards. The line cards are upgraded one at a time, so that traffic continues to flow through the other line cards while a line card is being upgraded.

You cannot use NSSU to downgrade the software running on a switch.

For more information about NSSU, see [“Understanding Nonstop Software Upgrade on EX Series Switches” on page 136](#). See the *EX Series Switch Software Features Overview* for a list of switches that support NSSU.

- Upgrade each Routing Engine manually—You can perform a Junos OS installation on each Routing Engine separately, starting with the backup Routing Engine. You can use this procedure to downgrade the software running on a switch. See *Installing Software on an EX Series Switch with Redundant Routing Engines (CLI Procedure)*.

Installing Software Using Automatic Software Download

The automatic software download feature uses the DHCP message exchange process to download and install software packages. Users can define a path to a software package on the DHCP server, and then the DHCP server communicates this path to EX Series switches acting as DHCP clients as part of the DHCP message exchange process. The DHCP clients that have been configured for automatic software download receive these

messages and, when the software package name in the DHCP server message is different from that of the software package that booted the DHCP client switch, download and install the software package. See [“Upgrading Software by Using Automatic Software Download”](#) on page 144.

Autoinstalling a Configuration File on an EX2200 or EX3300 Switch from a Disk-on-Key USB Memory Stick

You can use an autoinstallation process to configure the software on an EX2200 or EX3300 switch. You can use a configuration file that is in either text format or XML format. If you want to use an XML-formatted file, you use a Junos Space platform to create the configuration file. You place the configuration file on a Disk-on-Key USB memory stick.

Installing Software on an EX2300 or EX3400 Switch

Before installing software on an EX2300 or EX3400 switch:

- Ensure that at least 620 MB of disk space is available in the system before downloading the software installation package to the `/var/tmp` directory. Use the command **show system storage** to get details of the available space.
- If the space available is inadequate, use the command **request system storage cleanup**. Additionally, you can manually delete any other log or unwanted files from the `/var/tmp` or `/var/log` directories.

You can now follow the procedure in *Installing Software on an EX Series Switch with a Single Routing Engine (CLI Procedure)* to complete the software installation.

Troubleshooting Software Installation

If Junos OS loads but the CLI is not working for any reason, or if the switch has no software installed, you can use the recovery installation procedure to install the software on the switch. See [“Troubleshooting Software Installation”](#) on page 329.



NOTE: You can also use this procedure to load two versions of Junos OS in separate partitions on the switch.

Related Documentation

- [Downloading Software Packages from Juniper Networks on page 51](#)
- [Installing Software on EX Series Switches \(J-Web Procedure\) on page 66](#)
- [Installing Software on an EX Series Switch with a Single Routing Engine \(CLI Procedure\)](#)
- [Installing Software on an EX Series Switch with Redundant Routing Engines \(CLI Procedure\)](#)
- [Understanding Nonstop Software Upgrade on EX Series Switches on page 136](#)

CHAPTER 4

Performing a Standard or Change Category Installation

- [Checking the Current Configuration and Candidate Software Compatibility on page 47](#)
- [Determining the Junos OS Version on page 48](#)
- [Downloading Software on page 48](#)
- [Downloading Software Packages from Juniper Networks on page 51](#)
- [Understanding Download Manager for SRX Series Devices on page 51](#)
- [Understanding the Console Port on page 54](#)
- [Accessing a Junos OS Device the First Time on page 55](#)
- [Backing Up the Existing Installation on Routers on page 56](#)
- [Backing Up the Current Installation on SRX Series Devices on page 57](#)
- [Installing the Software Package on a Router with a Single Routing Engine on page 59](#)
- [Installing the Software Package on a Router with Redundant Routing Engines on page 60](#)
- [Repartitioning Routing Engine System Storage to Increase the Swap Partition on page 66](#)
- [Installing Software on EX Series Switches \(J-Web Procedure\) on page 66](#)
- [Registering the EX Series Switch with the J-Web Interface on page 68](#)
- [Connecting and Configuring an EX Series Switch \(CLI Procedure\) on page 69](#)
- [Connecting and Configuring an EX9200 Switch \(CLI Procedure\) on page 72](#)
- [Preparing the USB Flash Drive to Upgrade Junos OS on SRX Series Devices on page 75](#)
- [Installing Junos OS on SRX Series Devices Using a USB Flash Drive on page 77](#)
- [Upgrading the Boot Loader on SRX Series Devices on page 78](#)
- [Installing Junos OS on SRX Series Devices from the Boot Loader Using a TFTP Server on page 79](#)
- [Installing Junos OS on SRX Series Devices from the Boot Loader Using a USB Storage Device on page 82](#)

Checking the Current Configuration and Candidate Software Compatibility

When you upgrade or downgrade Junos OS, we recommend that you include the **validate** option with the **request system software add** command to check that the candidate

software is compatible with the current configuration. By default, when you add a package with a different release number, the validation check is done automatically.



NOTE: On an ACX Series router, you must ensure that the primary and backup partitions are synchronized after an upgrade by issuing the `request system snapshot` command.

**Related
Documentation**

- [Preparing Your SRX Series Device for Junos OS Upgrades on page 162](#)
- [Downloading Software Packages from Juniper Networks on page 177](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
- [Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server on page 181](#)
- [request system snapshot \(Maintenance\) on page 416](#)
- [request system software add \(Maintenance\) on page 432](#)

Determining the Junos OS Version

To determine which software packages are running on the device and to get information about these packages, use the **show version** operational mode command at the top level of the command-line interface (CLI).



NOTE: The `show version` command does not show the software category installed, only the release number of the software.

**Related
Documentation**

- [Preparing Your SRX Series Device for Junos OS Upgrades on page 162](#)
- [Downloading Software Packages from Juniper Networks on page 177](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
- [Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server on page 181](#)

Downloading Software

You can download the software in one of two ways:

- [Downloading Software with a Browser on page 49](#)
- [Downloading Software Using the Command-Line Interface on page 49](#)

Downloading Software with a Browser

You download the software package you need from the Juniper Networks Support website at <http://www.juniper.net/support/>.



NOTE: To access the download section, you must have a service contract and an access account. If you need help obtaining an account, complete the registration form at the Juniper Networks website: <https://www.juniper.net/registration/Register.jsp>.

To download the software:

1. In a browser, go to <http://www.juniper.net/support/>.
The Support page opens.
2. In the Download Software section, select the software version to download.
Depending on your location, select Junos Canada and US, or Junos Worldwide.
3. Select the current release to download.
4. Click the Software tab and select the Junos OS installation package to download.
A dialog box opens.
5. Save the file to your system. If you are placing the file on a remote system, you must make sure that the file can be accessible by the router or switch using HTTP, FTP, or scp.

Downloading Software Using the Command-Line Interface

Download the software package you need from the Juniper Networks Support website at <http://www.juniper.net/support/>, and place the package on a local system. You can then transfer the downloaded package to the device using either the router or switch command-line interface, or the local system command-line interface.



NOTE: To access the download section, you must have a service contract and an access account. If you need help obtaining an account, complete the registration form at the Juniper Networks website: <https://www.juniper.net/registration/Register.jsp>.

Before you transfer the software package, ensure that the FTP service is enabled on the device.

Enable the FTP service using the **set system services ftp** command:

```
user@host# set system services ftp
```

To transfer the software package using the device command-line interface:

1. From the router or switch command line, initiate an FTP session with the local system (host) where the package is located using the **ftp** command:

```
user@host> ftp host
```

host is the hostname or address of the local system.

2. Log in with your customer support–supplied username and password:

```
User Name: username  
331 Password required for username.  
Password: password
```

Once your credentials have been validated, the FTP session opens.

3. Navigate to the software package location on the local system, and transfer the package using the **get** command:

```
user@host> get installation-package
```

Following is an example of an *installation-package* name:
jinstall-9.2R1.8–domestic-signed.tgz

4. Close the FTP session using the **bye** command:

```
user@host> bye  
Goodbye
```

To transfer the package using the local system command-line interface:

1. From the local system command line, initiate an FTP session with the device using the **ftp** command:

```
user@host> ftp host
```

host is the hostname or address of the router or switch.

2. Log in with your customer support–supplied username and password:

```
User Name: username  
331 Password required for username.  
Password: password
```

Once your credentials have been validated, the FTP session opens.

3. Navigate to the software package location on the local system, and transfer the package using the **put** command:

```
user@host> put installation-package
```

Following is an example of an *installation-package* name:
jinstall-9.2R1.8–domestic-signed.tgz

4. Close the FTP session using the **bye** command:

```
user@host> bye
Goodbye
```

- Related Documentation**
- [Downloading Software Packages from Juniper Networks on page 177](#)
 - [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
 - [Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server on page 181](#)

Downloading Software Packages from Juniper Networks

You can download Junos OS packages from the Juniper Networks website to upgrade software on your EX Series switch.

Before you begin to download software upgrades, ensure that you have a Juniper Networks Web account and a valid support contract. To obtain an account, complete the registration form at the Juniper Networks website: <https://www.juniper.net/registration/Register.jsp>.

To download software upgrades from Juniper Networks:

1. Using a Web browser, follow the links to the download URL on the Juniper Networks webpage. For EX Series, there are not separate software packages for Canada, the U.S., and other locations. Therefore, select **Canada and U.S. Version** regardless of your location:

<https://www.juniper.net/support/downloads/junos.html>

2. Log in to the Juniper Networks authentication system using the username (generally your e-mail address) and password supplied by Juniper Networks representatives.
3. Using the J-Web interface or the CLI, select the appropriate software package for your application.
4. Download the software to a local host or to an internal software distribution site.

- Related Documentation**
- [Installing Software on EX Series Switches \(J-Web Procedure\) on page 66](#)
 - [Installing Software on an EX Series Switch with a Single Routing Engine \(CLI Procedure\)](#)
 - [Understanding Software Installation on EX Series Switches on page 44](#)

Understanding Download Manager for SRX Series Devices

This topic includes the following sections:

- [Overview on page 52](#)
- [Using Download Manager to Upgrade Junos OS on page 52](#)

- [Handling Errors on page 53](#)
- [Considerations on page 53](#)

Overview

This download manager feature facilitates download of large files over low-bandwidth links. It enables you to download large Junos OS packages over low-bandwidth/flaky links so that the system can be upgraded. This feature allows you to download multiple files while monitoring their status and progress individually. It takes automatic action when required and displays status information when requested.

This feature is supported on SRX300, SRX320, SRX340, and SRX345 devices.

This feature provides the following functions:

- Bandwidth-limited downloads
- Scheduled downloads
- Automatic resume on error
- Automatic resume on reboot



NOTE: This feature supports only the FTP and HTTP protocols.

Using Download Manager to Upgrade Junos OS

The download manager acts as a substitute for the FTP utility. You can use the download manager CLI commands for all the functions where you previously used the FTP utility.

The download manager requires the following:

- FTP or HTTP server with a Junos OS image
- Server that is reachable from the device being upgraded

The download manager consists of the following CLI commands:

1. To download the Junos OS image to your device, use the **request system download start** command (set a bandwidth limit, if required). The file is saved to the `/var/tmp` directory on your device.

You can continue to use the device while the download runs in the background.

2. Use the **show system download** command to verify that the file has been downloaded. The command displays the state as "completed" when the downloaded file is ready to be installed.
3. Use the **request system software add** command to install the downloaded image file from the `/var/tmp` directory.

Handling Errors

If you encounter any problem with a download, use the **show system download id** command to obtain details about the download.

Table 10 on page 53 lists the output fields for the **show system download** command. Use this information to diagnose problems. Output fields are listed in the approximate order in which they appear.

Table 10: show system download Output Fields

Output Field	Description
Status	State of the download.
Creation Time	Time the start command was issued.
Scheduled Time	Time the download was scheduled to start.
Start Time	Time the download actually started (if it has already started).
Retry Time	Time for next retry (if the download is in the error state).
Error Count	Number of times an error was encountered by this download.
Retries Left	Number of times the system will retry the download automatically before stopping.
Most Recent Error	Message indicating the cause of the most recent error.

Considerations

- When no download limit is specified for a specific download or for all downloads, a download uses all available network bandwidth.
- Because the download limit that you set indicates an average bandwidth limit, it is possible that certain bursts might exceed the specified limit.
- When a download from an HTTP server fails, the server returns an HTML page. Occasionally, the error page is not recognized as an error page and is downloaded in place of the Junos image file.
- Remote server logins and passwords are stored by the download manager for the duration of a download. To encrypt these credentials provided along with the login keyword, define an encryption key with the **request system set-encryption-key** command. Any changes to encryption settings while download is in progress can cause the download to fail.
- A download command issued on a particular node in a chassis cluster takes place only on that node and is not propagated to the other nodes in the cluster. Downloads on different nodes are completely independent of each other. In the event of a failover, a download continues only if the server remains reachable from the node from which

the command was issued. If the server is no longer reachable on that node, the download stops and returns an error.

Related Documentation • [Installation Type Overview on page 41](#)

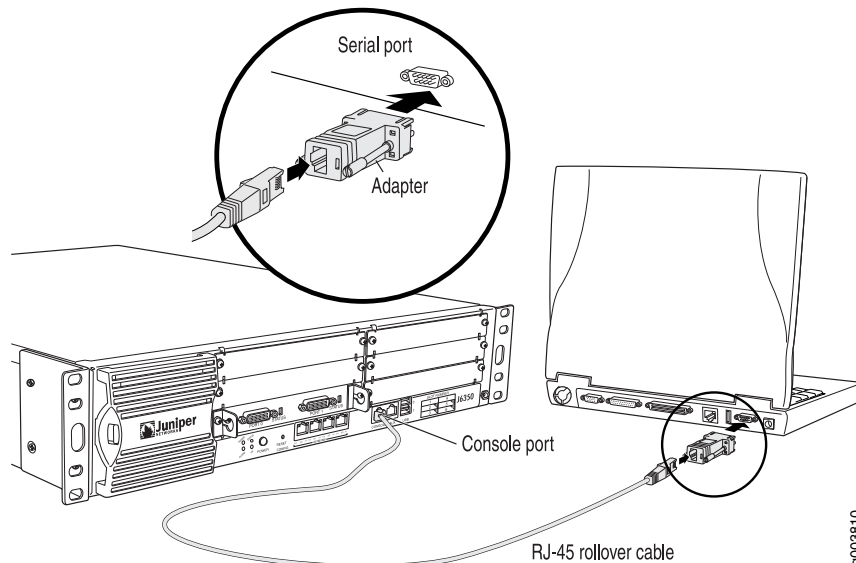
Understanding the Console Port

Console ports allow root access to the Junos operating system (Junos OS) devices through a terminal or laptop interface, regardless of the state of the Junos OS device, unless it is completely powered off. By connecting to the console port, you can access the root level of the Junos OS device without using the network to which the device might or might not be connected. This creates a secondary path to the Junos OS device without relying on the network.

Using the terminal interface provides a technician sitting in a Network Operations Center a long distance away the ability to restore a Junos OS device or perform an initialization configuration securely, using a modem, even if the primary network has failed. Without a connection to the console port, a technician would have to visit the site to perform repairs or initialization. A remote connection to the Junos OS device through a modem requires the cable and connector (provided in the device accessory box), plus a DB-9 male to DB-25 male (or similar) adapter for your modem, which you must purchase separately. For more information about connecting to the console port, see the administration guide for your particular router or switch.

To configure the device initially, you must connect a terminal or laptop computer to the device through the console port, as shown in [Figure 9 on page 54](#).

Figure 9: Connecting to the Console Port on a Junos OS Device



- Related Documentation**
- *Initial Configuration Overview for Junos OS Devices*
 - [Accessing a Junos OS Device the First Time on page 55](#)

Accessing a Junos OS Device the First Time

When you power on a Junos OS device the first time, Junos OS automatically boots and starts.

To configure the device initially, you must connect a terminal or laptop computer to the device through the console port—a serial port on the front of the router. Only console access to the device is enabled by default. Remote management access to the router and all management access protocols, including Telnet, FTP, and SSH, are disabled by default.

To access a Junos OS device the first time:

1. Connect a terminal or laptop computer to the Junos OS device through the console port—a serial port on the front of the device.

2. Power on the device and wait for it to boot.

Junos OS boots automatically. The boot process is complete when you see the **login:** prompt on the console.

3. Log in as the user **root**.

Initially, the root user account requires no password. You can see that you are the **root** user, because the prompt on the device shows the username **root@%**.

4. Start the Junos OS command-line interface (CLI).

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

5. Enter Junos OS configuration mode.

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

- Related Documentation**
- *Initial Configuration Overview for Junos OS Devices*
 - *Understanding the Root Password*
 - [Protecting Network Security by Configuring the Root Password on page 239](#)
 - *Recovering the Root Password*

Backing Up the Existing Installation on Routers

The installation process removes some files stored on the router. In the case of Junos OS, all stored files except the **juniper.conf** and SSH files are removed. Therefore, you must back up your existing installation in case you need to return to it. This topic describes how to back up the existing Junos OS installation on routers.

As of Junos OS Release 15.1, certain platforms run Junos OS based on an upgraded FreeBSD kernel (Junos OS with upgraded FreeBSD). For information about backing up Junos OS with upgraded FreeBSD, see [“Upgrading Junos OS with Upgraded FreeBSD” on page 153](#). For platforms using Junos OS with upgraded FreeBSD, see [“Understanding Junos OS with Upgraded FreeBSD” on page 17](#).

For information about backing up the existing installation on SRX Series Services Gateways, see [“Backing Up the Current Installation on SRX Series Devices” on page 57](#).

On routers, you should back up the current installation so that you can return to it if needed.

In a dual Routing Engine system, you need to back up both Routing Engines.

To back up files to the router’s hard disk or solid-state drive (SSD):

- Issue the **request system snapshot** CLI operational command:

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

When the **request system snapshot** command is issued, the **/root** file system is backed up to **/altroot**, and **/config** is backed up to **/altconfig**. The **/root** and **/config** file systems are on the router’s CompactFlash card, and the **/altroot** and **/altconfig** file systems are on the router’s hard disk or SSD. When the backup is completed, the current and backup software installations are identical.



NOTE: On routers without a CompactFlash card, where the hard disk is the primary boot device, you cannot back up your software installation. On MX104 routers, which do not have a CompactFlash card, you can back up your software installation on an external USB storage media device.

To back up files on an MX104 to a specified external storage media device:

- Issue the **request system snapshot media** CLI operational command. For example:

```
user@host > request system snapshot media usb1
```

On MX104 routers, when you issue the **request system snapshot** operational command to back up the current software installation, the backup is done on the first USB storage media device.

To back up files to the router's NAND flash device on ACX Series routers:

- Issue the **request system snapshot slice alternate** CLI operational command:

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

When this command is issued, the **/root** file system is backed up to **/altroot**, and **/config** is backed up to **/altconfig** on the router's NAND flash device.

To back up files from the NAND flash device to a USB storage media device:

- Issue the **request system snapshot** CLI operational command:

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

When you issue the **request system snapshot** operational command to back up the NAND flash device, the backup is done on the first USB storage media device.

**Related
Documentation**

- [request system snapshot on page 406](#)
- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)
- [Upgrading Junos OS with Upgraded FreeBSD on page 153](#)
- [Backing Up the Current Installation on SRX Series Devices on page 57](#)

Backing Up the Current Installation on SRX Series Devices

This topic includes the following sections:

- [Backing Up the Current Installation on SRX5800, SRX5600, and SRX5400 Devices on page 57](#)
- [Backing Up the Current Installation on SRX300, SRX320, SRX340, SRX345, and SRX550M Devices on page 58](#)
- [Configuring External CompactFlash for SRX650 Devices on page 58](#)

Backing Up the Current Installation on SRX5800, SRX5600, and SRX5400 Devices

Back up the current installation so that you can return to the current software installation. The installation process using the installation package (jinstall*, for example) removes all stored files on the device except the juniper.conf and SSH files. Therefore, you should back up your current configuration in case you need to return to the current software installation after running the installation program.

To back up Junos OS on the SRX Series devices, issue the request system snapshot CLI operational command. This command saves the current software installation on the hard disk, external USB storage media device, or solid-state drive (SSD).

When the **request system snapshot** command is issued, the **/root** file system is backed up to **/altroot**, and **/config** is backed up to **/altconfig**. The **/root** and **/config** file systems are on the device's CompactFlash card, and the **/altroot** and **/altconfig** file systems are

on the device's hard disk or solid-state drive (SSD). When the backup is completed, the current and backup software installations are identical.

To copy the files to the device's hard disk or solid-state drive (SSD), use the following command:

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

Backing Up the Current Installation on SRX300, SRX320, SRX340, SRX345, and SRX550M Devices

On SRX Series devices, you can backup the current Junos OS image and configuration files onto a media (such as a USB or CompactFlash) so that you can retrieve it back if something goes wrong.

To back up the currently running and active file system partitions on the device, use the following command:

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

Following options are supported:

- **internal**— Copies the snapshot to internal media.
- **usb**— Copies the snapshot to the USB storage device. This is the default option for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
- **external**— Copies the snapshot to an external storage device. This option is available for the compact flash on the SRX650 Services Gateway only.

Configuring External CompactFlash for SRX650 Devices

Following procedure shows how to backup current installation on an SRX650 device.

The SRX650 Services Gateway includes the following 2 GB CompactFlash (CF) storage device:

- The Services and Routing Engine (SRE) contains a hot-pluggable external CF storage device used to upload and download files.
- The chassis contains an internal CF used to store the operating system.

By default, only the internal CF is enabled and an option to take a snapshot of the configuration from the internal CF to the external CF is not supported. This can be done only by using a USB storage device.

To take a snapshot of the configuration from the external CF:

1. Take a snapshot from the internal CF to a USB storage device using the **request system snapshot media usb** command.
2. Reboot the device from the USB storage device using the **request system reboot media usb** command.

3. Go to the U-boot prompt.
4. Stop at U-boot and set the following variables:

```
set ext.cf.pref 1
save
reset
```

5. Once the system is booted from the USB storage device, take a snapshot from the external CF using the **request system snapshot media external** command.



NOTE: Once the snapshot is taken on the external CF, we recommend that you set the `ext.cf.pref` to 0 at the U-boot prompt.

Related Documentation

- [Understanding Junos OS Upgrades for SRX Series Devices on page 160](#)
- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 201](#)
- [Installing Junos OS on SRX Series Devices from the Boot Loader Using a TFTP Server on page 79](#)
- [Installing Junos OS on SRX Series Devices from the Boot Loader Using a USB Storage Device on page 82](#)

Installing the Software Package on a Router with a Single Routing Engine

To upgrade the router or switch software:

1. Install the new software package using the **request system software add** command:

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

installation-package is the name of the installation package; for example `jinstall-9.2R1.8-domestic-signed.tgz`.

For M Series, MX Series, and T Series routers running Junos OS Release 12.2 and later, you can use the **request system software add set** command to install multiple software packages at one time:

```
user@host> request system software add set /var/tmp/installation-package
```

installation-package can either be a list of installation packages, each separated by a blank space, or the full URL to the directory or tar file containing the list of installation packages.



WARNING: Do not include the `re0 | re1` option when you install a package using the **request system software add** command, if the Routing Engine on which the package is located and the Routing Engine on which you want to install the package are the same. In such cases, the package gets deleted after a successful upgrade.

For more information about the **request system software add** command, see the [CLI Explorer](#).

2. Reboot the device to start the new software using the **request system reboot** command:

```
user@host> request system reboot
Reboot the system? [yes, no] (no) yes
```



NOTE: You must reboot the device to load the new installation of Junos OS on the device.

To abort the installation, do not reboot the device. Instead, finish the installation and then issue the **request system software delete jinstall** command. This is your last chance to stop the installation.

The software is loaded when you reboot the system. Installation can take between 5 and 10 minutes. The device then reboots from the boot device on which the software was just installed. When the reboot is complete, the device displays the login prompt.

While the software is being upgraded, the Routing Engine on which you are performing the installation does not route traffic.

3. Log in and issue the **show version** command to verify the version of the software installed.
4. (Optional) Add the **jweb** package using the **request system software add** command. Before you can add this package, you must first download the software as you did the installation package. For more information about downloading the **jweb** package, see [“Downloading Software” on page 48](#).

The **jweb** installation module adds a device management graphical user interface that you can use to view and configure your device. For more information about the **jweb** package, see [“Installation Modules” on page 12](#).

5. After you have upgraded or downgraded the software and are satisfied that the new software is successfully running, issue the **request system snapshot** command to back up the new software.

Related Documentation

- [Repartitioning Routing Engine System Storage to Increase the Swap Partition on page 66](#)

Installing the Software Package on a Router with Redundant Routing Engines

If the router has two Routing Engines, perform a Junos OS installation on each Routing Engine separately to avoid disrupting network operation.



WARNING: If graceful Routing Engine switchover (GRES) or nonstop active routing (NSR) is enabled when you initiate a software installation, the software does not install properly. Make sure you issue the CLI `delete chassis redundancy` command when prompted. If GRES is enabled, it will be removed with the `redundancy` command. By default, NSR is disabled. If NSR is enabled, remove the nonstop-routing statement from the [edit routing-options] hierarchy level to disable it.

Before you begin, download the software installation package you want to install and place it on the primary Routing Engine (in slot 0) in the `/var/tmp` directory.

To upgrade the router software, perform the following tasks:

1. [Preparing the Router for the Installation on page 61](#)
2. [Installing Software on the Backup Routing Engine on page 62](#)
3. [Installing Software on the Remaining Routing Engine on page 63](#)
4. [Finalizing the Installation on page 65](#)

Preparing the Router for the Installation

Perform the following steps before installing the software:

1. Log in to the primary Routing Engine's console.

For more information about logging in to the Routing Engine through the console port, see the specific hardware guide for your router.

2. From the router command line, enter configuration mode:

```
user@host#> configure
Entering configuration mode
```

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

3. Disable Routing Engine redundancy:

```
[edit]
user@host# delete chassis redundancy
```

4. Disable nonstop-bridging:

```
[edit]
user@host# delete protocols layer2-control nonstop-bridging
```

5. Save the configuration change on both Routing Engines:

```
[edit]
user@host# commit synchronize
```

6. Exit the CLI configuration mode:

```
[edit]
user@host# exit
```

Installing Software on the Backup Routing Engine

After the router has been prepared, you first install the new Junos OS release on the backup Routing Engine, while keeping the currently running software version on the primary Routing Engine. This enables the primary Routing Engine to continue operations, minimizing disruption to your network.

1. Log in to the console port on the current primary Routing Engine in slot 0.

For more information about logging in to the Routing Engine through the console port, see [“Accessing a Junos OS Device the First Time” on page 55](#) or the specific hardware guide for your router.

2. Install the new software package on the backup Routing Engine using the **request system software add** command:

```
user@host> request system software add re1 validate  
/var/tmp/jinstall-9.2R1.8-domestic-signed.tgz
```



WARNING: Do not include the `re0` or `re1` option when you install a package using the **request system software add** command, if the Routing Engine on which the package is located and the Routing Engine on which you want to install the package are the same. In such cases, the package gets deleted after a successful upgrade.

For M Series, MX Series, and T Series routers running Junos OS Release 12.2 and later, you can use the **request system software add set** command to install multiple software packages at the same time:

```
user@host> request system software add set re1 /var/tmp/installation-package
```

For more information about the **request system software add set** command, see [request system software add](#) or the [CLI Explorer](#).

3. Reboot the router to start the new software using the **request system reboot** command:

```
user@host> request system reboot  
Reboot the system? [yes, no] (no) yes
```



NOTE: You must reboot the device to load the new installation of Junos OS on the router.

To abort the installation, do not reboot your device. Instead, finish the installation and then issue the **request system software delete jinstall** command. This is your last chance to stop the installation.

All the software is loaded when you reboot the device. Installation can take between 5 and 10 minutes. The router then reboots from the boot device on which the software was just installed. When the reboot is complete, the router displays the login prompt.

While the software is being upgraded, the Routing Engine on which you are performing the installation is not routing traffic.

4. Log in to the primary Routing Engine (slot 0) and issue the **show version** command to verify the version of the software installed.
5. (Optional) Add the **jweb** package using the **request system software add** command. Before you can add this package, you must first download the software as you did the installation package. For more information about downloading the **jweb** package, see [“Downloading Software” on page 48](#).

The **jweb** installation module adds a router management graphical user interface that you can use to view and configure your router. For more information about the **jweb** package, see [“Installation Modules” on page 12](#).

Installing Software on the Remaining Routing Engine

Once the software is installed on the backup Routing Engine, you are ready to switch routing control to the backup Routing Engine, and then upgrade or downgrade the software on the remaining Routing Engine in slot 0:

1. Transfer routing control from the primary to the backup Routing Engine:

```
user@host> request chassis routing-engine master switch
```

For more information about the **request chassis routing-engine master** command, see the [CLI Explorer](#).

2. Verify that the Routing Engine in slot 1 is now the primary Routing Engine:

```
user@host> show chassis routing-engine
Routing Engine status:
Slot 0:
  Current state           Backup
  Election priority       Master (default)
Routing Engine status:
Slot 1:
  Current state           Master
  Election priority       Backup (default)
```

3. Install the new software package on the Routing Engine in slot 0 using the **request system software add** command:

```
user@host> request system software add validate re0
/var/tmp/jinstall-9.2R1.8-domestic-signed.tgz
```



WARNING: Do not include the **re0** or **re1** option when you install a package using the **request system software add** command, if the Routing Engine on which the package is located and the Routing Engine on which you want to install the package are the same. In such cases, the package gets deleted after a successful upgrade.

For M Series, MX Series, and T Series routers running Junos OS Release 12.2 and later, you can use the **request system software add set** command to install multiple software packages at the same time:

```
user@host> request system software add set re0 /var/tmp/installation-package
```

For more information about the **request system software add set** command, see [request system software add](#) or the [CLI Explorer](#).

4. Reboot the Routing Engine using the **request system reboot** command:

```
user@host> request system reboot
Reboot the system? [yes, no] (no) yes
```



NOTE: You must reboot to load the new installation of Junos OS on the router.

To abort the installation, do not reboot your system. Instead, finish the installation and then issue the **request system software delete jinstall** command. This is your last chance to stop the installation.

The software is loaded when you reboot the system. Installation can take between 5 and 10 minutes. The router then reboots from the boot device on which the software was just installed. When the reboot is complete, the router displays the login prompt.

While the software is being upgraded, the Routing Engine on which you are performing the installation does not route traffic.

5. Log in to the current primary Routing Engine (slot 1) and issue the **show version** command to verify the version of the software installed.
6. (Optional) Add the **jweb** package using the **request system software add** command. Before you can add this package, you must first download the software as you did the installation package. For more information about downloading the **jweb** package, see [“Downloading Software” on page 48](#).

The **jweb** installation module adds a router management graphical user interface that you can use to view and configure your router. For more information about the **jweb** package, see [“Installation Modules” on page 12](#).

7. Transfer routing control back to the original primary Routing Engine in slot 0:

```
user@host> request chassis routing-engine master switch
```

For more information about the **request chassis routing-engine master** command, see the [CLI Explorer](#).

8. Verify that the Routing Engine (slot 0) is indeed the primary Routing Engine:

```
user@host> show chassis routing-engine
Routing Engine status:
Slot 0:
  Current state           Master
  Election priority       Master (default)
Routing Engine status:
Slot 1:
  Current state           Backup
  Election priority       Backup (default)
```

Finalizing the Installation

Once the software is installed on both Routing Engines, you return the router back to its original configuration and back up the new installation.

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

```
{backup}
user@host-re0> configure
[edit]
user@host-re0# rollback 1
```



NOTE: The number on the rollback command should match the number of commits you did in preparing the router for the installation. For example, if you did a separate commit for disabling Routing Engine redundancy and disabling nonstop-bridging, you need to use rollback 2 in this step.

2. Save the configuration change on both Routing Engines:

```
[edit]
user@host-re0> commit synchronize and-quit
```

3. After you have installed the new software and are satisfied that it is successfully running, back up the new software on both primary and backup Routing Engines.

For backing up Junos OS with upgraded FreeBSD, see [“Understanding Junos OS with Upgraded FreeBSD Snapshots” on page 23](#).

For Junos OS, use the **request system snapshot** command:

```
{master}
user@host-re0> request system snapshot
{master}
user@host-re0> request routing-engine login other routing-engine
{backup}
user@host-re1> request system snapshot
{backup}
```

The root file system is backed up to **/altroot**, and **/config** is backed up to **/altconfig**. The root and **/config** file systems are on the router’s CompactFlash card, and the **/altroot** and **/altconfig** file systems are on the router’s hard disk or solid-state drive (SSD).

For more information about the **request routing-engine login** command, see the [CLI Explorer](#).



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

- Related Documentation**
- [Repartitioning Routing Engine System Storage to Increase the Swap Partition on page 66](#)

Repartitioning Routing Engine System Storage to Increase the Swap Partition

You can increase the size of the swap partition by repartitioning the drive (hard disk or solid-state drive [SSD]) on the Routing Engine. This feature is first available in Junos OS Release 10.4R5, 11.1R3, and 11.2R1; in earlier Junos OS releases, the swap partition is not increased by the methods described here.

This behavior applies only to Routing Engines with more than 2 GB of RAM. The new size of the swap partition depends on the size of the drive and the amount of Routing Engine RAM.

- When the drive is 32 GB or less, the swap partition is limited to 8 GB.
- When the drive is larger than 32 GB, the swap partition matches the size of the Routing Engine RAM.

To repartition the drive, perform one of the following actions:

- During the installation of a Junos OS software package (**jinstall***), issue the **request system reboot media disk** command to boot from the drive instead of issuing the **request system reboot** command. The drive is automatically repartitioned. The **request system reboot media disk** command repartitions the drive only during a software upgrade.
- Manually partition the drive by issuing the **request system partition hard-disk** command, and then reboot the router when the command completes.



CAUTION: Repartitioning the drive re-creates the `/config` and `/var` directories in the router file system. Although the contents of `/config` and `/var/db` are preserved, the remaining contents of `/var` are lost. For this reason, we recommend that you back up the `/var` directory before you repartition the SSD on a router with this configuration.

- Related Documentation**
- [Installing the Software Package on a Router with a Single Routing Engine on page 59](#)
 - [Installing the Software Package on a Router with Redundant Routing Engines on page 60](#)

Installing Software on EX Series Switches (J-Web Procedure)

You can upgrade software packages on a single, fixed-configuration switch, on an individual member of a Virtual Chassis, or for all members of a Virtual Chassis.

You can use the J-Web interface to install software upgrades from a server using FTP or HTTP, or by copying the file to the EX Series switch.

This topic describes:

1. [Installing Software Upgrades from a Server on page 67](#)
2. [Installing Software Upgrades by Uploading Files on page 68](#)

Installing Software Upgrades from a Server

To install software upgrades from a remote server by using FTP or HTTP:

1. Download the software package as described in [“Downloading Software Packages from Juniper Networks” on page 51](#).
2. Log in to the Juniper Networks authentication system using the username (generally your e-mail address) and password supplied by Juniper Networks representatives.
3. In the J-Web interface, select **Maintain > Software > Install Package**.
4. On the Install Remote page, enter information into the fields described in [Table 11 on page 67](#).
5. Click **Fetch and Install Package**. The software is activated after the switch has rebooted.

Table 11: Install Remote Summary

Field	Function	Your Action
Package Location (required)	Specifies the FTP or HTTP server, file path, and software package name.	Type the full address of the software package location on the FTP or HTTP server—one of the following: <i>ftp://hostname/pathname/package-name</i> <i>http://hostname/pathname/package-name</i>
User	Specifies the username, if the server requires one.	Type the username.
Password	Specifies the password, if the server requires one.	Type the password.
Reboot If Required	<p>NOTE: The Reboot check box will be disabled if you enter a J-Web Application package name in the Package Location text box. To enable the Reboot check box, enter a Junos OS package name in the Package Location text box.</p> <p>If this box is checked, the switching platform will automatically reboot when the upgrade is complete.</p>	Check the box if you want the switching platform to reboot automatically when the upgrade is complete.

Installing Software Upgrades by Uploading Files

To install software upgrades by uploading files:

1. Download the software package.
2. In the J-Web interface, select **Maintain>Software>Upload Package**.
3. On the Upload Package page, enter information into the fields described in [Table 12 on page 68](#).
4. Click **Upload and Install Package**. The software is activated after the switching platform completes the installation procedure.

Table 12: Upload Package Summary

Field	Function	Your Action
File to Upload (required)	Specifies the location of the software package.	Type the location of the software package, or click Browse to navigate to the location.
Reboot If Required	Specifies that the switching platform is automatically rebooted when the upgrade is complete.	Select the check box if you want the switching platform to reboot automatically when the upgrade is complete.

- Related Documentation**
- *Installing Software on an EX Series Switch with a Single Routing Engine (CLI Procedure)*
 - [Understanding Software Installation on EX Series Switches on page 44](#)
 - [Troubleshooting Software Installation on page 329](#)

Registering the EX Series Switch with the J-Web Interface



NOTE: This topic applies only to the J-Web Application package.

You can register your EX Series switch with the J-Web interface so that you can request technical assistance as and when required. To register an EX Series switch:

1. In the J-Web interface, select **Maintain > Customer Support > Product Registration**. For an EX8200 Virtual Chassis configuration, select the member from the list.
Note the serial number that is displayed.
2. Click **Register**. Enter the serial number in the page that is displayed.

- Related Documentation**
- *EX Series Switch Software Features Overview*

Connecting and Configuring an EX Series Switch (CLI Procedure)

There are two ways to connect and configure an EX Series switch: one method is through the console by using the CLI and the other is by using the J-Web interface.



NOTE: EX2200-24T-4G-DC switches do not support switch connection and configuration through the J-Web interface.

This topic describes the CLI procedure.



NOTE: To run the `ezsetup` script, the switch must have the factory-default configuration as the active configuration. If you have configured anything on the switch and want to run `ezsetup`, revert to the factory-default configuration. See *Reverting to the Default Factory Configuration for the EX Series Switch*.

Using the CLI, set the following parameter values in the console server or PC:

- Baud rate—9600
- Flow control—None
- Data—8
- Parity—None
- Stop bits—1
- DCD state—Disregard

To connect and configure the switch from the console by using the CLI:

1. Connect the console port to a laptop or PC by using the RJ-45 to DB-9 serial port adapter. An Ethernet cable that has an RJ-45 connector at either end and an RJ-45 to DB-9 serial port adapter are supplied with the switch.

For the location of the console port on different EX Series switches:

- See *EX2200 Switches Hardware Overview*.
- See *EX2300 Switches Hardware Overview*.
- See *Rear Panel of an EX3200 Switch*.
- See *Rear Panel of an EX3300 Switch*.
- See *Rear Panel of an EX3400 Switch*.
- See *Rear Panel of an EX4200 Switch*.
- See *EX4300 Switches Hardware Overview*.
- See *Front Panel of an EX4500 Switch*.
- See *EX4550 Switches Hardware Overview*.
- See *Switch Fabric and Routing Engine (SRE) Module in an EX6200 Switch*.
- See *Switch Fabric and Routing Engine (SRE) Module in an EX8208 Switch*.
- See *Routing Engine (RE) Module in an EX8216 Switch*.



NOTE: In EX2200-C, EX2300, EX3400, EX4300, and EX4550 switches, you can also use the Mini-USB Type-B console port to connect to a laptop or PC. See *Connecting an EX Series Switch to a Management Console by Using the Mini-USB Type-B Console Port*.

2. At the Junos OS shell prompt **root%**, type **ezsetup**.
3. Enter the hostname. This is optional.
4. Enter the root password you want to use for the device. Reenter the root password when prompted.
5. Enable services such as SSH and Telnet.



NOTE: You will not be able to log in to the switch as the **root** user through Telnet. Root login is allowed only through SSH.

- The default option for SSH is **yes**. Select this to enable SSH.
- The default option for Telnet is **no**. Change this to **yes** to enable Telnet.

6. Use the Management Options page to select the management scenario:



NOTE: On EX4500, EX6200, and EX8200 switches, only the out-of-band management option is available.

- *Configure in-band management.* In in-band management, you configure a network interface or an uplink module (expansion module) interface as the management interface and connect it to the management device. In this scenario, you have the following two options:
 - Use the automatically created VLAN *default* for management—Select this option to configure all data interfaces as members of the default VLAN. Specify the management IP address and the default gateway.
 - Create a new VLAN for management—Select this option to create a management VLAN. Specify the VLAN name, VLAN ID, management IP address, and default gateway. Select the ports that must be part of this VLAN.
 - *Configure out-of-band management.*—Configure the management port. In out-of-band management, you use a dedicated management channel (**MGMT** port) to connect to the management device. Specify the IP address and gateway of the management interface. Use this IP address to connect to the switch.
7. Specify the SNMP read community, location, and contact to configure SNMP parameters. These parameters are optional.
 8. Specify the system date and time. Select the time zone from the list. These options are optional.
 9. The configured parameters are displayed. Enter **yes** to commit the configuration. The configuration is committed as the active configuration for the switch.
 10. (For EX4500 switches only) Enter the operational mode command **request chassis pic-mode intraconnect** to set the PIC mode to intraconnect.

You can now log in with the CLI or the J-Web interface to continue configuring the switch. If you use the J-Web interface to continue configuring the switch, the Web session is redirected to the new management IP address. If the connection cannot be made, the J-Web interface displays instructions for starting a J-Web session.

Related Documentation

- *Connecting and Configuring an EX Series Switch (J-Web Procedure)*
- *Installing and Connecting an EX2200 Switch*
- *Installing and Connecting an EX2300 Switch*
- *Installing and Connecting an EX3200 Switch*
- *Installing and Connecting an EX3300 Switch*
- *Installing and Connecting an EX3400 Switch*

- *Installing and Connecting an EX4200 Switch*
- *Installing and Connecting an EX4300 Switch*
- *Installing and Connecting an EX4550 Switch*
- *Installing and Connecting an EX4500 Switch*
- *Installing and Connecting an EX6210 Switch*
- *Installing and Connecting an EX8208 Switch*
- *Installing and Connecting an EX8216 Switch*

Connecting and Configuring an EX9200 Switch (CLI Procedure)

The EX9200 switch is shipped with the Junos OS preinstalled and ready to be configured when the switch is powered on. There are three copies of the software: one on a CompactFlash card in the Routing Engine module (RE module), one on a rotating hard disk in the RE module, and one on a USB flash drive that can be inserted into the slot in the faceplate of the RE module.

When the switch boots, it first attempts to start the image on the USB flash drive. If there is no USB flash drive inserted into the RE module or if the attempt otherwise fails, the switch next attempts to start the software from the CompactFlash card (if installed), and finally from the hard disk.

You configure the switch by issuing Junos OS command-line interface (CLI) commands, either on a console device attached to the console (**CONSOLE**) port on the master RE module, or over a telnet connection to a network connected to the Ethernet management (<...>) port on the master RE module.

Gather the following information before configuring the switch:

- Name the switch will use on the network
- Domain name the switch will use
- IP address and prefix length information for the Ethernet interface
- IP address of a default switch
- IP address of a DNS server
- Password for the root user

This procedure connects the switch to the network, but does not enable it to forward traffic. For complete information about enabling the switch to forward traffic, including examples, see the Junos OS configuration guides.

To configure the software:

1. Verify that the switch is powered on.
2. Log in as the *root* user. There is no password.

3. Start the CLI.

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

4. Enter configuration mode.

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

5. Set the root authentication password by entering either a clear-text password, an encrypted password, or an SSH public key string (DSA or RSA).

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

or

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

or

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

or

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

6. Configure the name of the switch. If the name includes spaces, enclose the name in quotation marks (" ").

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

7. Create a user account.

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

8. Set the user account class to super-user.

```
[edit]
root@# set system login user user-name class super-user
```

9. Configure the switch's domain name.

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

10. Configure the IP address and prefix length for the switch's Ethernet interface.

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

11. Configure the IP address of a DNS server.

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

12. (Optional) Configure the static routes to remote subnets with access to the management port. Access to the management port is limited to the local subnet. For more information about static routes, see the *Junos OS Administration Library*.

```
[edit]
root@# set routing-options static route remote-subnet next-hop destination-IP retain
no-advertise
```

13. Configure the telnet service at the **[edit system services]** hierarchy level.

```
[edit]
root@# set system services telnet
```

14. (Optional) Display the configuration to verify that it is correct.

```
[edit]
root@# show
system {
  host-name host-name;
  domain-name domain-name;
  root-authentication {
    authentication-method (password | public-key);
  }
  name-server {
    address;
  }
}
interfaces {
  fxp0 {
    unit 0 {
      family inet {
        address address/prefix-length;
      }
    }
  }
}
}
```

15. Commit the configuration to activate it on the switch.

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

16. (Optional) Configure additional properties by adding the necessary configuration statements. Then commit the changes to activate them on the switch.

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

17. When you have finished configuring the switch, exit configuration mode.

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



NOTE: To reinstall Junos OS, you boot the switch from the removable media. Do not insert the removable media during normal operations. The switch does not operate normally when it is booted from the removable media.

When the switch boots from the storage media (removable media, CompactFlash card, or hard disk) it expands its search in the `/config` directory of the routing platform for the following files in the following order: **juniper.conf** (the main configuration file), **rescue.conf** (the rescue configuration file), and **juniper.conf.1** (the first rollback configuration file). When the first configuration file is found that can be loaded properly, the search ends and the file is loaded. If none of the file can be loaded properly, the routing platform does not function properly. If the switch boots from an alternate boot device, Junos OS displays a message indicating this when you log in to the switch.

- Related Documentation**
- *EX9204 Switch Hardware Overview*
 - *EX9208 Switch Hardware Overview*
 - *EX9214 Switch Hardware Overview*

Preparing the USB Flash Drive to Upgrade Junos OS on SRX Series Devices

This feature simplifies the upgrading of Junos OS images in cases where there is no console access to an SRX Series device located at a remote site. This functionality allows you to upgrade the Junos OS image with minimum configuration effort by simply copying the image onto a USB flash drive, inserting it into the USB port of the SRX Series device, and performing a few simple steps. You can also use this feature to reformat a boot device and recover an SRX Series device after boot media corruption.

All USB flash drives used on SRX Series devices must have the following features:

- USB 2.0 or later.
- Formatted with a FAT/FAT 32 or MS-DOS file system



NOTE: For the list of recommended USB drives, see Knowledge Base article [KB31622](#).



NOTE: The Junos OS package on a USB device is commonly stored in the root drive as the only file; for example, junos-srxsme-15.1X49-D30.3-domestic.tgz.



CAUTION: Any USB memory product not listed as supported for SRX Series devices has not been tested by Juniper Networks. The use of any unsupported USB memory product could expose your SRX Series device to unpredictable behavior. Juniper Networks Technical Assistance Center (JTAC) can provide only limited support for issues related to unsupported hardware. We strongly recommend that you use only supported USB flash drives.



NOTE: This feature is not supported on chassis clusters.

Before you begin:

- Copy the Junos OS upgrade image and its autoinstall.conf file to the USB device.
- Ensure that adequate space is available on the SRX Series device to install the software image.

To prepare the USB flash drive and copy the Junos OS image onto the USB flash drive:

1. Insert the USB flash drive into the USB port of a PC or laptop computer running Windows.
2. From My Computer, right-click the drive Devices with Removable Storage.
3. Format the drive with the FAT/FAT32 file system.
4. Copy the Junos OS image onto the USB device.
For the installation process to succeed, copy only one image onto the USB device. Only images named junos-srxsme* are recognized by the system.
5. Check the drive name detected in My Computer for the USB device. Open the command prompt window and type:

```
echo " " > <drive-name>:\autoinstall.conf
```

For example, if the drive detected is drive F, type `echo " " > F:\autoinstall.conf` at the command prompt. This empty file indicates to the system that the automatic installation of the Junos OS image from the USB device is supported.

6. (Optional) Create a text file named `junos-config.conf` and copy the file to the USB device. For example, the following file supports an automatic configuration update during the installation process:

```
system {
  host-name host-1;
  domain-name example.net;
  domain-search [ abc.exmaple.net example.net device1.example.net];
  root-authentication {
    encrypted-password "$ABC123"; ## SECRET-DATA
  }
}
...
routing-options {
  static {
    route 0.0.0.0/0 next-hop 10.207.31.254;
  }
}
```



NOTE: The `junos-config.conf` file is optional, and it is not necessary for the automatic installation of the Junos OS image from the USB device. You can use the `junos-config.conf` file for a backup configuration for recovery or if the existing configuration is accidentally deleted.

Related Documentation

- [Preparing Your SRX Series Device for Junos OS Upgrades on page 162](#)
- [Downloading Software Packages from Juniper Networks on page 177](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
- [Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server on page 181](#)

Installing Junos OS on SRX Series Devices Using a USB Flash Drive

To install the Junos OS image on an SRX Series device using a USB flash drive:

1. Insert the USB flash drive into the USB port of the SRX Series device and wait for the LEDs to blink amber, then steadily turn amber, indicating that the SRX Series device detects the Junos OS image.

If the LEDs do not change to amber, press the Power button or turn the device off and then on again and wait for the LEDs to blink amber.

2. Press the **Reset Config** button on the SRX Series device to start the installation and wait for the LEDs to glow steadily amber.

When the LEDs glow green, the Junos OS upgrade image has been successfully installed.

If the USB device is plugged in, the **Reset Config** button always performs as an image upgrade button. Any other functionality of this button is overridden until you remove the USB flash drive.

3. Remove the USB flash drive.

The SRX Series device restarts automatically and loads the new Junos OS version.



NOTE: On SRX300, SRX320, SRX340, SRX345, and SRX550M devices, frequent plug and play of USB keys is not supported. You must wait for the device node creation before removing the USB key.



NOTE: If an installation error occurs, the LEDs turn red, which might indicate that the Junos OS image on the USB flash drive is corrupted. An installation error can also occur if the current configuration on the SRX Series device is not compatible with the new Junos OS version on the USB or if there is not enough space on the SRX Series device to install the image. You must have console access to the SRX Series device to troubleshoot an installation error.



NOTE: You can use the `set system autoinstallation usb disable` command to prevent the automatic installation from the USB device. After using this command, if you insert the USB device into the USB port of the SRX Series device, the installation process does not work.

**Related
Documentation**

- [Preparing Your SRX Series Device for Junos OS Upgrades on page 162](#)
- [Preparing the USB Flash Drive to Upgrade Junos OS on SRX Series Devices on page 75](#)
- [Backing Up the Current Installation on SRX Series Devices on page 57](#)
- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 201](#)

Upgrading the Boot Loader on SRX Series Devices

To upgrade the boot loader to the latest version:

1. Upgrade to Junos OS Release 10.0 or later (with or without dual-root support enabled).

The Junos OS 10.0 image contains the latest boot loader binaries in this path:
`/boot/uboot, /boot/loader.`

2. Enter the shell prompt using the **start shell** command.
3. Run the following command from the shell prompt:

```
bootupgrade -u /boot/uboot -l /boot/loader
```



NOTE: For the new version to take effect, you should reboot the system after upgrading the boot loader.

To verify the boot loader version on the SRX Series device, enter the **show chassis routing-engine bios** command.

```
user@host> show chassis routing-engine bios
Routing Engine BIOS Version: 1.5
```

The command output displays the boot loader version.



NOTE: You can use the following commands to upgrade U-Boot or perform cyclic redundancy check (CRC):

- **bootupgrade -s -u** – To upgrade the secondary boot loader.
- **bootupgrade -c u-boot** – To check CRC of the boot loader.
- **bootupgrade -s -c u-boot** – To check CRC for the secondary boot loader.
- **bootupgrade -c loader** – To check CRC for the loader on boot loader.

Related Documentation

- [Preparing Your SRX Series Device for Junos OS Upgrades on page 162](#)
- [Downloading Software Packages from Juniper Networks on page 177](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
- [Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server on page 181](#)

Installing Junos OS on SRX Series Devices from the Boot Loader Using a TFTP Server

You can install Junos OS using the Trivial File Transfer Protocol (TFTP) method. The device is shipped with Junos OS loaded on the primary boot device. During Junos OS installation from the loader, the device retrieves the Junos OS package from a TFTP server. The internal media is then formatted, and the Junos OS image is installed.

From the loader installation, you can:

- Install Junos OS on the device for the first time.
- Recover the system from a file system corruption.



NOTE: Installation from a TFTP server can only be performed using the first onboard Ethernet interface.

Installation from the loader-over-TFTP method does not work reliably over slow speeds or large latency networks.

Before you begin, verify that:

- You have access to the TFTP server with the Junos OS package to be installed.
- That the TFTP server supports BOOTP or DHCP. If the TFTP server does not support BOOTP or DHCP, you must set the environment variables before performing the installation from the TFTP server.
- Functional network connectivity exists between the device and the TFTP server over the first onboard Ethernet interface.

To install the Junos OS image on the internal media of the device:

1. To access the U-boot prompt, use the console connection to connect to the device.

2. Reboot the device.

The following messages appear:

Clearing DRAM..... done BIST check passed. Net: pic init done (err = 0)octeth0 POST Passed

After this message appears, you see the following prompt:

Press SPACE to abort autoboot in 3 seconds

3. Press the space bar to stop the autoboot process.

The => U-boot prompt appears.

4. From the U-boot prompt, configure the environment variables listed in [Table 13 on page 80](#).

Table 13: Environment Variables Settings

Environment Variables	Description
gatewayip	IP address of the gateway device
ipaddr	IP address of the SRX Series device
netmask	network mask
serverip	IP address of the TFTP server

This example shows you how to configure the environment variables:

```
Clearing DRAM..... done
BIST check passed.
Net: pic init done (err = 0)octeth0
POST Passed
Press SPACE to abort autoboot in 3 seconds
=>
=> setenv ipaddr 10.157.70.170
=> setenv netmask 255.255.255.0
=> setenv gatewayip 10.157.64.1
=> setenv serverip 10.157.60.1
=> saveenv
```

5. Reboot the system using the **reset** command.
6. To access the loader prompt, use the console connection to connect to the device.
7. Reboot the device.

The following message appears:

Loading /boot/defaults/loader.conf

After this message appears, you see the following prompt:

Hit [Enter] to boot immediately, or space bar for command prompt.

8. Press the space bar to access the loader prompt (**loader>**).

The **loader>** prompt appears. Enter:

```
loader> install tftp://10.77.25.12/junos-srxsme-10.0R2-domestic.tgz
```



NOTE: The URL path is relative to the TFTP server's TFTP root directory, where the URL is *tftp://tftp-server-ipaddress/package*.

When this command is executed:

- The Junos OS package is downloaded from the TFTP server.
- The internal media on the system is formatted.
- The Junos OS package is installed on the internal media.



NOTE: The Installation from the loader-over-TFTP method installs Junos OS on the internal CF on SRX100, SRX210, SRX220, and SRX240 devices, whereas on SRX650 devices, this method can install Junos OS on the internal or external CF card.

After Junos OS is installed, the device boots from the internal media. Once the system boots up with Junos OS Release 10.0 or later, you must upgrade the U-boot and boot loader immediately.



CAUTION: When you install Junos OS using the loader-over-TFTP method, the media is formatted. The process attempts to save the current configuration. We recommend that you back up all important information on the device before using this process.

**Related
Documentation**

- [Preparing Your SRX Series Device for Junos OS Upgrades on page 162](#)
- [Preparing the USB Flash Drive to Upgrade Junos OS on SRX Series Devices on page 75](#)
- [Backing Up the Current Installation on SRX Series Devices on page 57](#)
- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 201](#)

Installing Junos OS on SRX Series Devices from the Boot Loader Using a USB Storage Device

To install Junos OS Release 10.0 or later from the boot loader using a USB storage device:

1. Format a USB storage device in MS-DOS format.
2. Copy the Junos OS image onto the USB storage device.
3. Plug the USB storage device into the SRX Series device.
4. Stop the device at the loader prompt and issue the following command:

```
loader> install file:///<image-path-on-usb>
```

An example of a command is as follows:

```
loader> install file:///junos-srxsme-10.0R2-domestic.tgz
```

This formats the internal media and installs the new Junos OS image on the media with dual-root partitioning.

5. Once the system boots up with Junos OS Release 10.0 or later, upgrade the U-boot and boot loader immediately.
6. Remove the USB flash drive.

**Related
Documentation**

- [Preparing Your SRX Series Device for Junos OS Upgrades on page 162](#)
- [Preparing the USB Flash Drive to Upgrade Junos OS on SRX Series Devices on page 75](#)
- [Backing Up the Current Installation on SRX Series Devices on page 57](#)

- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 201](#)

CHAPTER 5

Configuring Zero Touch Provisioning

- Understanding Zero Touch Provisioning on page 85
- Configuring Zero Touch Provisioning on page 89

Understanding Zero Touch Provisioning



NOTE: To see which platforms support zero touch provisioning, in a browser, go to [Feature Explorer](#). In the Explore Features section of the Feature Explorer page, select All Features. In the Features Grouped by Feature Family box, select Zero Touch Provisioning. You can also type the name of the feature in the Search for Features edit box. In previous Junos OS releases on EX Series switches, zero touch provisioning was called EZ Touchless Provisioning.

Zero touch provisioning (ZTP) allows you to provision new Juniper Networks switches and routers in your network automatically, without manual intervention. When you physically connect a switch or router to the network and boot it with a default factory configuration, it attempts to upgrade the Junos OS software automatically and autoinstall a configuration file from the network.

- As of Junos OS Release 15.2, you can provision using a script to be executed, as well as a configuration file to be loaded, for either switches or routers.
- As of Junos OS Release 15.1, you can provision using a script to be executed, as well as a configuration file to be loaded, but only for switches.
- In releases earlier than Junos OS Release 15.1, you can automatically provision switches only, and only by loading a configuration file.

The switch or router uses information that you configure on a Dynamic Host Configuration Protocol (DHCP) server to locate the necessary software image and configuration files on the network or to find the script to execute. If you do not configure the DHCP server to provide this information, the device boots with the preinstalled software and default factory configuration. To make sure you have the default factory configuration loaded on the switch or router, issue the **request system zeroize** command on the device you want to provision.

Upon connection and booting of a new router or switch, if Junos OS detects a file on the DHCP server, the first line of the file is examined. If Junos OS finds the characters **#!**

followed by an interpreter path, it treats the file as a script file and executes the script with the interpreter mentioned. If the script returns error (that is, a nonzero value), the ZTP state machine refetches the script and attempts to execute the script again. This continues until the script executes successfully. The script can be, for example, a shell script (`#!/bin/sh`), a slax script (`#!/usr/bin/slax`), or a python script (`#!/usr/bin/python`).

If Junos OS does not find the characters `#!` followed by an interpreter path, it treats the file as a Junos OS configuration in text format and loads the file.

The zero touch provisioning process will either upgrade or downgrade the Junos OS release.

There are two downgrade limitations for EX Series switches:

- If you downgrade to a software version earlier than Junos OS Release 12.2, in which zero touch provisioning is not supported, the configuration file autoinstall phase of the zero touch provisioning process does not happen.
- To downgrade to a software version that does not support resilient dual-root partitions (Junos OS Release 10.4R2 or earlier), you must perform some manual work on the switch. For more information, see [“Understanding Resilient Dual-Root Partitions on Switches” on page 111](#).



NOTE: On QFX3500 and QFX3600 switches running the original CLI, you cannot use ZTP to upgrade from Junos OS Release 12.2 or later to Junos OS Release 13.2X51-D15 or later.

When you boot a switch or router with the default factory configuration, the following process happens:

1. If DHCP option 43, suboption 00 (the name of the software image file on the FTP, HTTP, or TFTP server) is configured, the switch or router compares the version of the provided software image to the version of the software installed on the switch or router.



NOTE: When the DHCP server cannot use suboption 00, configure the image file using suboption 04. If both suboption 00 and suboption 4 are defined, suboption 04 is ignored.

2. If DHCP option 43, suboption 02 (a symbolic link to the software image file on the FTP, HTTP, or TFTP server) is configured, the switch or router compares the version of the provided software image to the version of the software installed on the switch or router.
 - If the Junos OS releases are different, the switch or router downloads the software image from the FTP, HTTP, or TFTP server, installs Junos OS, and reboots using the default factory configuration.

- If the software versions are the same, the switch or router does not upgrade the software.
3. If DHCP option 43, suboption 01 (the name of the configuration file or the script file on the FTP, TFTP, or HTTP server) is configured, the switch or router compares the version of the provided configuration file to the version of the configuration file on the switch or router.

If DHCP option 43, suboption 01 is not specified, the switch or router uses the default factory configuration.

If the file version on the FTP, HTTP, or TFTP server is newer than the file on the switch or router, the file is updated on the switch or router.

If both DHCP option 43, suboption 01 and suboption 2 are specified, suboption 01 is processed before suboption 02. Junos OS is upgraded, and then the configuration or script file is applied.

4. If DHCP option 43, suboption 03 (the transfer mode setting) is configured, the switch or router accesses the FTP, HTTP, or TFTP server using the specified transfer mode setting, for example, FTP.

If DHCP option 43, suboption 03 is not configured, TFTP becomes the transfer mode automatically.

5. If DHCP option 43, suboption 04 (the name of the software image file on the FTP, HTTP, or TFTP server) is configured, the switch or router compares the version of the provided software image to the version of the software installed on the switch or router.



NOTE: When the DHCP server cannot use suboption 00, configure the image file using suboption 04. If both suboption 00 and suboption 4 are defined, suboption 04 is ignored.

6. If DHCP option 150 or option 66 is specified, the IP address of the FTP, HTTP, or TFTP server is configured.



NOTE: You must configure either option 150 or option 66. If you configure both option 150 and option 66, option 150 takes precedence and option 66 is ignored. Also, make sure you specify an IP address, not a hostname, because name resolution is not supported.

7. (Optional) If DHCP option 7 is specified, you can configure one or more syslog servers.

8. (Optional) If DHCP option 42 is specified, you can configure one or more NTP servers.
9. (Optional) If DHCP option 12 is specified, you can configure the hostname of the switch or router.

**Related
Documentation**

- [Configuring Zero Touch Provisioning on page 89](#)

Configuring Zero Touch Provisioning



NOTE: To see which platforms support zero touch provisioning, in a browser, go to [Feature Explorer](#). In the Explore Features section of the Feature Explorer page, select All Features. In the Features Grouped by Feature Family box, select Zero Touch Provisioning. You can also type the name of the feature in the Search for Features edit box. In previous Junos OS releases on EX Series switches, zero touch provisioning was called EZ Touchless Provisioning. Search for that feature name if you want to know if this feature is supported on EX Series switches.

Zero touch provisioning (ZTP) allows you to provision new switches or routers in your network automatically, without manual intervention. You can use either management ports or network ports on your switch to connect to the network. When you physically connect a switch or router to the network and boot it with a default configuration, it attempts to upgrade the Junos OS software automatically using information detected on a Dynamic Host Control Protocol (DHCP) server. You can include this information on the DHCP server either as a script to be executed or as a configuration file to be loaded on the router or switch. The presence or absence of the characters `!#` plus a script interpreter are what determines whether the file is treated as a script or as a configuration file. If you do not configure the DHCP server to provide this information, the switch or router boots with the preinstalled software and default configuration.



NOTE: If you have both DHCP and ZTP enabled, the switch or router broadcasts a DHCP DISCOVER packet every six minutes. If a DHCP server on the network responds with a DHCP ACK packet with DHCP vendor options set with the necessary values to initiate ZTP, then ZTP proceeds.

To disable broadcasting the DHCP DISCOVER packet every six minutes, without performing the ZTP process, manually delete the `auto-image-upgrade` statement located in the `[edit chassis]` hierarchy. If ZTP completes without errors, the `auto-image-upgrade` statement is automatically deleted.



NOTE: For detailed information regarding the DHCP and DHCP options, refer to RFC 2131 (<http://www.ietf.org/rfc/rfc2131.txt>) and RFC 2132 (www.ietf.org/rfc/rfc2132.txt). Also, this document refers to Internet Systems Consortium (ISC) DHCP version 4.2. For more information regarding this version, refer to <http://www.isc.org/software/dhcp/documentation>.

Before you begin:

- Ensure that the switch or router has access to the following network resources:

- The DHCP server that provides the location of the software image and configuration files on the network

Refer to your DHCP server documentation for configuration instructions.

- The File Transfer Protocol (anonymous FTP), Hypertext Transfer Protocol (HTTP), or Trivial File Transfer Protocol (TFTP) server on which the software image and configuration files are stored



NOTE: Although TFTP is supported, we recommend that you use FTP or HTTP instead, because these transport protocols are more reliable.

- A Domain Name System (DNS) server to perform reverse DNS lookup
 - (Optional) An NTP server to perform time synchronization on the network
 - (Optional) A system log (syslog) server to manage system log messages and alerts
- Locate and record the MAC address printed on the switch or router chassis.



CAUTION: HTTP URLs are limited to 256 characters in length.



CAUTION: You cannot commit a configuration while the switch or router is performing the software update process. If you commit a configuration while the switch or router is performing the configuration file autoinstallation process, the process stops, and the configuration file is not downloaded from the network.

To configure zero touch provisioning for a switch or router:

1. Make sure the switch or router has the default factory configuration installed.

Issue the **request system zeroize** command on the switch or router that you want to provision.

2. Download the software image file and the configuration file to the FTP, HTTP, or TFTP server from which the switch or router will download these files.

You can download either one or both of these files.

3. Configure the DHCP server to provide the necessary information to the switch or router.

Configure IP address assignment.

You can configure dynamic or static IP address assignment for the management address of the switch or router. To determine the management MAC address for static IP address mapping, add 1 to the last byte of the MAC address of the switch or router, which you noted before you began this procedure.

4. Define the format of the vendor-specific information for DHCP option 43 in the **dhcpd.conf** file.

The following is an example of an ISC DHCP 4.2 server **dhcpd.conf** file:

```
option space NEW_OP;
option NEW_OP.config-file-name code 1 = text;
option NEW_OP.image-file-type code 2 = text;
option NEW_OP.transfer-mode code 3 = text;
option NEW_OP.alt-image-file-name code 4 = text;
option NEW_OP-encapsulation code 43 = encapsulate NEW_OP;
```

- 5.



NOTE: You must configure either option 150 or option 66. If you configure both option 150 and option 66, option 150 takes precedence, and option 66 is ignored. Also, make sure you specify an IP address, not a hostname, because name resolution is not supported.

Configure DHCP option 150 to specify the IP address of the FTP, HTTP, or TFTP server.

```
option option-150 code 150={ip-address};
option option-150 10.100.31.71;
```

6. Configure DHCP option 66 to specify the IP address of the FTP, HTTP, or TFTP server.

```
option tftp-server-name "10.100.31.71";
```

7. Configure the following DHCP option 43 suboptions:

- Suboption 00: The name of the software image file to install



NOTE: When the DHCP server cannot use suboption 00, configure the image file using suboption 04. If both suboption 00 and suboption 4 are defined, suboption 04 is ignored.

```
option NEW_OP.image-file-name
"/dist/images/jinstall-ex-4200-13.2R1.1-domestic-signed.tgz";
```

- Suboption 01: The name of the script or configuration file to install.

```
option NEW_OP.config-file-name "/dist/config/jn-switch35.config";
```

- Suboption 02: The symbolic link to the software image file to install.

```
option NEW_OP.image-file-type "symlink";
```



NOTE: If you do not specify suboption 2, the zero touch provisioning process handles the software image as a filename, not a symbolic link.

- Suboption 03: The transfer mode that the switch or router uses to access the TFTP, FTP, or HTTP server.

```
option NEW_OP.transfer-mode "ftp";
```



NOTE: If suboption 03 is not configured, TFTP becomes the transfer mode by default.

- Suboption 04: The name of the software image file to install.



NOTE: When the DHCP server cannot use suboption 00, configure the image file using suboption 04. If both suboption 00 and suboption 4 are defined, suboption 04 is ignored.

```
option NEW_OP.alt-image-file-name
"/dist/images/jinstall-ex-4200-13.2R1.1-domestic-signed.tgz";
```

8. (Optional) Configure DHCP option 7 to specify one or more system log (syslog) servers.

```
option log-servers 10.100.31.72;
```

9. (Optional) Configure DHCP option 42 to specify one or more NTP servers.

```
option ntp-servers 10.100.31.73;
```

10. (Optional) Configure DHCP option 12 to specify the hostname of the switch or router.

```
option hostname "jn-switch35";
```

The following sample configuration shows the DHCP options you just configured:

```
host jn-switch35 {
  hardware ethernet ac:4b:c8:29:5d:02;
  fixed-address 10.100.31.36;
  option tftp-server-name "10.100.31.71";
  option host-name "jn-switch35";
  option log-servers 10.100.31.72;
  option ntp-servers 10.100.31.73;
  option NEW_OP.image-file-name
    "/dist/images/jinstall-ex-4200-13.2R1.1-domestic-signed.tgz";
  option NEW_OP.transfer-mode "ftp";
  option NEW_OP.config-file-name "/dist/config/jn-switch35.config";
}
```

Based on the DHCP options you just configured, the following statements are appended to the Junos OS configuration file (for example, `jn-switch35.config`):

```
system {
  host-name jn-switch35;
  syslog {
    host 10.100.31.72 {
      any any;
    }
  }
  ntp {
    server 10.100.31.73;
```



```
}  
}
```

11. Connect the switch or router to the network that includes the DHCP server and the FTP, HTTP, or TFTP server.
12. Boot the switch or router with the default configuration.
13. Monitor the ZTP process by looking at the following log files:
 - `/var/log/dhcp_logfile`
 - `/var/log/image_load_log`
 - `/var/log/op-script.log`
 - `/var/log/event-script.log`

**Related
Documentation**

- [Understanding Zero Touch Provisioning on page 85](#)
- *Understanding NTP Time Servers*
- *Op Script Overview*
- *Understanding DHCP Services for Switches*
- [Reverting to the Default Factory Configuration by Using the `request system zeroize` Command on page 225](#)

CHAPTER 6

Configuring Automatic Installation of Configuration Files

- [Autoinstallation Overview on page 95](#)
- [Example: Configuring Autoinstallation on SRX Series Devices on page 100](#)
- [Configuring Autoinstallation on JNU Satellite Devices on page 103](#)
- [Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 105](#)
- [Autoinstallation of Satellite Devices in a Junos Node Unifier Group on page 107](#)
- [Verifying Autoinstallation on JNU Satellite Devices on page 108](#)

Autoinstallation Overview

If you are setting up many devices, autoinstallation can help automate the configuration process by loading configuration files onto new or existing devices automatically over the network. You can use either the J-Web configuration editor or the CLI configuration editor to configure a device for autoinstallation.

Autoinstallation provides automatic configuration for a new device that you connect to the network and turn on, or for a device configured for autoinstallation. The autoinstallation process begins any time a device is powered on and cannot locate a valid configuration file in the CompactFlash (CF) card. Typically, a configuration file is unavailable when a device is powered on for the first time, or if the configuration file is deleted from the CF card. The autoinstallation feature enables you to deploy multiple devices from a central location in the network.

For the autoinstallation process to work, you must store one or more host-specific or default configuration files on a configuration server in the network and have a service available—typically Dynamic Host Configuration Protocol (DHCP)—to assign an IP address to the device.

Autoinstallation takes place automatically when you connect an Ethernet or serial port on a new Juniper Networks device to the network and power on the device. To simplify the process, you can explicitly enable autoinstallation on a device and specify a configuration server, an autoinstallation interface, and a protocol for IP address acquisition.

This section contains the following topics:

- [Automatic Installation of Configuration Files on page 96](#)
- [Supported Autoinstallation Interfaces and Protocols on page 96](#)
- [Typical Autoinstallation Process on a New Device on page 97](#)

Automatic Installation of Configuration Files

On SRX Series devices, you can specify a remote server where configuration files are located. If a configuration file cannot be found on the device's CompactFlash card, the device automatically retrieves the configuration file from this remote server. For security purposes, you can encrypt these remote files using the DES cipher, and once they have been retrieved, the device decrypts them for use on the server.

To encrypt the files, we recommend the OpenSSL tool. You can get the OpenSSL tool at <http://www.openssl.org/>. To encrypt the file, use the following syntax:

```
% openssl enc -des -k passphrase -in original-file -out encrypted-file
```

- ***passphrase***—Passphrase used to encrypt the configuration file. The passphrase should be the name of the file without the path information or file extension.
- ***original-file***—Unencrypted configuration file.
- ***encrypted-file***—Name of the encrypted configuration file.

For example, if you are encrypting the active configuration file **juniper.conf.gz**, the passphrase is **juniper.conf**. The openssl syntax used to encrypt the file is:

```
% openssl enc -des -k juniper.conf -in juniper.conf.gz -out juniper.conf.gz.enc
```

Supported Autoinstallation Interfaces and Protocols

Before autoinstallation on a device can take place, the device must acquire an IP address. The protocol or protocols you choose for IP address acquisition determine the device interface to connect to the network for autoinstallation. The device detects the connected interface and requests an IP address with a protocol appropriate for the interface. Autoinstallation is supported over an Ethernet LAN interface or a serial LAN or WAN interface. [Table 14 on page 96](#) lists the protocols that the device can use on these interfaces for IP address acquisition.

Table 14: Interfaces and Protocols for IP Address Acquisition During Autoinstallation

Interface and Encapsulation Type	Protocol for Autoinstallation
Ethernet LAN interface with High-Level Data Link Control (HDLC)	DHCP, BOOTP, or Reverse Address Resolution Protocol (RARP)
Serial WAN interface with HDLC	Serial Line Address Resolution Protocol (SLARP)
Serial WAN interface with Frame Relay	BOOTP

If the server with the autoinstallation configuration file is not on the same LAN segment as the new device, or if a specific device is required by the network, you must configure an intermediate device directly attached to the new device through which the new device can send Trivial File Transfer Protocol (TFTP), BOOTP, and Domain Name System (DNS) requests. In this case, you specify the IP address of the intermediate device as the location to receive TFTP requests for autoinstallation.

Typical Autoinstallation Process on a New Device

When a device is powered on for the first time, it performs the following autoinstallation tasks:

1. The new device sends out DHCP, BOOTP, RARP, or SLARP requests on each connected interface simultaneously to obtain an IP address.

If a DHCP server responds, it provides the device with some or all of the following information:

- An IP address and subnet mask for the autoinstallation interface.
- The location of the TFTP (typically), Hypertext Transfer Protocol (HTTP), or FTP server on which the configuration file is stored.
- The name of the configuration file to be requested from the TFTP server.
- The IP address or hostname of the TFTP server.

If the DHCP server provides only the hostname, a DNS server must be available on the network to resolve the name to an IP address.

- The IP address of an intermediate device if the configuration server is on a different LAN segment from the new device.
2. After the new device acquires an IP address, the autoinstallation process on the device attempts to download a configuration file in the following ways:
 - a. If the DHCP server specifies the host-specific configuration file (boot file) **hostname.conf**, the device uses that filename in the TFTP server request. (In the filename, **hostname** is the hostname of the new device.) The autoinstallation process on the new device makes three unicast TFTP requests for **hostname.conf**. If these attempts fail, the device broadcasts three requests to any available TFTP server for the file.
 - b. If the new device cannot locate **hostname.conf**, the autoinstallation process unicasts or broadcasts TFTP requests for a default device configuration file called **network.conf**, which contains hostname-to-IP address mapping information, to attempt to find its hostname.
 - c. If **network.conf** contains no hostname entry for the new device, the autoinstallation process sends out a DNS request and attempts to resolve the new device's IP address to a hostname.

- d. If the new device can determine its hostname, it sends a TFTP request for the **hostname.conf** file.
 - e. If the new device is unable to map its IP address to a hostname, it sends TFTP requests for the default configuration file **router.conf**.
3. After the new device locates a configuration file on a TFTP server, autoinstallation downloads the file, installs the file on the device, and commits the configuration.



NOTE:

- If you configure the DHCP server to provide only the TFTP server hostname, add an IP address-to-hostname mapping entry for the TFTP server to the DNS database file on the DNS server in the network.
 - If the new device is not on the same network segment as the DHCP server (or other device providing IP address resolution), configure an existing device as an intermediate to receive TFTP and DNS requests and forward them to the TFTP server and the DNS server. You must configure the LAN or serial interface on the intermediate device with the IP addresses of the hosts providing TFTP and DNS service. Connect this interface to the new device.
-



NOTE: Starting in Junos OS Release 15.1X49-D60 and in Junos OS Release 17.3R1, on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices, some of the factory-default configurations are changed.

- The name-server statement, used to configure one or more Domain Name System (DNS) name servers, is changed to 8.8.8.8 and 8.8.8.4. Previously, it was 208.67.222.222 and 208.67.220.220.
- A new system service, NETCONF service over SSH, is introduced at the [edit system services] hierarchy:

```
edit system services netconf ssh
```

- The following configuration setting for HTTPS (secure management) access using the J-Web interface is changed. Now, there is no need to specify the interface details for J-Web management. With this configuration, you can manage the device from any interface through HTTPS.

```
edit system services web-management https interface [irb.0]
```

- A license autoupdate URL (https://ae1.juniper.net/junos/key_retrieval) is now supported under the [edit system] hierarchy:

```
license {
  autoupdate {
    url https://ae1.juniper.net/junos/key_retrieval;
  }
}
```

- A new system log configuration is introduced to configure system log messages to record all commands entered by users and all authentication or authorization attempts under the [edit system] hierarchy:

```
syslog {
  archive size 100k files 3;
  user * {
    any emergency;
  }
  file messages {
    any notice;
    authorization info;
  }
  file interactive-commands {
    interactive-commands any;
  }
}
```



NOTE: In Junos OS Release 15.1X49-D40 and earlier, configuring autoinstallation using USB and Layer Ethernet switching was supported on the same interface. However, the command caused improper installation of the interface-related configurations.

Starting with Junos OS Release 15.1X49-D50, Layer 2 Ethernet switching is not supported on the same interface for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

The system autoinstallation interfaces <interface names> command and the set interface <interface names> unit 0 family ethernet-switching command cannot be configured on the same interface.



NOTE: USB auto-installation is not supported on SRX1500 devices and vSRX instances.

Release History Table

Release	Description
15.1X49-D60	Starting in Junos OS Release 15.1X49-D60 and in Junos OS Release 17.3R1, on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices, some of the factory-default configurations are changed.

Related Documentation

- [Example: Configuring Autoinstallation on SRX Series Devices on page 100](#)

Example: Configuring Autoinstallation on SRX Series Devices

This example shows how to configure a device for autoinstallation.

- [Requirements on page 100](#)
- [Overview on page 101](#)
- [Configuration on page 101](#)
- [Verification on page 102](#)

Requirements

Before you begin:

- Configure a DHCP server on your network to meet your network requirements. You can configure a device to operate as a DHCP server.
- Create one of the following configuration files, and store it on a TFTP server in the network (see [“Configuration Files” on page 13](#)):
 - A host-specific file with the name **hostname.conf** for each device undergoing autoinstallation. Replace **hostname** with the name of a device. The **hostname.conf**

file typically contains all the configuration information necessary for the device with this hostname.

- A default configuration file named **router.conf** with the minimum configuration necessary to enable you to telnet into the new device for further configuration.
- Physically attach the device to the network using one or more of the following interface types:
 - Fast Ethernet
 - Gigabit Ethernet
 - Serial with HDLC encapsulation

Overview

No configuration is required on a device on which you are performing autoinstallation, because it is an automated process. However, to simplify the process, you can specify one or more interfaces, protocols, and configuration servers to be used for autoinstallation.

The device uses these protocols to send a request for an IP address for the interface.

- BOOTP—Sends requests over all interfaces.
- RARP—Sends requests over Ethernet interfaces.

Configuration

CLI Quick Configuration To quickly configure this section of the example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set system autoinstallation configuration-servers tftp://tftpconfig.sp.com
set system autoinstallation interfaces ge-0/0/0 bootp rarp
```

Step-by-Step Procedure The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure a device for autoinstallation:

1. Enable autoinstallation and specify the URL address of one or more servers from which to obtain configuration files.

```
[edit system]
user@host# set autoinstallation configuration-servers tftp://tftpconfig.sp.com
```



NOTE: You can also use an FTP address, for example, `ftp://user:password@sftpconfig.sp.com`.

2. Configure one or more Ethernet or serial interfaces to perform autoinstallation, and configure one or two procurement protocols for each interface.

[edit system]

user@host# set autoinstallation interfaces ge-0/0/0 bootp rarp

Results From configuration mode, confirm your configuration by entering the **show system autoinstallation status** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

[edit]

user@host# show system autoinstallation status

```
Autoinstallation status:
Master state: Active
Last committed file: None
Configuration server of last committed file: 10.25.100.1
Interface:
  Name: ge-0/0/0
  State: Configuration Acquisition
  Acquired:
    Address: 192.168.124.75
    Hostname: host-ge-000
    Hostname source: DNS
    Configuration filename: router-ge-000.conf
    Configuration filename server: 10.25.100.3
  Address acquisition:
    Protocol: BOOTP Client
    Acquired address: None
    Protocol: RARP Client
    Acquired address: None
```

If you are done configuring the device, enter **commit** from configuration mode.



NOTE: When there is a user-specified configuration for a particular interface, delete the factory default for that interface. Having two configurations for the same device might lead to errors. For example, if PPP encapsulation is set on a T1 interface through user configuration while the factory default configuration configures CISCO HDLC on the same interface, then the interface might not come up and the following error is logged in the message file: “DCD_CONFIG_WRITE_FAILED failed.”

Verification

Confirm that the configuration is working properly.

- [Verifying Autoinstallation on page 103](#)

Verifying Autoinstallation

- Purpose** Verify that the device has been configured for autoinstallation.
- Action** From operational mode, enter the **show system autoinstallation status** command. The output shows the settings configured for autoinstallation. Verify that the values displayed are correct for the device when it is deployed on the network.
- Related Documentation**
- [Autoinstallation Overview on page 95](#)

Configuring Autoinstallation on JNU Satellite Devices

No configuration is required on a device on which you are performing autoinstallation because it is an automated process. However, to simplify the process, you can specify one or more interfaces, protocols, and configuration servers to be used for autoinstallation. In this scenario, satellite devices, such as EX Series Ethernet Switches, QFX Series devices, and ACX Series Universal Access Routers, that are managed by the controller are considered.

To configure autoinstallation:

1. Load the JNU factory-default configuration file on the satellite device to enable the device to function in JNU mode.

```
user@satellite# load override /etc/config/jnu-factory.conf
```

An override operation discards the current candidate configuration and loads the configuration in the specified filename or the one that you type at the terminal. When you use the override option and commit the configuration, all system processes reparse the configuration.

2. Specify the URL address of one or more servers from which to obtain configuration files:

```
[edit system]
user@host# set autoinstallation configuration-servers tftp://tftpconfig.sp.com
```



NOTE: You can also use an HTTP or FTP address—for example, `http://user:password@httpconfig.sp.com` or `ftp://user:password@sftpconfig.sp.com`.

3. Configure one or more Ethernet interfaces to perform autoinstallation and IP address acquisition protocols for each interface. The router uses the protocols to send a request for an IP address for the interface:

```
[edit system]
user@host# set autoinstallation interfaces ge-0/0/0 bootp
```

4. Set the root password, entering a clear-text password that the system will encrypt, a password that is already encrypted, or an SSH public key string.

Choose one of the following:

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

```
[edit system]
user@host# set root-authentication plain-text-password
New password: type password here
Retype new password: retry password here
```

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

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

- To enter an SSH public key, use the following command:

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

5. Save the Junos OS configuration changes, activate the configuration on the device, and exit configuration mode, using the **commit-and-quit** command.

```
[edit]
user@host# commit-and-quit
```

When the satellite device reboots, it triggers the autoinstallation mechanism to retrieve its initial configuration and downloads the settings from the configuration file stored on a configuration server in the network. On the controller, you must enable the FTP service by using the **set system services ftp** command and save the configuration on the satellite device at the **/var/jnu/** directory.

The following configuration is generated on the satellite device as a result of the preceding procedure to configure autoinstallation:

```
system {
  autoinstallation {
    traceoptions {
      flags {
        all;
      }
      file autod;
      level all;
    }
    delete-after-commit; /* After initial config, no need to keep */
    interfaces {
      ge-* {
        bootp;
      }
      xe-* {
        bootp;
      }
      configuration-servers {
        "ftp://192.168.0.1/var/jnu/sat1.conf";
      }
    }
  }
}
```

```

root-authentication {
    encrypted-password "$ABC123";
}

```

Related Documentation

- [Autoinstallation of Satellite Devices in a Junos Node Unifier Group on page 107](#)
- [Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 105](#)
- [Verifying Autoinstallation on JNU Satellite Devices on page 108](#)
- [autoinstallation on page 346](#)
- [delete-after-commit \(JNU Satellites\) on page 350](#)
- [configuration-servers](#)

Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group

Autoinstallation provides automatic configuration for a new router that you connect to the network and power on, or for a router configured for autoinstallation. The autoinstallation process begins any time a router is powered on and cannot locate a valid configuration file in the CompactFlash card. Typically, a configuration file is unavailable when a router is powered on for the first time, or if the configuration file is deleted from the CompactFlash card. The autoinstallation feature enables you to deploy multiple routers from a central location in the network.

For the autoinstallation process to work, you must store one or more host-specific or default configuration files on a configuration server in the network and have a service available—typically Dynamic Host Configuration Protocol (DHCP)—to assign an IP address to the router.

Autoinstallation takes place automatically when you connect an Ethernet interface on a new Juniper Networks router to the network and power on the router. To simplify the process, you can explicitly enable autoinstallation on a router and specify a configuration server, an autoinstallation interface, and a protocol for IP address acquisition.

This topic describes:

- [Supported Autoinstallation Interfaces and Protocols on page 105](#)
- [Typical Autoinstallation Process on a New Router on page 106](#)

Supported Autoinstallation Interfaces and Protocols

Before autoinstallation on a router can take place, the router must acquire an IP address or a USB key. The protocol or protocols you choose for IP address acquisition determine the router interface to connect to the network for autoinstallation. The router detects the connected interface and requests an IP address with a protocol appropriate for the interface. Autoinstallation is supported over an Ethernet LAN interface. For IP address

acquisition, the JNU satellite router uses DHCP, BOOTP, or Reverse Address Resolution Protocol (RARP) on an Ethernet LAN interface.

If the server with the autoinstallation configuration file is not on the same LAN segment as the new router, or if a specific router is required by the network, you must configure an intermediate router directly attached to the new router, through which the new router can send HTTP, FTP, Trivial File Transfer Protocol (TFTP), BOOTP, and Domain Name System (DNS) requests. In this case, you specify the IP address of the intermediate router as the location to receive HTTP, FTP, or TFTP requests for autoinstallation.

Typical Autoinstallation Process on a New Router

When a router is powered on for the first time, it performs the following autoinstallation tasks:

1. The new router sends out DHCP, BOOTP, or RARP requests on each connected interface simultaneously to obtain an IP address.

If a DHCP server responds, it provides the router with some or all of the following information:

- An IP address and subnet mask for the autoinstallation interface.
- The location of the TFTP (typically), HTTP, or FTP server on which the configuration file is stored.
- The name of the configuration file to be requested from the HTTP, FTP, or TFTP server.
- The IP address or hostname of the HTTP, FTP, or TFTP server.

If the DHCP server provides only the hostname, a DNS server must be available on the network to resolve the name to an IP address.

- The IP address of an intermediate router if the configuration server is on a different LAN segment from the new router.
2. After the new router acquires an IP address, the autoinstallation process on the router attempts to download a configuration file in the following ways:
 - a. If the configuration file is specified as a URL, the router fetches the configuration file from the URL by using HTTP, FTP, or TFTP, depending on the protocol specified in the URL.
 - b. If the DHCP server specifies the host-specific configuration file (boot file) **hostname.conf**, the router uses that filename in the TFTP server request. (In the filename, **hostname** is the hostname of the new router.) The autoinstallation process on the new router makes three unicast TFTP requests for **hostname.conf**. If these attempts fail, the router broadcasts three requests to any available TFTP server for the file.
 - c. If the new router cannot locate **hostname.conf**, the autoinstallation process unicasts or broadcasts TFTP requests for a default router configuration file called

network.conf, which contains hostname-to-IP address mapping information, to attempt to find its hostname.

- d. If **network.conf** contains no hostname entry for the new router, the autoinstallation process sends out a DNS request and attempts to resolve the new router's IP address to a hostname.
 - e. If the new router can determine its hostname, it sends a TFTP request for the **hostname.conf** file.
 - f. If the new router is unable to map its IP address to a hostname, it sends TFTP requests for the default configuration file **router.conf**.
3. After the new router locates a configuration file on a TFTP server, the autoinstallation process downloads the file, installs the file on the router, and commits the configuration.

**Related
Documentation**

- [Autoinstallation of Satellite Devices in a Junos Node Unifier Group on page 107](#)
- [Configuring Autoinstallation on JNU Satellite Devices on page 103](#)
- [Verifying Autoinstallation on JNU Satellite Devices on page 108](#)
- [autoinstallation on page 346](#)
- [delete-after-commit \(JNU Satellites\) on page 350](#)
- [configuration-servers](#)

Autoinstallation of Satellite Devices in a Junos Node Unifier Group

In a Junos Node Unifier (JNU) group that contains an MX Series router as a controller that manages satellite devices, such as EX Series Ethernet Switches, QFX Series devices, and ACX Series Universal Access Routers, the autoinstallation functionality is supported for the satellite devices. JNU has an autoinstallation mechanism that enables a satellite device to configure itself out-of-the-box with no manual intervention, using the configuration available either on the network or locally through a removable media, or using a combination of both. This autoinstallation method is also called the *zero-touch* facility.

The zero-touch configuration delivers the following benefits:

- The router can be sent from the warehouse to the deployment site without any preconfiguration steps.
- The procedure required to deploy the device at the cell site is simplified, resulting in reduced operational and administrative costs.
- You can roll out large numbers of these devices in a very short time.

The factory default setting is autoinstallation-enabled. After you make the first configuration to the router, you can do either of the following:

- A JNU factory default file, **jnu-factory.conf**, is present in the **/etc/config/** directory and contains the configuration to perform autoinstallation on satellite devices. The zero-touch configuration can be disabled by including the **delete-after-commit** statement at the **[edit system autoinstallation]** hierarchy level and committing the configuration. This way, the saved configuration is used the next time the system reboots.
- Alternatively, if the router must get the configuration from the server each time a system reboot occurs, the zero-touch configuration must not be changed (that is, you must not include the **delete-after-commit** statement at the **[edit system autoinstallation]** hierarchy level and commit the settings).

Related Documentation

- [Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 105](#)
- [Configuring Autoinstallation on JNU Satellite Devices on page 103](#)
- [Verifying Autoinstallation on JNU Satellite Devices on page 108](#)
- [autoinstallation on page 346](#)
- [delete-after-commit \(JNU Satellites\) on page 350](#)
- [configuration-servers](#)

Verifying Autoinstallation on JNU Satellite Devices

- Purpose** After you have configured autoinstallation, display the status of autoinstallation on a satellite device, such as an ACX Series router, an EX Series switch, or a QFX Series device, in a Junos Node Unifier (JNU) group that is managed by a controller, which is an MX Series router.
- Action** From the CLI, enter the **show system autoinstallation status** command. The following example displays the autoinstallation settings of an ACX Series router that operates as a satellite in a JNU group.

Sample Output

```
user@host> show system autoinstallation status
Autoinstallation status:
  Master state: Active
  Last committed file: None
  Configuration server of last committed file: 10.25.100.1
Interface:
  Name: ge-0/1/0
  State: Configuration Acquisition
Acquired:
  Address: 192.168.124.75
  Hostname: host-ge-000
  Hostname source: DNS
  Configuration filename: router-ge-000.conf
  Configuration filename server: 10.25.100.3
```



```
Address acquisition:
  Protocol: DHCP Client
  Acquired address: None
  Protocol: RARP Client
  Acquired address: None
Interface:
  Name: ge-0/1/1
  State: None
  Address acquisition:
    Protocol: DHCP Client
    Acquired address: None
    Protocol: RARP Client
    Acquired address: None
```

Meaning The output shows the settings configured for autoinstallation. Verify that the values displayed are correct for the router when it is deployed on the network.

- Related Documentation**
- [Autoinstallation of Satellite Devices in a Junos Node Unifier Group on page 107](#)
 - [Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 105](#)
 - [Configuring Autoinstallation on JNU Satellite Devices on page 103](#)
 - [autoinstallation on page 346](#)
 - [delete-after-commit \(JNU Satellites\) on page 350](#)
 - [configuration-servers](#)
 - [show system autoinstallation status on page 477](#)

CHAPTER 7

Configuring Dual-Root Partitions for High Availability

- [Understanding Resilient Dual-Root Partitions on Switches on page 111](#)
- [Dual-Root Partitioning Scheme on SRX Series Devices on page 114](#)
- [Example: Installing Junos OS on SRX Series Devices Using the Partition Option on page 120](#)
- [Reinstalling the Single-Root Partition on SRX Series Devices on page 124](#)

Understanding Resilient Dual-Root Partitions on Switches

Resilient dual-root partitioning, introduced on Juniper Networks EX Series Ethernet Switches in Juniper Networks Junos operating system (Junos OS) Release 10.4R3, provides additional resiliency to switches in the following ways:

- Allows the switch to boot transparently from the second (alternate) root partition if the system fails to boot from the primary root partition.
- Provides separation of the root Junos OS file system from the `/var` file system. If corruption occurs in the `/var` file system (a higher probability than in the root file system because of the greater frequency of reads and writes in `/var`), the root file system is insulated from the corruption.



NOTE: For instructions on upgrading to a release that supports resilient dual-root partitions from a release that does not, see the release notes. The procedure for upgrading to a resilient dual-root partition release is different from the normal upgrade procedure.

This topic covers:

- [Resilient Dual-Root Partition Scheme \(Junos OS Release 10.4R3 and Later\) on page 112](#)
- [Automatic Fixing of Corrupted Primary Root Partition with the Automatic Snapshot Feature on page 112](#)

- [Earlier Partition Scheme \(Junos OS Release 10.4R2 and Earlier\)](#) on page 113
- [Understanding Upgrading or Downgrading Between Resilient Dual-Root Partition Releases and Earlier Releases](#) on page 114

Resilient Dual-Root Partition Scheme (Junos OS Release 10.4R3 and Later)

EX Series switches that ship with Junos OS Release 10.4R3 or later are configured with a root partition scheme that is optimized for resiliency, as shown in [Table 15 on page 112](#).

Table 15: Resilient Dual-Root Partition Scheme

Slice 1	Slice 2	Slice 3		Slice 4
s1a	s2a	s3e	s3d	s4d
/	/	/var	/var/tmp	/config
(root Junos OS)	(root Junos OS)			

In the resilient dual-root partition scheme, the **/var** file system is contained in a separate slice (Slice 3) from the root file systems, the **/config** directory is contained in its own slice (Slice 4), and switches ship from the factory with identical Junos OS images in Slice 1 and Slice 2. The **/var** file system, which has a greater frequency of reads and writes than the root file systems and is therefore more likely to have corruption issues, is isolated from the root directories and the **/config** directory. If the switch fails to boot from the active partition, the switch automatically boots from the alternate root partition and triggers an alarm.

Automatic Fixing of Corrupted Primary Root Partition with the Automatic Snapshot Feature

Resilient dual-root partitioning also provides the *automatic snapshot* feature, which allows the switch to automatically fix a corrupt Junos OS file in the primary root partition. If the automatic snapshot feature is enabled, the switch automatically takes a snapshot of the Junos OS root file system in the alternate root partition and copies it onto the primary root partition, thereby repairing the corrupt file in the primary root partition. The automatic snapshot procedure takes place whenever the system reboots from the alternate root partition, regardless of whether the reboot is due to a command or due to corruption of the primary root partition.

**NOTE:**

- EX9200 switches do not support the automatic snapshot feature.
- The automatic snapshot feature is enabled by default on the following EX Series switches:
 - EX4550 switches
 - EX Series switches that ship with Junos OS Release 12.3R1 or later
- The automatic snapshot feature is disabled by default on EX Series switches (except the EX4550 switches) running Junos OS Release 12.2 or earlier.
- If the automatic snapshot feature was disabled by default before the switch was upgraded to Junos OS Release 12.3R1 or later, the feature remains disabled (for backward compatibility) by default after the upgrade.
- If the automatic snapshot feature is enabled in a Virtual Chassis configuration, the automatic snapshot procedure takes place whenever any member of the Virtual Chassis reboots from its alternate root partition.
- You can enable the automatic snapshot feature by configuring the `auto-snapshot` statement at the `[edit system]` hierarchy level.

The automatic snapshot feature provides an additional layer of fault protection if you maintain the same version of Junos OS in both partitions of resilient dual-root partitions. When **auto-snapshot** is enabled, repair happens automatically. Therefore, the switch does not issue an alarm to indicate that the system has rebooted from the alternate partition. However, it does log the event. You cannot execute a manual snapshot when an automatic snapshot procedure is in process. The login banner indicates that an automatic snapshot operation is in progress and that banner is removed only after the snapshot operation is complete. The next reboot happens from the primary partition.



NOTE: EX Series switches that ship with Junos OS Release 10.4R3 or later are configured with identical Junos OS images in the primary root partition (Slice 1) and the alternate root partition (Slice 2).

However, if you do *not* maintain the same version of Junos OS in both partitions, you might want to disable the automatic snapshot feature. If you have an earlier version of Junos OS in the alternate partition and the system reboots from the alternate root partition, the automatic snapshot feature causes the later Junos OS version to be replaced with the earlier version.

When automatic snapshot is disabled and the system reboots from the alternate root partition, it triggers an alarm indicating that the system has rebooted from its alternate partition.

Earlier Partition Scheme (Junos OS Release 10.4R2 and Earlier)

The partition scheme used in Junos OS 10.4R2 and earlier is shown in [Table 16 on page 114](#).

Table 16: Earlier Partition Scheme

Slice 1		Slice 2		Slice 3	
s1a	s1f	s2a	s2f	s3d	s3e
/ (root Junos OS)		/var (empty until initial software upgrade)		/var/tmp	/config

This is the partitioning scheme for a switch shipped with Release 10.4R2 or earlier (or after you reformat the disk during a downgrade from Release 10.4R3 or later to Release 10.4R2 or earlier). In this partitioning scheme, the switch comes from the factory with only one Junos OS image installed in the root Junos OS partition of Slice 1. The first time that you perform a software upgrade, the new Junos OS image is installed in Slice 2. If the switch fails to boot, you must manually trigger it to boot from the alternate partition (rebooting from the alternate partition does not occur automatically).

Understanding Upgrading or Downgrading Between Resilient Dual-Root Partition Releases and Earlier Releases

Upgrading from Release 10.4R2 or earlier to Release 10.4R3 or later differs from other upgrades in two important ways:

- You must install a new loader software package in addition to installing the new Junos OS image.
- Rebooting after the upgrade reformats the disk from three partitions to four partitions. See [Table 15 on page 112](#).

You can perform all operations for this special software upgrade from the CLI.



CAUTION: Back up any important log files because the `/var/log` files are not saved or restored during an upgrade from Release 10.4R2 or earlier to a release that supports resilient dual-root partitions (Release 10.4R3 or later).

We recommend that you also save your `/config` files and any important log files to an external medium because if there is a power interruption during the upgrade process, they might be lost.

Related Documentation

- [auto-snapshot on page 344](#)

Dual-Root Partitioning Scheme on SRX Series Devices

Junos OS Release 10.0 and later support dual-root partitioning on SRX Series devices. Dual-root partitioning allows the SRX Series device to remain functional even if there is file system corruption and to facilitate easy recovery of the file system.



NOTE: Starting in Junos OS Release 12.1X45, single root partitioning is not supported on SRX Series devices.

SRX Series devices running Junos OS Release 9.6 or earlier support a single-root partitioning scheme where there is only one root partition. Because both the primary and backup Junos OS images are located on the same root partition, the system fails to boot if there is corruption in the root file system. The dual-root partitioning scheme guards against this scenario by keeping the primary and backup Junos OS images in two independently bootable root partitions. If the primary root partition becomes corrupted, the system can still boot from the backup Junos OS image located in the other root partition and remain fully functional.

SRX Series devices that ship with Junos OS Release 10.0 or later are formatted with dual-root partitions from the factory. SRX Series devices that are running Junos OS Release 9.6 or earlier can be formatted with dual-root partitions when they are upgraded to Junos OS Release 10.0 or later.



NOTE: Although you can install Junos OS Release 10.0 or later on SRX Series devices with the single-root partitioning scheme, we strongly recommend the use of the dual-root partitioning scheme.

This section contains the following topics:

- [Boot Media and Boot Partition on SRX Series Devices on page 115](#)
- [Important Features of the Dual-Root Partitioning Scheme on page 116](#)
- [Understanding Automatic Recovery of the Primary Junos OS Image with Dual-Root Partitioning on page 117](#)
- [Understanding How the Primary Junos OS Image with Dual-Root Partitioning Recovers Devices on page 118](#)
- [Understanding How Junos OS Release 10.0 or Later Upgrades with Dual-Root Partitioning on page 120](#)

Boot Media and Boot Partition on SRX Series Devices

When the SRX Series device powers on, it tries to boot the Junos OS from the default storage media. If the device fails to boot from the default storage media, it tries to boot from the alternate storage media.

[Table 17 on page 115](#) provides information on the storage media available on SRX Series devices.

Table 17: Storage Media on SRX Series Devices

SRX Series Devices	Storage Media
SRX100, SRX210, and SRX240	<ul style="list-style-type: none"> • Internal NAND flash (default; always present) • USB storage device (alternate)

Table 17: Storage Media on SRX Series Devices (*continued*)

SRX Series Devices	Storage Media
SRX110, SRX220	<ul style="list-style-type: none"> • CompactFlash (default; always present) • USB storage device (alternate)
SRX300, SRX320, and SRX340, and SRX345	<ul style="list-style-type: none"> • eUSB disk (default; always present) • USB storage device (alternate)
SRX550	<ul style="list-style-type: none"> • Internal CF (default; always present) • USB storage device (alternate)
SRX550M	<ul style="list-style-type: none"> • Internal CF (default; always present) • USB storage device (alternate)
SRX650	<ul style="list-style-type: none"> • Internal CF (default; always present) • External flash card (alternate) • USB storage device (alternate)

With the dual-root partitioning scheme, the SRX Series device first tries to boot Junos OS from the primary root partition and then from the backup root partition on the default storage media. If both primary and backup root partitions of a media fail to boot, then the SRX Series device tries to boot from the next available type of storage media. The SRX Series device remains fully functional even if it boots Junos OS from the backup root partition of the storage media.

Important Features of the Dual-Root Partitioning Scheme

The dual-root partitioning scheme has the following important features:

- The primary and backup copies of Junos OS images reside in separate partitions. The partition containing the backup copy is mounted only when required. With the single-root partitioning scheme, there is one root partition that contains both the primary and the backup Junos OS images.
- The **request system software add** command for a Junos OS package erases the contents of the other root partition. The contents of the other root partition will not be valid unless software installation is completed successfully.
- Add-on packages, such as **jais** or **jfirmware**, can be reinstalled as required after a new Junos OS image is installed.
- The **request system software rollback** command does not delete the current Junos OS image. It is possible to switch back to the image by issuing the **rollback** command again.
- The **request system software delete-backup** and **request system software validate** commands do not take any action.

Understanding Automatic Recovery of the Primary Junos OS Image with Dual-Root Partitioning

The auto-snapshot feature repairs the corrupted primary root when the device reboots from the alternate root. This is accomplished by taking a snapshot of the alternate root onto the primary root automatically rather than manually from the CLI.

When this feature is enabled, and the device reboots from the alternate root (because of a corrupted primary root or power cycle during restart), the following actions take place:

1. A prominent message is displayed indicating a failure to boot from the primary root.

```
*****
**                                     **
** WARNING: THIS DEVICE HAS BOOTED FROM THE BACKUP JUNOS IMAGE **
**                                     **
** It is possible that the primary copy of JUNOS failed to boot up **
** properly, and so this device has booted from the backup copy. **
**                                     **
** Please re-install JUNOS to recover the primary copy in case **
** it has been corrupted and if auto-snapshot feature is not **
** enabled. **
**                                     **
*****
```

2. A system **boot from backup root** alarm is set. This is useful for devices that do not have console access.
3. A snapshot of the alternate root onto the primary root is made.
4. Once the snapshot is complete, the system **boot from backup root** alarm is cleared.

During the next reboot, the system determines the good image on the primary root and boots normally.



NOTE: We recommend performing the snapshot once all the processes start. This is done to avoid any increase in the reboot time.



NOTE:

- Auto-snapshot feature is supported on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
- By default the auto-snapshot feature is disabled.
- If you do not maintain the same version of Junos OS in both partitions, ensure that the automatic snapshot feature remains disabled. Otherwise, if you have an earlier version of Junos OS in the alternate partition and the system reboots from the alternate root partition, the automatic snapshot feature causes the later Junos OS version to be replaced with the earlier version.
- When automatic snapshot is disabled and the system reboots from the alternate root partition, it triggers an alarm indicating that the system has rebooted from its alternate partition.

Enable this feature with the **set system auto-snapshot** command. Once the primary root partition is recovered using this method, the device will successfully boot from the primary root partition on the next reboot.

Execute the **delete system auto-snapshot** command to delete all backed up data and disable auto-snapshot, if required.

Use the **show system auto-snapshot** command to check the auto-snapshot status.

When auto-snapshot is in progress, you cannot run a manual snapshot command concurrently and the following error message appears:

Snapshot already in progress. Please try after sometime.



NOTE: If you log into the device when the snapshot is in progress, the following banner appears: The device has booted from the alternate partition, auto-snapshot is in progress.

Understanding How the Primary Junos OS Image with Dual-Root Partitioning Recovers Devices

If the SRX Series Services Gateway is unable to boot from the primary Junos OS image, and boots up from the backup Junos OS image in the backup root partition, a message appears on the console at the time of login indicating that the device has booted from the backup Junos OS image.

login: user

Password:

```

**                                                                    **
**  WARNING: THIS DEVICE HAS BOOTED FROM THE BACKUP JUNOS IMAGE  **
**                                                                    **
**  It is possible that the active copy of JUNOS failed to boot up  **
**  properly, and so this device has booted from the backup copy.  **
**                                                                    **
**  Please re-install JUNOS to recover the active copy in case    **
**  it has been corrupted.                                         **
**                                                                    **
*****

```

Because the system is left with only one functional root partition, you must immediately restore the primary Junos OS image using one of the following methods:

- Install a new image using the CLI or J-Web user interface. The newly installed image will become the primary image, and the device will boot from it on the next reboot.
- Use a snapshot of the backup root partition by entering the **request system snapshot slice alternate** command. Once the primary root partition is recovered using this method, the device will successfully boot from the primary root partition on the next reboot. After the procedure, the primary root partition will contain the same version of Junos OS as the backup root partition.



NOTE: You can use the CLI command **request system snapshot slice alternate** to back up the currently running root file system (primary or secondary) to the other root partition on the system along with following:

- Save an image of the primary root partition in the backup root partition when system boots from the primary root partition.
- Save an image of the backup root partition in the primary root partition when system boots from the backup root partition.



WARNING: The process of restoring the alternate root by using the CLI command **request system snapshot slice alternate** takes several minutes to complete. If you terminate the operation before completion, the alternate root might not have all required contents to function properly.

Understanding How Junos OS Release 10.0 or Later Upgrades with Dual-Root Partitioning



NOTE: If you are upgrading to Junos OS Release 10.0 without transitioning to dual-root partitioning, use the conventional CLI and J-Web user interface installation methods.

To format the media with dual-root partitioning while upgrading to Junos OS Release 10.0 or later, use one of the following installation methods:

- Installation from the boot loader using a TFTP server. We recommend this if console access to the system is available and a TFTP server is available in the network. See [“Installing Junos OS on SRX Series Devices from the Boot Loader Using a TFTP Server” on page 79](#)
- Installation from the boot loader using a USB storage device. We recommend this method if console access to the system is available and the system can be physically accessed to plug in a USB storage device. See [“Installing Junos OS on SRX Series Devices from the Boot Loader Using a USB Storage Device” on page 82](#)
- Installation from the CLI using the **partition** option. We recommend this method only if console access is not available. This installation can be performed remotely.



NOTE: After upgrading to Junos OS Release 10.0 or later, the U-boot and boot loader must be upgraded for the dual-root partitioning scheme to work properly.

Release History Table

Release	Description
12.1X45-D10	Starting in Junos OS Release 12.1X45, single root partitioning is not supported on SRX Series devices.

Related Documentation

- [Understanding Junos OS Upgrades for SRX Series Devices on page 160](#)
- [Example: Installing Junos OS on SRX Series Devices Using the Partition Option on page 120](#)

Example: Installing Junos OS on SRX Series Devices Using the Partition Option

This example shows how to install Junos OS Release 10.0 or later with the **partition** option.

- [Requirements on page 121](#)
- [Overview on page 121](#)
- [Configuration on page 122](#)
- [Verification on page 123](#)

Requirements

Before you begin, back up any important data.

Overview

This example formats the internal media and installs the new Junos OS image on the media with dual-root partitioning. Reinstall the Release 10.0 or later image from the CLI using the **request system software add** command with the **partition** option. This copies the image to the device, and then reboots the device for installation. The device boots up with the Release 10.0 or later image installed with the dual-root partitioning scheme. When the **partition** option is used, the format and install process is scheduled to run on the next reboot. Therefore, we recommend that this option be used together with the **reboot** option.



NOTE: The process might take 15 to 20 minutes. The system is not accessible over the network during this time.



WARNING: Using the **partition** option with the **request system software add** command erases the existing contents of the media. Only the current configuration is preserved. You must back up any important data before starting the process.



NOTE: Partition install is supported on the default media on SRX300, SRX320, 340, and SRX345 devices (internal NAND flash) and *not* supported on the alternate media (USB storage key).



NOTE: Partition install is supported on the default media on SRX100, SRX210, and SRX240 devices (internal NAND flash) and on SRX650 devices (internal CF card). Partition install is not supported on the alternate media on SRX100, SRX210, and SRX240 devices (USB storage key) or on SRX650 devices (external CF card or USB storage key).

In this example, add the software package `junos-srxsme-10.0R2-domestic.tgz` with the following options:

- **no-copy** option to install the software package but do not save the copies of package files. You must include this option if you do not have enough space on the internal media to perform an upgrade that keeps a copy of the package on the device.
- **no-validate** option to bypass the compatibility check with the current configuration before installation starts.

- **partition** option to format and re-partition the media before installation.
- **reboot** option to reboots the device after installation is completed.

Configuration

CLI Quick Configuration

To install Junos OS Release 10.0 or later with the **partition** option, enter the following command from operational mode:

```
user@host>request system software add junos-srxsme-10.0R2-domestic.tgz no-copy  
no-validate partition reboot
```

GUI Step-by-Step Procedure

To install Junos OS Release 10.0 or later with the **partition** option:

1. In the J-Web user interface, select **Maintain>Software>Install Package**.
2. On the Install Package page, specify the FTP or HTTP server, file path, and software package name. Type the full address of the software package location on the FTP or HTTP. Example: **ftp://hostname/pathname/junos-srxsme-xx.0R2-domestic.tgz** or **http://hostname/pathname/junos-srxsme-xx.0R2-domestic.tgz**.



NOTE: Specify the username and password, if the server requires one.

3. Select the **Reboot If Required** check box to set the device to reboot automatically when the upgrade is complete.
4. Select the **Do not save backup** check box to bypass saving the backup copy of the current Junos OS package.
5. Select the **Format and re-partition the media before installation** check box to format the internal media with dual-root partitioning.
6. Click **Fetch and Install Package**. The software is activated after the device reboots.
This formats the internal media and installs the new Junos OS image on the media with dual-root partitioning.

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To install Junos OS Release 10.0 or later with the **partition** option:

1. Upgrade the device to Junos OS Release 10.0 or later using the CLI.

2. After the device reboots, upgrade the boot loader to the latest version. See [“Preparing the USB Flash Drive to Upgrade Junos OS on SRX Series Devices” on page 75](#).

3. Reinstall the Release 10.0 or later image.

```
user@host>request system software add junos-srxsme-10.0R2-domestic.tgz no-copy
no-validate partition reboot
Copying package junos-srxsme-10.0R2-domestic.tgz to var/tmp/install
Rebooting ...
```

Results From configuration mode, confirm your configuration by entering the **show system storage partitions** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Sample output on a system with single root partitioning:

```
user@host> show system storage partitions

Boot Media: internal (da0)

Partitions Information:
  Partition  Size  Mountpoint
    s1a      898M  /
    s1e      24M   /config
    s1f       61M   /var
```

Sample output on a system with dual-root partitioning:

```
user@host> show system storage partitions

Boot Media: internal (da0)
Active Partition: da0s2a
Backup Partition: da0s1a
Currently booted from: active (da0s2a)

Partitions Information:
  Partition  Size  Mountpoint
    s1a      293M  altroot
    s2a      293M  /
    s3e      24M   /config
    s3f      342M  /var
    s4a       30M  recovery
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- [Verifying the Partitioning Scheme Details on page 124](#)

Verifying the Partitioning Scheme Details

Purpose Verify that the partitioning scheme details on the SRX Series device were configured.

Action From operational mode, enter the **show system storage partitions** command.

Related Documentation

- [Dual-Root Partitioning Scheme on SRX Series Devices on page 114](#)
- [Reinstalling the Single-Root Partition on SRX Series Devices on page 124](#)

Reinstalling the Single-Root Partition on SRX Series Devices

Junos OS Release 9.6 and earlier is not compatible with the dual-root partitioning scheme. These releases can only be installed if the media is reformatted with single-root partitioning. Any attempt to install Junos OS Release 9.6 or earlier on a device with dual-root partitioning without reformatting the media will fail with an error. You must install the Junos OS Release 9.6 or earlier image from the boot loader using a TFTP server or USB storage device.



NOTE: Junos OS Release 12.1X45 and later do not support single root partitioning.



NOTE: You do not need to reinstall the earlier version of the boot loader if you are installing Junos OS Release 9.6.

You cannot install a Junos OS Release 9.6 or earlier package on a system with dual-root partitioning using the Junos OS CLI or J-Web. If this is attempted, an error will be returned.

You can install the Junos OS Release 9.6 (9.6R3 and 9.6R4 [only]) on a system with dual-root partitioning using **request system software add** command with **partition** option.

To reinstall the single-root partition:

1. Enter the **request system software add partition** command to install the previous Junos OS version (9.6R3 and 9.6R4):

```
user@host>request system software add partition
```

2. Reboot the device

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

The previous software version gets installed after rebooting the device.



NOTE: Using the `request system software add` CLI command with the `partition` option to install Junos OS Release 9.6 (9.6R3 and 9.6R4) reformats the media with single-root partitioning. This process erases the dual-root partitioning scheme from the system, so the benefits of dual-root partitioning will no longer be available.

**Related
Documentation**

- [Dual-Root Partitioning Scheme on SRX Series Devices on page 114](#)
- [Example: Installing Junos OS on SRX Series Devices Using the Partition Option on page 120](#)

CHAPTER 8

Upgrading Software

- [Upgrading Software Packages on page 128](#)
- [Overview of Upgrading to 64-bit Junos OS on page 131](#)
- [Upgrading Routers Using Unified ISSU on page 136](#)
- [Understanding Nonstop Software Upgrade on EX Series Switches on page 136](#)
- [Upgrading Software by Using Automatic Software Download on page 144](#)
- [Verifying That Automatic Software Download Is Working Correctly on page 145](#)
- [Verifying Junos OS and Boot Loader Software Versions on an EX Series Switch on page 146](#)
- [Upgrading the Loader Software on the Line Cards in a Standalone EX8200 Switch or an EX8200 Virtual Chassis on page 149](#)
- [Upgrading Junos OS with Upgraded FreeBSD on page 153](#)
- [Understanding Junos OS Upgrades for SRX Series Devices on page 160](#)
- [Preparing Your SRX Series Device for Junos OS Upgrades on page 162](#)
- [Understanding Junos OS with Upgraded FreeBSD for SRX5400, SRX5600, and SRX5800 Devices on page 165](#)
- [Downloading Software Packages from Juniper Networks on page 177](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
- [Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server on page 181](#)
- [Understanding BIOS Upgrades on SRX Series Devices on page 182](#)
- [Disabling Auto BIOS Upgrade on SRX Series Devices on page 184](#)
- [Reverting the Junos OS Software Image Back to the Previous Version on page 184](#)
- [Overview of CoS Upgrade Requirements \(Junos OS Release 11.1 or 11.2 to a Later Release\) on page 187](#)
- [Before You Begin Installing or Upgrading the Firmware on page 189](#)
- [Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\) on page 190](#)
- [Upgrading Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\) on page 191](#)

- [Installing Firmware on the 100-Gigabit DWDM OTN MIC \(MIC3-100G-DWDM\) on page 193](#)
- [Upgrading Firmware on the 100-Gigabit DWDM OTN MIC \(MIC3-100G-DWDM\) on page 194](#)

Upgrading Software Packages



NOTE: When you install individual software packages, the following notes apply:

- When upgrading from Junos OS Release 8.2 or earlier to Junos OS Release 8.5, use the `system software add <image> no-validate` command option.
- Only use the `jinstall` Junos OS image when upgrading or downgrading to or from Junos OS Release 8.5. Do not use the `jbundle` image.
- Before upgrading to Junos OS Release 8.5, ensure that the routing platform's CompactFlash card is 256 MB or larger to avoid disk size restrictions. (M7i routers without a CompactFlash card are excluded.)



NOTE: If you are upgrading a Routing Engine on a PTX Series router to run Junos OS Release 13.2R2 and later, and then make that Routing Engine the master Routing Engine, then the master Routing Engine reports a major alarm `CB 0/1 ESW PFE Port Fail` even though the Control Board's Ethernet switch links are up and running on both the master and the backup Routing Engines. This is because the backup Routing Engine is still on Junos OS Release 13.2R1 or earlier. The alarm is cleared after you have completed the upgrade of Junos OS on the backup Routing Engine.

```
User@router# show chassis alarms
2 alarms currently active
Alarm time Class Description
2014-10-15 00:44:31 BST Major CB 0 ESW PFE Port Fail
2014-10-15 00:42:42 BST Minor Backup RE Active
```

To upgrade an individual Junos OS package:

1. Download the software packages you need from the Juniper Networks Support Web site at <http://www.juniper.net/support/>. For information about downloading software packages, see “[Downloading Software](#)” on page 48.



NOTE: We recommend that you upgrade all individual software packages using an out-of-band connection from the console or management Ethernet interface, because in-band connections can be lost during the upgrade process.

2. Back up the currently running and active file system so that you can recover to a known, stable environment in case something goes wrong with the upgrade:

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

The root file system is backed up to `/altroot`, and `/config` is backed up to `/altconfig`. The root and `/config` file systems are on the router's CompactFlash card, and the `/altroot` and `/altconfig` file systems are on the router's hard disk or solid-state drive (SSD).



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



NOTE: SRX5000-line devices, the root file system is backed up to `/altroot`, and `/config` is backed up to `/altconfig`. The root and `/config` file systems are on the router's CompactFlash card, and the `/altroot` and `/altconfig` file systems are on the router's hard disk or solid-state drive (SSD).



NOTE: This step is optional for SRX300, SRX320, SRX340, SRX345, and SRX550M; for these devices, ensure that a USB flash drive is plugged into the USB port of the device.

3. If you are copying multiple software packages to the router, copy them to the `/var/tmp` directory on the hard disk or solid-state drive (SSD):

```
user@host> file copy ftp://username :prompt@ftp.hostname
.net/filename/var/tmp/filename
```

4. Add the new software package:

- To add an individual software package:

```
user@host> request system software add/var/tmp/ installation-package validate
```

installation-package is the full URL to the file.



NOTE: For SRX5800, SRX5600, and SRX5400 devices, do not include the `re0 | re1` option when you install a package using the `request system software add` command, if the Routing Engine on which the package is located and the Routing Engine on which you want to install the package is the same. In such cases, the package gets deleted after a successful upgrade.

If you are upgrading more than one package at the same time, add **jbase** first. If you are using this procedure to upgrade all packages at once, add them in the following order:

```
user@host> request system software add /var/tmp/jbase-release-signed.tgz
user@host> request system software add /var/tmp/jkernel-release-signed.tgz
user@host> request system software add /var/tmp/jpfe-release-signed.tgz
user@host> request system software add /var/tmp/jdocs-release-signed.tgz
user@host> request system software add /var/tmp/jweb-release-signed.tgz
user@host> request system software add /var/tmp/jroute-release-signed.tgz
user@host> request system software add /var/tmp/jcrypto-release-signed.tgz
```

- For M Series, MX Series, and T Series routers running Junos OS Release 12.2 and later, you can add more than one software package at the same time. To add multiple software packages:

```
user@host> request system software add set /var/tmp/
installation-package/var/tmp/ installation-package validate
```

installation-package can be any of the following:

- A list of installation packages, each separated by a blank space. For example:

```
user@host> request system software add set /var/tmp/
install-10.2R1.8-domestic-signed.tgz /var/tmp/jtools*.tgz validate
```
- The full URL to the directory or tar file containing the list of installation packages.

Use the **request system software add set** command to retain any SDK configuration by installing the SDK add-on packages along with the core Junos OS installation package.



WARNING: Do not include the `re0 | re1` option when you install a package using the `request system software add` command, if the Routing Engine on which the package is located and the Routing Engine on which you want to install the package are the same. In such cases, the package gets deleted after a successful upgrade.

The system might display the following message:

```
pkg_delete: couldn't entirely delete package
```

This message indicates that someone manually deleted or changed an item that was in a package. You do not need to take any action; the package is still properly deleted.

For more information about the **request system software add** command, see the [CLI Explorer](#).

5. Reboot the router to start the new software:

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

6. After you have upgraded or downgraded the software and are satisfied that the new software is successfully running, issue the **request system snapshot** command to back up the new software:

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



NOTE: On an ACX router, you must issue the **request system snapshot slice alternate** command.

The root file system is backed up to **/altroot**, and **/config** is backed up to **/altconfig**. The root and **/config** file systems are on the router's CompactFlash card, and the **/altroot** and **/altconfig** file systems are on the router's hard disk or solid-state drive (SSD).



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



NOTE: To install the Junos OS software package and host software package on routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines, see [“VM Host Installation” on page 268](#)

Related Documentation

- [VM Host Installation on page 268](#)

Overview of Upgrading to 64-bit Junos OS

Just like any other operating system, the 64-bit version of Junos OS can address more memory than the 32-bit version of Junos OS. In order to support larger Routing Engine memory sizes, an upgrade from the 32-bit to the 64-bit Junos OS running on the Routing Engine hardware is necessary.

The in-service software upgrade (ISSU) procedure is not supported while upgrading from the 32-bit version of Junos OS to the 64-bit version of Junos OS. The upgrade process involves some downtime, so traffic will be affected.

If you are starting with 32-bit Junos OS running on Routing Engines that are not 64-bit capable, there are two parts of the upgrade: upgrading the hardware and upgrading the software. This topic provides an overview of the upgrade tasks and the order in which they must be performed. For more detailed information about replacing the Routing Engines, see the hardware guide for your router.

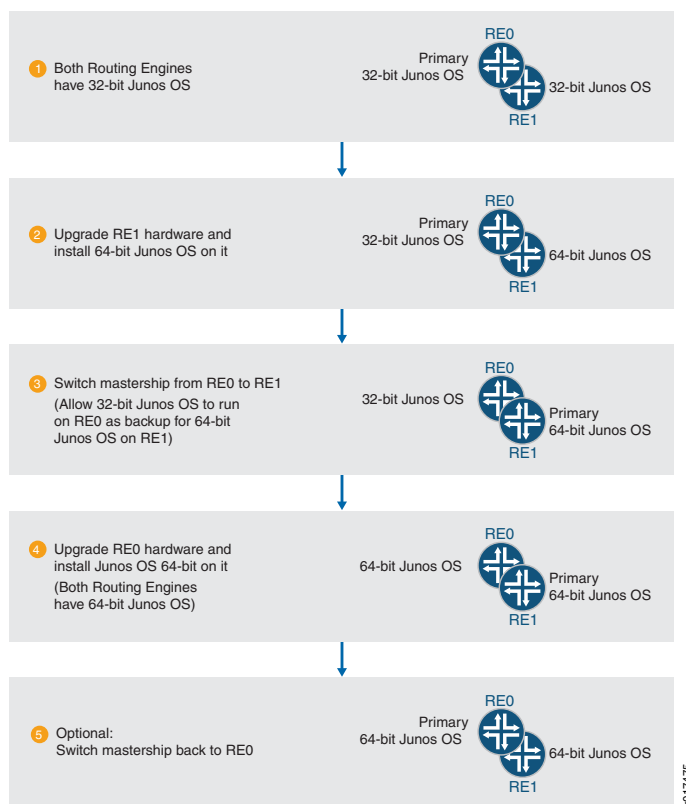
The following upgrade scenarios are covered in this overview:

- [Upgrading Redundant Routing Engines from 32-bit to 64-bit Junos OS on page 132](#)
- [Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using One Slot on page 134](#)
- [Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using Two Slots on page 134](#)

Upgrading Redundant Routing Engines from 32-bit to 64-bit Junos OS

For a diagram of this procedure, see [Figure 10 on page 132](#). For the purposes of this procedure, slot 0 has the primary Routing Engine initially.

Figure 10: Upgrading to 64-bit Junos OS with Redundant Routing Engines



To upgrade redundant Routing Engines from 32-bit Junos OS to 64-bit Junos OS:

1. If the backup Routing Engine in slot 1 is not 64-bit capable, replace it with a 64-bit capable Routing Engine.



NOTE: The 64-bit version of Junos OS is not supported on every Routing Engine. To determine whether your router and Routing Engine support a 64-bit version of Junos OS, see *Supported Routing Engines by Router*.

For instructions on replacing a Routing Engine, see the hardware guide for your router.

2. Log in to the primary Routing Engine in slot 0, and prepare the router for software package upgrade.

See [Preparing the Router for the Installation](#).

3. Install 64-bit Junos OS on the backup Routing Engine in slot 1.



CAUTION: Mixing 32-bit Junos OS and 64-bit Junos OS can only be done temporarily. It is not supported for normal operations.

See [Installing Software on the Backup Routing Engine](#).

4. Switch mastership from slot 0 to slot 1.

```
user@host> request chassis routing-engine master switch
```

Now the Routing Engine in slot 1 is the primary Routing Engine.

5. If the Routing Engine in slot 0 is not 64-bit capable, replace it with a 64-bit capable Routing Engine.



NOTE: The 64-bit version of Junos OS is not supported on every Routing Engine. To determine whether your router and Routing Engine support a 64-bit version of Junos OS, see *Supported Routing Engines by Router*.

For instructions on replacing a Routing Engine, see the hardware guide for your router.

6. Install 64-bit Junos OS on the Routing Engine in slot 0.

See [Installing Software on the Remaining Routing Engine](#).

7. (Optional) Switch mastership from slot 1 to slot 0.

```
user@host> request chassis routing-engine master switch
```

8. Finalize the installation.

See [Finalizing the Installation](#). This includes synchronization of the configuration on the Routing Engines.

Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using One Slot

To upgrade a single Routing Engine, from 32-bit to 64-bit Junos using one slot:

1. If the Routing Engine is not 64-bit capable, replace it with a 64-bit capable Routing Engine.



NOTE: The 64-bit version of Junos OS is not supported on every Routing Engine. To determine whether your router and Routing Engine support a 64-bit version of Junos OS, see *Supported Routing Engines by Router*.

For instructions on replacing a Routing Engine, see the hardware guide for your router.

2. Install 64-bit Junos OS on the Routing Engine using the **no-validate** option.

```
user@host> request system software add /var/tmp/software-package no-validate
```

For more details on installing software on a single router, see [Installing the Software Package on a Router with a Single Routing Engine](#).

3. Reboot.

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

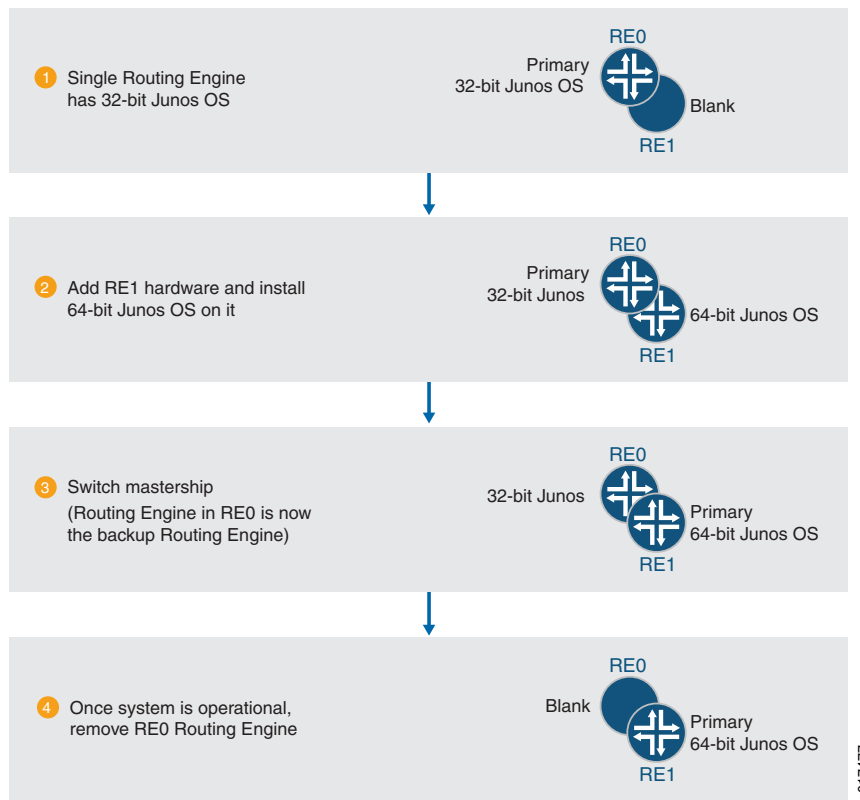
Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using Two Slots

This procedure provides a way to upgrade to a 64-bit Junos OS using two Routing Engine slots. Using two slots reduces the amount of network downtime.

If you have only one slot, use procedure “[Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using One Slot](#)” on page 134.

For a diagram of this procedure, see [Figure 11 on page 135](#). In this procedure, slot 1 is initially empty.

Figure 11: Upgrading a Single Routing Engine to 64-bit Junos OS Using Two Slots



To upgrade a single Routing Engine to 64-bit Junos OS using two Routing Engine slots:

1. Install a 64-bit capable Routing Engine in slot 1.



NOTE: The 64-bit version of Junos OS is not supported on every Routing Engine. To determine whether your router and Routing Engine support a 64-bit version of Junos OS, see *Supported Routing Engines by Router*.

For instructions on installing a Routing Engine, see the hardware guide for your router.

2. Install 64-bit Junos OS on the now backup Routing Engine in slot 1.

See [Installing Software on the Backup Routing Engine](#).



CAUTION: Mixing 32-bit Junos OS and 64-bit Junos OS is not supported for normal operations.

You need to remove the Routing Engine from slot 0 to avoid mixing 32-bit Junos OS and 64-bit Junos OS.

3. Switch the primary Routing Engine from slot 0 to slot 1.

```
user@host> request chassis routing-engine master switch
```

4. When the 64-bit Junos OS is configured properly, remove the Routing Engine from slot 0 .

For instructions on removing a Routing Engine, see the hardware guide for your router.

**Related
Documentation**

- [Installing the Software Package on a Router with Redundant Routing Engines on page 60](#)
- [Installing the Software Package on a Router with a Single Routing Engine on page 59](#)

Upgrading Routers Using Unified ISSU

Unified in-service software upgrade (ISSU) enables you to upgrade between two different Junos OS releases with no disruption on the control plane and with minimal disruption of traffic. Unified ISSU is only supported by dual Routing Engine platforms. In addition, graceful Routing Engine switchover (GRES) and nonstop active routing (NSR) must be enabled.

For additional information about using unified ISSU, see the *Junos OS High Availability Library for Routing Devices*.

**Related
Documentation**

- [Upgrading Software Packages on page 128](#)

Understanding Nonstop Software Upgrade on EX Series Switches

Nonstop software upgrade (NSSU) enables you to upgrade the software running on Juniper Networks EX Series Ethernet Switches with redundant Routing Engines and all member switches in EX Series Virtual Chassis by using a single command. During the upgrade there might be minimal network traffic disruption during mastership switchover, and the extent of disruption could be dependent on the network topology, configuration, network traffic, and other environment factors .



NOTE: When an EX Series switch in a mixed Virtual Chassis is upgraded to Junos OS Release 15.1 or later from a release earlier than Release 15.1, there might be a drop in traffic for up to 60 seconds.

NSSU is supported on the following EX Series platforms:

- EX3300 Virtual Chassis
- EX3400 Virtual Chassis
- EX4200 Virtual Chassis

- EX4300 Virtual Chassis
- EX4500 Virtual Chassis
- EX4550 Virtual Chassis
- All mixed Virtual Chassis composed of EX4200, EX4500, and EX4550 switches
- EX6200 switches
- EX8200 switches
- EX8200 Virtual Chassis

Performing an NSSU provides these benefits:

- No disruption to the control plane—An NSSU takes advantage of graceful Routing Engine switchover (GRES) and nonstop active routing (NSR) to ensure no disruption to the control plane. During the upgrade process, interface, kernel, and routing protocol information is preserved.
- Minimal disruption to network traffic—An NSSU minimizes network traffic disruption by:
 - Upgrading line cards one at a time in an EX6200 switch, EX8200 switch, or EX8200 Virtual Chassis, permitting traffic to continue to flow through the line cards that are not being upgraded.
 - Upgrading member switches one at a time in an EX3300, EX3400, EX4200, EX4300, EX4500, or mixed Virtual Chassis, permitting traffic to continue to flow through the members that are not being upgraded.

To achieve minimal disruption to traffic, you must configure link aggregation groups (LAGs) such that the member links of each LAG reside on different line cards or Virtual Chassis members. When one member link of a LAG is down, the remaining links are up, and traffic continues to flow through the LAG.



NOTE: Because NSSU upgrades the software on each line card or on each Virtual Chassis member one at a time, an upgrade using NSSU can take longer than an upgrade using the `request system software add` command.

In releases prior to Junos OS Release 16.1, for EX6200 switches, EX8200 switches, and EX8200 Virtual Chassis, you can reduce the amount of time an upgrade takes by configuring line-card upgrade groups. The line cards in an upgrade group are upgraded simultaneously, reducing the amount of time it takes to complete an upgrade. See *Configuring Line-Card Upgrade Groups for Nonstop Software Upgrade (CLI Procedure)*.

This topic covers:

- [Requirements for Performing an NSSU on page 138](#)
- [How an NSSU Works on page 139](#)
- [NSSU Limitations on page 142](#)

- [NSSU and Junos OS Release Support on page 142](#)
- [Overview of NSSU Configuration and Operation on page 143](#)

Requirements for Performing an NSSU

The following requirements apply to all switches and Virtual Chassis:

- All Virtual Chassis members and all Routing Engines must be running the same Junos OS release.
- Graceful Routing Engine switchover (GRES) must be enabled.
- Nonstop active routing (NSR) must be enabled.



NOTE: Although nonstop bridging (NSB) does not have to be enabled to perform an NSSU, we recommend enabling NSB before performing an NSSU. Enabling NSB ensures that all NSB-supported Layer 2 protocols operate seamlessly during the Routing Engine switchover that is part of the NSSU. In releases prior to Junos OS Release 16.1, see *Configuring Nonstop Bridging on Switches (CLI Procedure)*.

- For minimal traffic disruption, you must define link aggregation groups (LAGs) such that the member links reside on different Virtual Chassis members or on different line cards.

The following are requirements for EX3300, EX3400, EX4200, EX4300, EX4500, and mixed Virtual Chassis:

- The Virtual Chassis members must be connected in a ring topology so that no member is isolated as a result of another member being rebooted. This topology prevents the Virtual Chassis from splitting during an NSSU.
- The Virtual Chassis master and backup must be adjacent to each other in the ring topology. Adjacency permits the master and backup to always be in sync, even when the switches in linecard roles are rebooting.
- The Virtual Chassis must be preprovisioned so that the linecard role has been explicitly assigned to member switches acting in a linecard role. During an NSSU, the Virtual Chassis members must maintain their roles—the master and backup must maintain their master and backup roles (although mastership will change), and the remaining switches must maintain their linecard roles.
- A two-member Virtual Chassis must have **no-split-detection** configured so that the Virtual Chassis does not split when an NSSU upgrades a member.



NOTE: For the EX4300 Virtual Chassis, you should enable the `vcp-no-hold-time` statement at the `[edit virtual-chassis]` hierarchy level before performing a software upgrade using NSSU. If you do not enable the `vcp-no-hold-time` statement, the Virtual Chassis might split during the upgrade. A split Virtual Chassis can cause disruptions to your network, and you might have to manually reconfigure your Virtual Chassis after the NSSU if the split and merge feature was disabled. For more information about a split Virtual Chassis, see *Understanding Split and Merge in a Virtual Chassis*.

How an NSSU Works

This section describes what happens when you request an NSSU on these switches and Virtual Chassis:

- [EX3300, EX3400, EX4200, EX4300, EX4500, and Mixed Virtual Chassis on page 139](#)
- [EX6200 and EX8200 Switches on page 140](#)
- [EX8200 Virtual Chassis on page 141](#)

EX3300, EX3400, EX4200, EX4300, EX4500, and Mixed Virtual Chassis

When you request an NSSU on an EX3300, EX3400, EX4200, EX4300, EX4500, or mixed Virtual Chassis:

1. The Virtual Chassis master verifies that:
 - The backup is online and running the same software version.
 - Graceful Routing Engine switchover (GRES) and nonstop active routing (NSR) are enabled.
 - The Virtual Chassis has a preprovisioned configuration.
2. The master installs the new software image on the backup and reboots it.
3. The master resynchronizes the backup.
4. The master installs the new software image on member switches that are in the linecard role and reboots them, one at a time. The master waits for each member to become online and active before starting the software upgrade on the next member.
5. When all members that are in the linecard role have been upgraded, the master performs a graceful Routing Engine switchover, and the upgraded backup becomes the master.
6. The software on the original master is upgraded and the original master is automatically rebooted. After the original master has rejoined the Virtual Chassis, you can optionally return control to it by requesting a graceful Routing Engine switchover.

EX6200 and EX8200 Switches

When you request an NSSU on a standalone switch with redundant Routing Engines:

1. The switch verifies that:
 - Both Routing Engines are online and running the same software version.
 - Both Routing Engines have sufficient storage space for the new software image.
 - Graceful Routing Engine switchover and nonstop active routing are enabled.
2. The switch installs the new software image on the backup Routing Engine and reboots it.
3. The switch resynchronizes the backup Routing Engine to the master Routing Engine.
4. The line cards in the first upgrade group (or the line card in slot 0, if no upgrade groups are defined) download the new image and then restart. Traffic continues to flow through the line cards in the other upgrade groups during this process.
5. When line cards restarted in Step 4 are online again, the line cards in the next upgrade group download the new image and restart. This process continues until all online line cards have restarted with the new software.



NOTE: If you have taken a line card offline with the CLI before you start the NSSU, the line card is not restarted and remains offline.

6. The switch performs a graceful Routing Engine switchover, so that the upgraded backup Routing Engine becomes the master.
7. The switch installs the new software on the original master Routing Engine.

To complete the upgrade process, the original master Routing Engine must be rebooted. You can do so manually or have the switch perform an automatic reboot by including the **reboot** option when you request the NSSU. After the original master has been rebooted, you can optionally return control to it by requesting a graceful Routing Engine switchover.
8. (EX6200 switch only) The original master Routing Engine reboots to complete the software upgrade.



NOTE: To complete the upgrade process on an EX8200 switch, you must intervene to reboot the original master Routing Engine. You can reboot the original master Routing Engine manually or have the switch perform an automatic reboot by including the `reboot` option when you request the NSSU.

9. (Optional) After the original master has been rebooted, you can return control to it by requesting a graceful Routing Engine switchover.

The switch can maintain normal operations with either Routing Engine acting as the master Routing Engine after the software upgrade, so you only have to perform this switchover if you want to return Routing Engine control to the original master Routing Engine.

EX8200 Virtual Chassis

When you request an NSSU on an EX8200 Virtual Chassis:

1. The master external Routing Engine verifies that:
 - It has a backup external Routing Engine that is online.
 - All Virtual Chassis members have redundant Routing Engines and the Routing Engines are online.
 - All Routing Engines are running the same software version.
 - All Routing Engines have sufficient storage space for the new software image.
 - Graceful Routing Engine switchover and nonstop active routing (NSR) are enabled.
2. The master external Routing Engine installs the new software image on the backup external Routing Engine and reboots it.
3. The backup external Routing Engine resynchronizes with the master external Routing Engine.
4. The master external Routing Engine installs the new software on the backup Routing Engines in the member switches and reboots the backup Routing Engines.
5. When the reboot of the backup Routing Engines complete, the line cards in the first upgrade group download the new image and then restart. (If no upgrade groups are defined, the line card in slot 0 of member 0 downloads the new image and restarts.) Traffic continues to flow through the line cards in the other upgrade groups during this process.
6. When line cards restarted in Step 5 are online again, the line cards in the next upgrade group (or the next sequential line card) download the new image and restart. This process continues until all online line cards have restarted with the new software.



NOTE: If you have taken a line card offline with the CLI before you start the NSSU, the line card is not restarted and remains offline.

7. The new software image is installed on the master Routing Engines, both external and internal.
8. The member switches perform a graceful Routing Engine switchover, so that the upgraded backup Routing Engines become masters.
9. The master external Routing Engine performs a graceful Routing Engine switchover so that the backup external Routing Engine is now the master.

To complete the upgrade process, the original master Routing Engines, both external and internal, must be rebooted. You can do so manually by establishing a console connection to each Routing Engine or have the reboot performed automatically by including the **reboot** option when you request the NSSU. After the original master external Routing Engine has been rebooted, you can optionally return control to it by requesting a graceful Routing Engine switchover.

NSSU Limitations

You cannot use an NSSU to downgrade the software—that is, to install an earlier version of the software than is currently running on the switch. To install an earlier software version, use the **request system software add** command.

You cannot roll back to the previous software version after you perform an upgrade using NSSU. If you need to roll back to the previous software version, you can do so by rebooting from the alternate root partition if you have not already copied the new software version into the alternate root partition.

NSSU and Junos OS Release Support

A Virtual Chassis must be running a Junos OS release that supports NSSU before you can perform an NSSU. If a Virtual Chassis is running a software version that does not support NSSU, use the **request system software add** command.

Table 18 on page 142 lists the EX Series switches and Virtual Chassis that support NSSU and the Junos OS release at which they began supporting it.

Table 18: Platform and Release Support for NSSU

Platform	Junos OS Release
EX3300 Virtual Chassis	12.2 or later
EX3400 Virtual Chassis	15.1X53-D55
EX4200 Virtual Chassis	12.1 or later

Table 18: Platform and Release Support for NSSU (*continued*)

Platform	Junos OS Release
EX4300 Virtual Chassis	13.2X51-D20 or later
EX4500 Virtual Chassis	12.1 or later
EX4550 Virtual Chassis	12.2 or later
Mixed EX4200 and EX4500 Virtual Chassis	12.1 or later
Mixed EX4200 and EX4550 Virtual Chassis	12.2 or later
Mixed EX4200, EX4500, and EX4550 Virtual Chassis	12.2 or later
Mixed EX4500 and EX4550 Virtual Chassis	12.2 or later
EX6200 switch	12.2 or later
EX8200 switch	10.4 or later
EX8200 Virtual Chassis	11.1 or later

Overview of NSSU Configuration and Operation

You must ensure that the configuration of the switch or Virtual Chassis meets the requirements described in [“Requirements for Performing an NSSU” on page 138](#). NSSU requires no additional configuration.

In releases prior to Junos OS Release 16.1, for EX6200 switches, EX8200 switches, and EX8200 Virtual Chassis, you can optionally configure line-card upgrade groups using the CLI. See *Example: Configuring Line-Card Upgrade Groups for Nonstop Software Upgrade on EX Series Switches*.

You perform an NSSU by executing the **request system software nonstop-upgrade** command. For detailed instructions on how to perform an NSSU, see the topics in Related Documentation.

Release History Table

Release	Description
16.1	In releases prior to Junos OS Release 16.1, for EX6200 switches, EX8200 switches, and EX8200 Virtual Chassis, you can reduce the amount of time an upgrade takes by configuring line-card upgrade groups.

Related Documentation

- *Upgrading Software Using Nonstop Software Upgrade on EX Series Virtual Chassis and Mixed Virtual Chassis (CLI Procedure)*
- *Upgrading Software on an EX6200 or EX8200 Standalone Switch Using Nonstop Software Upgrade (CLI Procedure)*

- *Upgrading Software on an EX8200 Virtual Chassis Using Nonstop Software Upgrade (CLI Procedure)*
- *Configuring Nonstop Active Routing on Switches*
- *Configuring Graceful Routing Engine Switchover in a Virtual Chassis (CLI Procedure)*
- *Example: Configuring Line-Card Upgrade Groups for Nonstop Software Upgrade on EX Series Switches*

Upgrading Software by Using Automatic Software Download

The automatic software download feature uses the Dynamic Host Configuration Protocol (DHCP) message exchange process to download and install software packages. You configure the automatic software download feature on switches that act as DHCP clients. You must enable automatic software download on a switch before the software upgrade can occur.

You configure a path to a software package file on the DHCP server. The server communicates the path to the software package file through DHCP server messages.

If you enable automatic software download, the DHCP client switch compares the software package name in the DHCP server message with the name of the software package that booted the switch. If the software packages are different, the DHCP client switch downloads and installs the software package specified in the DHCP server message.

Before you upgrade software by using automatic software download, ensure that you have configured DHCP services for the switch, including configuring a path to a boot server and a boot file.

To configure a path to a boot server and a boot file:

1. Configure the name of the boot server advertised to DHCP clients. The client uses a boot file located on the boot server to complete DHCP setup. This configuration is equivalent to DHCP option 66:

```
[edit system services dhcp]
user@switch# set boot-server (address | hostname)
```

2. Set the boot file advertised to DHCP clients. After the client receives an IP address and the boot file location from the DHCP server, the client uses the boot image stored in the boot file to complete the DHCP setup. This configuration is equivalent to DHCP option 67:

```
[edit system services dhcp]
user@switch# set boot-file filename
```

To enable automatic software download on a switch that acts as a DHCP client:

```
[edit chassis]
user@switch# set auto-image-upgrade
```

After automatic software download is enabled on your DHCP client switch and after DHCP services are enabled on your network, an automatic software download can occur at any time as part of the DHCP message exchange process.

If an automatic software download occurs, you see the following message on the switch:

```
Auto-image upgrade started
On successful installation system will reboot automatically
```

The switch reboots automatically to complete the upgrade.

- Related Documentation**
- [Verifying That Automatic Software Download Is Working Correctly on page 145](#)
 - [Understanding Software Installation on EX Series Switches on page 44](#)
 - [Configuring a DHCP Server on Switches \(CLI Procedure\)](#)
 - [Configuring DHCP Services \(J-Web Procedure\)](#)

Verifying That Automatic Software Download Is Working Correctly

Purpose Verify that the automatic software download feature is working correctly.

Action Use the `show system services dhcp client interface-name` command to verify that the automatic software download feature has been used to install a software package.

```
user@switch> show system services dhcp client ge-0/0/1.0
Logical Interface Name      ge-0/0/1.0
Hardware address           00:0a:12:00:12:12
Client Status              bound
Vendor Identifier          ether
Server Address             10.1.1.1
Address obtained           10.1.1.89
Lease Obtained at          2009-08-20 18:13:04 PST
Lease Expires at           2009-08-22 18:13:04 PST

DHCP Options :
  Name: name-server, Value: [ 10.209.194.131, 203.0.113.2, 203.0.113.3 ]

  Name: server-identifier, Value: 10.1.1.1
  Name: router, Value: [ 10.1.1.80 ]
  Name: boot-image,
  Value: jinstall-ex-4200-9.6R1.5-domestic-signed.tgz
  Name: boot-image-location,
  Value: 10.1.1.25:/bootfiles/
```

Meaning The output from this command shows the name and location of the software package under DHCP options when automatic software download was last used to install a software package. The sample output in DHCP options shows that the last DHCP server message to arrive on the DHCP client had a boot server address of 192.168.1.165 and a boot file named `jinstall-ex-4200-9.6R1.5-domestic-signed.tgz`. If automatic software

download was enabled on this client switch during the last DHCP message exchange, these values were used by the switch to upgrade the software.

- Related Documentation**
- [Upgrading Software by Using Automatic Software Download on page 144](#)

Verifying Junos OS and Boot Loader Software Versions on an EX Series Switch

Before or after upgrading or downgrading Junos OS, you might need to verify the Junos OS version. You might also need to verify the boot loader software version if you are upgrading to or downgrading from a release that supports resilient dual-root partitions (Junos OS Release 10.4R3 and later).

This topic includes:

- [Verifying the Number of Partitions and File System Mountings on page 146](#)
- [Verifying the Loader Software Version on page 147](#)
- [Verifying Which Root Partition Is Active on page 148](#)
- [Verifying the Junos OS Version in Each Root Partition on page 148](#)

Verifying the Number of Partitions and File System Mountings

Purpose Between Junos OS Release 10.4R2 and Release 10.4R3, upgrades were made to further increase resiliency of root partitions, which required reformatting the disk from three partitions to four partitions. If your switch is running Release 10.4R2 or earlier, it has three partitions, and if it is running Release 10.4R3 or later, it has four partitions.

Action Verify how many partitions the disk has, as well as where each file system is mounted, by using the following command:

```
user@switch> show system storage
fpc0:
```

```
-----
Filesystem Size Used Avail Capacity Mounted on
/dev/da0s1a 184M 124M 45M 73% /
devfs 1.0K 1.0K 0B 100% /dev
/dev/md0 37M 37M 0B 100% /packages/mnt/jbase
/dev/md1 18M 18M 0B 100%
/packages/mnt/jcrypto-ex-10.4I20110121_0509_hbRPSRLI15184421081
/dev/md2 6.1M 6.1M 0B 100%
/packages/mnt/jdocs-ex-10.4I20110121_0509_hbRPSRLI15184421081
/dev/md3 154M 154M 0B 100%
/packages/mnt/jkernel-ex-10.4I20110121_0509_hbRPSRLI15184421081
/dev/md4 23M 23M 0B 100%
/packages/mnt/jpfe-ex42x-10.4I20110121_0509_hbRPSRLI15184421081
/dev/md5 46M 46M 0B 100%
/packages/mnt/jroute-ex-10.4I20110121_0509_hbRPSRLI15184421081
/dev/md6 28M 28M 0B 100%
/packages/mnt/jswitch-ex-10.4I20110121_0509_hbRPSRLI15184421081
/dev/md7 22M 22M 0B 100%
/packages/mnt/jweb-ex-10.4I20110121_0509_hbRPSRLI15184421081
/dev/md8 126M 10.0K 116M 0% /tmp
```

```

/dev/da0s3e 123M 632K 112M 1% /var
/dev/da0s3d 369M 20K 339M 0% /var/tmp
/dev/da0s4d 62M 62K 57M 0% /config
/dev/md9 118M 12M 96M 11% /var/rundb
procfs 4.0K 4.0K 0B 100% /proc
/var/jail/etc 123M 632K 112M 1%
/packages/mnt/jweb-ex-10.4I20110121_0509_hbRPSRLI15184421081/jail/var/etc
/var/jail/run 123M 632K 112M 1%
/packages/mnt/jweb-ex-10.4I20110121_0509_hbRPSRLI15184421081/jail/var/run
/var/jail/tmp 123M 632K 112M 1%
/packages/mnt/jweb-ex-10.4I20110121_0509_hbRPSRLI15184421081/jail/var/tmp
/var/tmp 369M 20K 339M 0%
/packages/mnt/jweb-ex-10.4I20110121_0509_hbRPSRLI15184421081/jail/var/tmp/uploads
devfs 1.0K 1.0K 0B 100%
/packages/mnt/jweb-ex-10.4I20110121_0509_hbRPSRLI15184421081/jail/dev

```

Meaning The presence of the partition name containing **s4d** indicates that there is a fourth slice. If this were a three-slice partition scheme, in place of **s1a**, **s3e**, **s3d**, and **s4d**, you would see **s1a**, **s1f**, **s2a**, **s2f**, **s3d**, and **s3e**, and you would not see **s4d**.

Verifying the Loader Software Version

Purpose For the special case of upgrading from Junos OS Release 10.4R2 or earlier to Release 10.4R3 or later, you must upgrade the loader software.

Action For EX Series switches except EX8200 switches:

```

user@switch> show chassis firmware
Part      Type      Version
FPC 0     uboot     U-Boot 1.1.6 (Jan  3 2011 - 16:14:58) 1.0.0

              loader     FreeBSD/PowerPC U-Boot bootstrap loader 2.4

```

For EX8200 switches:

```

user@switch> show chassis firmware
Part      Type      Version
FPC 0     uboot     U-Boot 1.1.6 (Jan  3 2011 - 16:14:58) 3.5.0

              loader     FreeBSD/PowerPC U-Boot bootstrap loader 2.4

```

Meaning For EX Series switches other than EX8200 switches, with Junos OS Release 10.4R3 or later installed:

- If there is version information following the timestamp for **U-Boot** (**1.0.0** in the preceding example), then the loader software does not require upgrading.
- If there is no version number following the timestamp for **U-boot**, then the loader software requires upgrading.



NOTE: If the software version is Release 10.4R2 or earlier, no version number is displayed following the timestamp for U-boot, regardless of the loader software version installed. If you do not know whether you have installed the new loader software, we recommend that you upgrade the loader software when you upgrade the software version.

For EX8200 switches, if the version number following the timestamp for **U-Boot** is earlier than **3.5.0**, you must upgrade the loader software when you upgrade the software version.

Verifying Which Root Partition Is Active

Purpose Switches running Release 10.4R3 or later have resilient dual-root partition functionality, which includes the ability to boot transparently from the inactive partition if the system fails to boot from the primary root partition.

You can verify which root partition is active using the following command:

Action user@switch> **show system storage partitions**
fpc0:

```
-----  
Boot Media: internal (da0)  
Active Partition: da0s1a  
Backup Partition: da0s2a  
Currently booted from: active (da0s1a)  
Partitions information:  
  Partition  Size  Mountpoint  
  s1a        184M  /  
  s2a        184M  altroot  
  s3d        369M  /var/tmp  
  s3e        123M  /var  
  s4d         62M  /config  
  s4e                unused (backup config)
```

Meaning The **Currently booted from:** field shows which root partition is active.

Verifying the Junos OS Version in Each Root Partition

Purpose Each switch contains two root partitions. We recommend that you copy the same Junos OS version in each partition when you upgrade. In Junos OS Release 10.4R2 and earlier, you might choose to have different Junos OS release versions in each partition. You might have different versions during a software upgrade and before you have finished verifying the new software installation. To enable a smooth reboot if corruption is found in the primary root file system, ensure that the identical Junos OS images are in each root partition. For Release 10.4R2 and earlier, you must manually reboot the switch from the backup root partition. However, for Release 10.4R3 and later, the switch reboots

automatically from the backup root partition if it fails to reboot from the active root partition.

Action Verify whether both root partitions contain the same image by using the following command:

```
user@switch> show system snapshot media internal
Information for snapshot on      internal (/dev/da0s1a) (backup)
Creation date: Jan 11 03:02:59 2012
JUNOS version on snapshot:
  jbase      : ex-12.2I20120305_2240_user
  jcrypto-ex: 12.2I20120305_2240_user
  jdocs-ex   : 12.2I20120305_2240_user
  jroute-ex  : 12.2I20120305_2240_user
  jswitch-ex : 12.2I20120305_2240_user
  jweb-ex    : 12.2I20120305_2240_user
Information for snapshot on      internal (/dev/da0s2a) (primary)
Creation date: Mar 6 02:24:08 2012
JUNOS version on snapshot:
  jbase      : ex-12.2I20120305_2240_user
  jcrypto-ex: 12.2I20120305_2240_user
  jdocs-ex   : 12.2I20120305_2240_user
  jroute-ex  : 12.2I20120305_2240_user
  jswitch-ex : 12.2I20120305_2240_user
  jweb-ex    : 12.2I20120305_2240_user
```

Meaning The command shows which Junos OS version is installed on each media partition. Verify that the same version is installed on both partitions.

- Related Documentation**
- [Troubleshooting Software Installation on page 329](#)
 - [Troubleshooting a Switch That Has Booted from the Backup Junos OS Image on page 332](#)
 - [Understanding Resilient Dual-Root Partitions on Switches on page 111](#)

Upgrading the Loader Software on the Line Cards in a Standalone EX8200 Switch or an EX8200 Virtual Chassis

You are almost never required to upgrade the loader software on the line cards in an EX8200 switch.

Upgrading the loader software version for a line card is not a requirement to complete any software upgrade. In rare cases, a line card might go offline immediately after a software upgrade because the loader software version on the line card requires an upgrade to become compatible with the upgraded Junos OS. You can upgrade the loader software on the line cards as a best practice to avoid this problem and other less severe issues.

The loader software on any line card in an EX8200 switch is updated using the same loader software package that upgrades the EX8200 Routing Engine loader software. The line card software loader contains two banks, each with a single loader software

version. This procedure is used to upgrade the loader software for both banks of a line card in a standalone EX8200 switch or an EX8200 Virtual Chassis.

To upgrade the loader software on the line cards in a standalone EX8200 switch or an EX8200 Virtual Chassis:



NOTE: If you are upgrading Junos OS, the Routing Engine loader software, and the line card loader software, we recommend that you upgrade in this order: Junos OS, line card loader software, Routing Engine loader software.

1. Determine the version of the loader software for the line cards:

```
user@switch> show chassis firmware
```

Part	Type	Version
FPC 6	U-Boot loader	U-Boot 1.1.6 (Jan 13 2009 - 06:55:22) 2.3.0 FreeBSD/PowerPC U-Boot bootstrap loader 2.2
FPC 7	U-Boot loader	U-Boot 1.1.6 (Jan 13 2009 - 06:55:22) 2.3.0 FreeBSD/PowerPC U-Boot bootstrap loader 2.2
Routing Engine 0	U-Boot loader	U-Boot 1.1.6 (Mar 11 2011 - 04:29:01) 3.5.0 FreeBSD/PowerPC U-Boot bootstrap loader 2.4
Routing Engine 1	U-Boot loader	U-Boot 1.1.6 (Mar 11 2011 - 04:29:01) 2.3.0 FreeBSD/PowerPC U-Boot bootstrap loader 2.4



NOTE: On an EX8200 Virtual Chassis, you cannot execute the `show chassis firmware` command on the master external Routing Engine. You must execute this command on each member switch.

- a. From the master external Routing Engine, start a shell session on the member switch, for example:

```
user@external-routing-engine> request session member 0
```

- b. Enter the CLI and execute the `show chassis firmware` command.
- c. Repeat these steps for the other member switch.

The loader software version appears after the timestamp (see the **Version** column in the output) for each component. For example, In the example given in this step, look at the first FPC listed (FPC 6). Ignore the U-Boot version number (1.1.6) and find the loader software version number (2.3.0) after the timestamp (U-Boot 1.1.6 (Jan 13 2009 - 06:55:22)). The U-Boot version number has nothing to do with the loader software version that you need to determine.

If the loader software version is earlier than 3.5.0 for any FPC, you should consider upgrading the loader software for that line card.

2. Download the loader software package from the Juniper Networks website and place the software package on an internal software distribution site or in a local directory on the switch. We recommend using `/var/tmp` as the local directory on the switch.



NOTE: To obtain the loader software package, see the Download Software page at <http://www.juniper.net/support/downloads/junos.html>. Click on the version, then the Software tab, and then the name of the software install package. In the pop-up Alert box, click the link to the PSN document.

3. Disable graceful Routing Engine switchover (GRES) and nonstop active routing (NSR), if enabled. Commit the configuration:

```
user@switch# deactivate chassis redundancy graceful-switchover
user@switch# deactivate routing-options nonstop-routing
user@switch# commit synchronize
```

4. Install the loader package:

```
user@switch> request system software add package
```

Replace *package* with one of the following paths:

- For a software package in the `/var/tmp` directory on the switch or external Routing Engine—`/var/tmp/package.tgz`.
- For a software package on a remote server:
 - `ftp://hostname/pathname/package.tgz`
 - `http://hostname/pathname/package.tgz`

In the above options, *package.tgz* might be, for example, `jloader-ex-8200-11.3build-signed.tgz`.

5. Upgrade the loader software.

- To upgrade the loader software for a line card on a standalone EX8200 switch:

```
user@switch> request system firmware upgrade fpc slot slot-number
Firmware upgrade initiated....
Please wait for ~2mins for upgrade to complete....
```

- To upgrade the loader software for a line card on an EX8200 member switch in an EX8200 Virtual Chassis:

```
user@switch> request system firmware upgrade fpc slot slot-number member member-id
Firmware upgrade initiated....
Please wait for ~2mins for upgrade to complete....
```

6. Confirm the loader software upgrade:

```
user@switch> show system firmware
```

Part	Type	Tag	Current version	Available version	Status
FPC 6	U-Boot	0	2.3.0		UPGRADED SUCCESSFULLY
FPC 7	U-Boot	0	2.3.0		OK
Routing Engine 0	RE BIOS	0	3.1.1		OK
Routing Engine 1		0	3.1.1		OK

The status is **UPGRADED SUCCESSFULLY** if the boot loader version update process is complete.

The status is **PROGRAMMING** if the boot loader version update process is still in progress.

Do not proceed to the next step until the **show system firmware** output confirms that the loader software upgrade is complete.

7. Restart the line card.

- To restart a line card on a standalone EX8200 switch:

```
user@switch> request chassis fpc restart slot slot-number
```

- To restart a line card on an EX8200 member switch in an EX8200 Virtual Chassis:

```
user@switch> request chassis fpc restart slot slot-number member member-id
```



NOTE: You can monitor the status of the line card restart by using the **show chassis fpc** command.

8. After the line card restart has completed, confirm the loader software version update:

```
user@switch> show chassis firmware
```

Part	Type	Tag	Current version	Available version	Status
FPC 6	U-Boot	0	3.5.0		OK
FPC 7	U-Boot	0	2.3.0		OK
Routing Engine 0	RE BIOS	0	3.1.1		OK
Routing Engine 1		0	3.1.1		OK

The current version has updated to 3.5.0. You have upgraded the loader software for one bank of the line card.

9. Repeat Steps 4 through 7 to upgrade the loader software on the other bank of the line card.



NOTE: A bank switchover occurs automatically as part of the line card restart. Repeating Steps 3 through 6 updates the loader software on the other bank.

10. Repeat Steps 4 through 8 for all other line cards that require a line card loader version upgrade.

Related Documentation

- *Upgrading Software on an EX6200 or EX8200 Standalone Switch Using Nonstop Software Upgrade (CLI Procedure)*
- *Upgrading Software on an EX8200 Virtual Chassis Using Nonstop Software Upgrade (CLI Procedure)*
- *Troubleshooting an EX8200 Line Card's Failure to Power On*

Upgrading Junos OS with Upgraded FreeBSD



NOTE: If you are upgrading or installing Junos OS on a VM host, see “[VM Host Installation](#)” on page 268.

Starting in Junos OS Release 15.1, certain hardware platforms run an upgraded FreeBSD kernel instead of older versions of FreeBSD.

Direct validation of the running configuration does not work for upgrading to Junos OS with upgraded FreeBSD from Junos OS based on older versions of the FreeBSD kernel. Therefore, when upgrading or downgrading between Junos OS and Junos OS with upgraded FreeBSD, you might have to validate on a different host. It does not matter where that other host is, as long as you can reach it with NETCONF over SSH (see *Establishing an SSH Connection for a NETCONF Session*). The target system uses the network to contact the other host, run the validation and authentication, and return the result.

Upgrading to Junos OS with upgraded FreeBSD reformats the file system. Only specific files and directories are preserved unless precautions are taken. By default, the upgrade process only preserves the following directories:

- `/config`
- `/etc/localtime`
- `/var/db`
- `/var/etc/master.passwd`
- `/var/etc/inetd.conf`
- `/var/etc/pam.conf`

- `/var/etc/resolv.conf`
- `/var/etc/syslog.conf`
- `/var/etc/localtime`
- `/var/etc/exports`
- `/var/etc/extensions.allow`
- `/var/preserve`
- `/var/tmp/baseline-config.conf`
- `/var/tmp/preinstall_boot_loader.conf`



NOTE: On EX2300 and EX3400 switches, the following directories are not applicable:

- `/etc/localtime`
- `/var/etc/localtime`
- `/var/etc/exports`
- `/var/preserve`
- `/var/tmp/preinstall_boot_loader.conf`

For specific installation procedures, see the following sections:

- [Determining Which Package or Packages to Install on page 154](#)
- [Installing Junos OS with Upgraded FreeBSD Over Plain Junos OS on page 156](#)
- [Installing Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Different Release on page 158](#)
- [To Install Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Later Release on page 159](#)

Determining Which Package or Packages to Install

In Junos OS releases earlier than 15.1, the partition swap pages were counted as part of the memory file system partition. Using this method leaves 4 GB of memory as the maximum that is theoretically accessible when you are using a 32-bit image. However, when Junos OS with upgraded FreeBSD is run, the system only counts the actual partition size, which leaves around 3.4 GB of available physical address space, or only 3 GB of usable RAM. Therefore, we recommend you use a 64-bit image with Junos OS with upgraded FreeBSD.

Before you begin:

1. Verify that the upgrade applies to your platform, as listed in [“Understanding Junos OS with Upgraded FreeBSD” on page 17](#).
2. Download the Junos OS with upgraded FreeBSD package.

See “Understanding Junos OS with Upgraded FreeBSD Package Names” on page 21.

3. Determine the upgrade path to follow.

The Junos OS release currently running on the device determines the upgrade path you need to follow to upgrade to Junos OS with upgraded FreeBSD, as shown in [Table 19 on page 155](#). Other upgrade paths might work, but they are not supported.



NOTE: Upgrades cannot span more than three releases in one upgrade.

Table 19: Upgrade Path to Junos OS with the Upgraded FreeBSD

Current Router's Junos OS Release	Upgrade Path	Example
earlier than Release 12.3	Upgrade in multiple steps, skipping no more than two releases in one upgrade.	To upgrade from Release 12.1, upgrade first to Release 13.1, then to Release 14.1, then from there to either Release 15.1 or 16.1.
12.3 to 13.2	Upgrade in two steps.	<p>To upgrade from Release 12.3, first upgrade to Release 13.3, then upgrade to Release 15.1.</p> <p>To upgrade from Release 13.2, first upgrade to Release 14.2, then upgrade to Release 15.1, 16.1, or 17.1.</p>
13.3 to 14.2	Upgrade in a single step.	<p>To upgrade from Release 13.3, upgrade directly to Release 15.1.</p> <p>To upgrade from Release 14.2, upgrade directly to either Release 15.1, 16.1, or 17.1.</p>



NOTE: You can also downgrade from Junos OS Release 15.1 to an earlier release of Junos OS, as long as the path complies with the Junos OS policy of skipping at most two earlier releases.

Installing Junos OS with Upgraded FreeBSD Over Plain Junos OS



NOTE: If you have important files in other directories, copy them from the router or switch to a secure location before upgrading the router or switch.



NOTE: The following procedure refers to routers, but it also applies to switches.

To install Junos OS with upgraded FreeBSD over plain Junos OS:

1. Enter the **request system software add *package-name* no-validate** command from the operational mode in the CLI:



NOTE: The **no-copy** option is enabled by default.

Use the **no-validate** option with the **request system software add** command. If you leave out the **no-validate** option, the command uses the **validate** option by default, and direct validation of the running configuration does not work for upgrading to Junos OS with upgraded FreeBSD from Junos OS based on older versions of the FreeBSD kernel.



NOTE: You can also use the **reboot** option along with the **request system software add** command, but it is not recommended to do this in a single step while upgrading from a FreeBSD 6.1 based Junos OS to FreeBSD 10 based Junos OS.



NOTE: To validate the current configuration on an upgrade to Junos OS with upgraded FreeBSD from Junos OS, use the **request system software validate on (Junos OS with Upgraded FreeBSD)** command.

```
user@host>request system software add
/var/tmp/junos-install-mx-x86-32-15.1R1.9.tgz no-validate
Installing package '/var/tmp/junos-install-mx-x86-32-15.1R1.9.tgz' ...
Verified manifest signed by PackageProductionEc_2015
Verified manifest signed by PackageProductionRSA_2015
Verified contents.iso
Verified issu-indb.tgz
Verified junos-x86-32.tgz
Verified kernel
Verified metatags
Verified package.xml
Verified pkgtools.tgz
camcontrol: not found
camcontrol: not found
Verified manifest signed by PackageProductionEc_2015
```



```

Saving the config files ...
NOTICE: uncommitted changes have been saved in
/var/db/config/juniper.conf.pre-install
Saving package file in
/var/sw/pkg/junos-install-x86-32-domestic-20150618.043753_builder_junos_151_r1.tgz
...
Saving state for rollback ...

```

The new Junos OS image is installed on the router.

2. Reboot the device to start the new software using the **request system reboot** command:

```

user@host> request system reboot
Reboot the system? [yes, no] (no) yes

```



NOTE: You must reboot the device to load the newly installed version of Junos OS on the device.

To abort the installation, do not reboot the device. Instead, finish the installation and then issue the **request system software delete *package-name*** command where *package* is, for example, **junos-install-mx-x86-32-15.1R1.9.tgz**. This is your last chance to stop the installation (not applicable on EX2300 and EX3400 platforms).

The software is loaded when you reboot the system. Installation can take between 5 and 10 minutes. The device then reboots from the boot device on which the software was just installed. When the reboot is complete, the device displays the login prompt.

While the software is being upgraded, the Routing Engine on which you are performing the installation does not route traffic.

3. Log in and issue the **show version** command to verify the version of the software installed.

```

user@host> show version
Hostname: host
Model: mx240
Junos: 15.1R1.9
JUNOS OS Kernel 32-bit [20150617.306001_builder_stable_10]
JUNOS OS runtime [20150617.306001_builder_stable_10]
JUNOS OS time zone information [20150617.306001_builder_stable_10]
JUNOS py base [20150618.043753_builder_junos_151_r1]
JUNOS OS crypto [20150617.306001_builder_stable_10]
JUNOS network stack and utilities [20150618.043753_builder_junos_151_r1]
JUNOS libs [20150618.043753_builder_junos_151_r1]
JUNOS runtime [20150618.043753_builder_junos_151_r1]
JUNOS platform support [20150618.043753_builder_junos_151_r1]
JUNOS modules [20150618.043753_builder_junos_151_r1]
JUNOS daemons [20150618.043753_builder_junos_151_r1]
JUNOS Voice Services Container package [20150618.043753_builder_junos_151_r1]
JUNOS Services SSL [20150618.043753_builder_junos_151_r1]
JUNOS Services Stateful Firewall [20150618.043753_builder_junos_151_r1]
JUNOS Services RPM [20150618.043753_builder_junos_151_r1]
JUNOS Services PTSP Container package [20150618.043753_builder_junos_151_r1]
JUNOS Services NAT [20150618.043753_builder_junos_151_r1]
JUNOS Services Mobile Subscriber Service Container package

```

```
[20150618.043753_builder_junos_151_r1]
JUNOS Services MobileNext Software package
[20150618.043753_builder_junos_151_r1]
JUNOS Services LL-PDF Container package
[20150618.043753_builder_junos_151_r1]
JUNOS Services Jflow Container package [20150618.043753_builder_junos_151_r1]
JUNOS Services IPSec [20150618.043753_builder_junos_151_r1]
JUNOS IDP Services [20150618.043753_builder_junos_151_r1]
JUNOS Services HTTP Content Management package
[20150618.043753_builder_junos_151_r1]
JUNOS Services Crypto [20150618.043753_builder_junos_151_r1]
JUNOS Services Captive Portal and Content Delivery Container package
[20150618.043753_builder_junos_151_r1]
JUNOS Border Gateway Function package [20150618.043753_builder_junos_151_r1]
JUNOS AppId Services [20150618.043753_builder_junos_151_r1]
JUNOS Services Application Level Gateways
[20150618.043753_builder_junos_151_r1]
JUNOS Services ACL Container package [20150618.043753_builder_junos_151_r1]
JUNOS Packet Forwarding Engine Support (MX/EX92XX Common)
[20150618.043753_builder_junos_151_r1]
JUNOS Packet Forwarding Engine Support (M/T Common)
[20150618.043753_builder_junos_151_r1]
JUNOS Online Documentation [20150618.043753_builder_junos_151_r1]
JUNOS FIPS mode utilities [20150618.043753_builder_junos_151_r1]
```



NOTE: The output shows the OS kernel, OS runtime, and other packages installed on the router.

Installing Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Different Release



NOTE: If you have important files in other directories, copy them from the router or switch to a secure location before upgrading the router or switch.



NOTE: The following procedure refers to routers, but it also applies to switches.

To install Junos OS with upgraded FreeBSD over Junos OS with upgraded FreeBSD of an earlier release:

1. Enter the **request system software add *package-name* validate reboot** command from the operational mode in the CLI:



NOTE: The **no-copy** option is enabled by default.

Use the **validate** and **reboot** options with the **request system software add** command. The command uses the **validate** option by default. We encourage users to validate

using the **validate** option when upgrading from Junos OS to Junos OS or from Junos OS with upgraded FreeBSD to Junos OS with upgraded FreeBSD.

If you leave out the **reboot** option, you can take care of that in a separate reboot step.

The new Junos OS image is installed on the router.

2. Verify the installation of Junos OS with upgraded FreeBSD.

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



NOTE: The output shows the OS kernel, OS runtime, and other packages installed on the router.

To Install Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Later Release

To install Junos OS with upgraded FreeBSD over Junos OS with upgraded FreeBSD of a different release:

1. Enter the **request system software add package-name validate reboot** command from the operational mode in the CLI:



NOTE: The **no-copy** option is enabled by default.

Use the **validate** and **reboot** options with the **request system software add** command. The command uses the **validate** option by default. We encourage users to validate using the **validate** option when upgrading from Junos OS to Junos OS, or from Junos OS with upgraded FreeBSD to Junos OS with upgraded FreeBSD.

If you leave out the **reboot** option, you can take care of that in a separate reboot step.

The new Junos OS image is installed on the router.

2. Verify the installation of Junos OS with upgraded FreeBSD:

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



NOTE: You need to install the Junos OS software package and host software package on the routers with the RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines. For upgrading the host OS and Junos OS on these routers with VM host support, use the `junos-vmhost-install-x.tgz` image and specify the name of the regular package in the **request vmhost software add** command. For more information, see the [“VM Host Installation” on page 268](#) topic in *Installation and Upgrade Guide*.



NOTE: The output shows the OS kernel, OS runtime, and other packages installed on the router.

**Related
Documentation**

- [Downgrading Junos OS from Upgraded FreeBSD on page 249](#)
- [Understanding Junos OS with Upgraded FreeBSD on page 17](#)
- [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 414](#)
- [request system reboot \(Junos OS with Upgraded FreeBSD\) on page 398](#)
- [Establishing an SSH Connection for a NETCONF Session](#)

Understanding Junos OS Upgrades for SRX Series Devices

SRX Series devices are delivered with Junos OS preinstalled on them. When you power on a device, it starts (boots) up using its primary boot device. These devices also support secondary boot devices, allowing you to back up your primary boot device and configuration.

As new features and software fixes become available, you must upgrade Junos OS to use them. Before an upgrade, we recommend that you back up your primary boot device.

Understanding Junos OS Upgrades

On a services gateway, you can configure the primary or secondary boot device with a snapshot of the current configuration, default factory configuration, or rescue configuration. You can also replicate the configuration for use on another device.

If the SRX Series device does not have a secondary boot device configured and the primary boot device becomes corrupted, you can reload the Junos OS package onto the corrupted internal media from a USB flash drive or TFTP server.

Junos OS Upgrade Methods on the SRX Series Devices

SRX Series devices that ship from the factory with Junos OS Release 10.0 or later are formatted with the dual-root partitioning scheme.



NOTE: Junos OS Release 12.1X45 and later do not support single root partitioning.



NOTE: SRX100, SRX110, SRX210, SRX220, and SRX240 devices with 2 GB RAM cannot be upgraded to any Junos OS 12.1X46 Release after 12.1X46-D65. Attempting to upgrade to this release on devices with 2 GB RAM will trigger the following error: **ERROR: Unsupported platform for 12.1X46 releases after 12.1X46-D65**

Existing SRX Series devices that are running Junos OS Release 9.6 or earlier use the single-root partitioning scheme. While upgrading these devices to Junos OS Release 10.0 or later, you can choose to format the storage media with dual-root partitioning (strongly recommended) or retain the existing single-root partitioning.

Certain Junos OS upgrade methods format the internal media before installation, whereas other methods do not. To install Junos OS Release 10.0 or later with the dual-root partitioning scheme, you must use an upgrade method that formats the internal media before installation.



NOTE: If you are upgrading to Junos OS Release 10.0 without transitioning to dual-root partitioning, use the conventional CLI and J-Web user interface installation methods.

These upgrade methods format the internal media before installation:

- Installation from the boot loader using a TFTP server
- Installation from the boot loader using a USB storage device
- Installation from the CLI using the **partition** option (available in Junos OS Release 10.0)
- Installation using the J-Web user interface

These upgrade methods retain the existing partitioning scheme:

- Installation using the CLI
- Installation using the J-Web user interface



WARNING: Upgrade methods that format the internal media before installation wipe out the existing contents of the media. Only the current configuration is preserved. Any important data must be backed up before starting the process.



NOTE: Once the media has been formatted with the dual-root partitioning scheme, you can use conventional CLI or J-Web user interface installation methods, which retain the existing partitioning and contents of the media, for subsequent upgrades.

Release History Table

Release	Description
12.1X46	SRX100, SRX110, SRX210, SRX220, and SRX240 devices with 2 GB RAM cannot be upgraded to any Junos OS 12.1X46 Release after 12.1X46-D65. Attempting to upgrade to this release on devices with 2 GB RAM will trigger the following error: ERROR: Unsupported platform for 12.1X46 releases after 12.1X46-D65
12.1X45-D10	Junos OS Release 12.1X45 and later do not support single root partitioning

Related Documentation

- [Software Naming Convention for SRX Series Devices on page 8](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
- [Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server on page 181](#)

Preparing Your SRX Series Device for Junos OS Upgrades

Before you begin upgrading Junos OS on an SRX Series device, ensure the following:

- Obtain a Juniper Networks Web account and a valid support contract. You must have an account to download software upgrades. To obtain an account, complete the registration form at the Juniper Networks website:
<https://www.juniper.net/registration/Register.jsp>.
- Back up your primary boot device onto a secondary storage device.

Creating a backup has the following advantages:

- The device can boot from backup and come back online in case of failure or corruption of the primary boot device in the event of power failure during an upgrade.
- Your active configuration files and log files are retained.
- The device can recover from a known, stable environment in case of an unsuccessful upgrade.

You can use either the J-Web user interface or the CLI to back up the primary boot device on the secondary storage device.

Secondary Storage Devices Available on SRX Series Devices

You can use either the J-Web user interface or the CLI to back up the primary boot device on the secondary storage device.

Table 20 on page 163 lists the secondary storage devices available on an SRX Series devices.

Table 20: Secondary Storage Devices for SRX Series Devices

Storage Device	Available on Services Gateways	Minimum Storage Required
USB storage device	SRX100, SRX110, SRX210, SRX220, and SRX240 Services Gateways	1 GB
	SRX550, and SRX650 Services Gateway	2 GB
	SRX300, SRX320, SRX340, and SRX345 Services Gateways	1 GB
	SRX550M	2 GB
External CompactFlash (CF)	SRX650 Services Gateway	2 GB
SSD	SRX340, SRX345, SRX550M, and SRX1500	100 GB
SSD	SRX4100, SRX4200	240 GB
Routing Engine (RE2) SSD card	SRX5000 line devices	120 GB



NOTE:

- During a successful upgrade, the upgrade package completely reinstalls the existing Junos OS. It retains configuration files, log files, and similar information from the previous version.
- After a successful upgrade, remember to back up the new current configuration to the secondary device.

Verifying Available Disk Space on SRX Series Devices

The amount of free disk space necessary to upgrade a device with a new version of Junos OS can vary from one release to another. Check the Junos OS software version you are installing to determine the free disk space requirements.

If the amount of free disk space on a device is insufficient for installing Junos OS, you might receive a warning similar to the following messages, that the /var filesystem is low on free disk space:

WARNING: The /var filesystem is low on free disk space.

WARNING: This package requires 1075136k free, but there is only 666502k available.

To determine the amount of free disk space on the device, issue the **show system storage detail** command. The command output displays statistics about the amount of free disk space in the device file systems.

A sample of the **show system storage detail** command output is shown below:

```
user> show system storage detail
```

Filesystem	1024-blocks	Used	Avail	Capacity	Mounted on
/dev/da0s2a	300196	154410	121772	56%	/
devfs	1	1	0	100%	/dev
/dev/md0	409000	409000	0	100%	/junos
/cf	300196	154410	121772	56%	/junos/cf
devfs	1	1	0	100%	/junos/dev/
procfs	4	4	0	100%	/proc
/dev/bo0s3e	25004	52	22952	0%	/config
/dev/bo0s3f	350628	178450	144128	55%	/cf/var
/dev/md1	171860	16804	141308	11%	/mfs
/cf/var/jail	350628	178450	144128	55%	/jail/var
/cf/var/log	350628	178450	144128	55%	/jail/var/log
devfs	1	1	0	100%	/jail/dev
/dev/md2	40172	4	36956	0%	/mfs/var/run/utm
/dev/md3	1884	138	1596	8%	/jail/mfs

Cleaning Up the System File Storage Space

When the system file storage space on the device is full, rebooting the device does not solve the problem. The following error message is displayed during a typical operation on the device after the file storage space is full.

```
user@host% cli
user@host> configure/var: write failed, filesystem is full
```

You can clean up the file storage on the device by deleting system files using the **request system storage cleanup** command as shown in following procedure:

1. Request to delete system files on the device.

```
user@host> request system storage cleanup
```

The list of files to be deleted is displayed.

List of files to delete:

Size	Date	Name
11B	Oct 28 23:40	/var/jail/tmp/alarmd.ts
92.4K	Jan 11 17:12	/var/log/chassisd.0.gz
92.4K	Jan 11 06:06	/var/log/chassisd.1.gz
92.5K	Jan 10 19:00	/var/log/chassisd.2.gz
92.5K	Jan 10 07:53	/var/log/chassisd.3.gz
92.2K	Jan 10 15:00	/var/log/hostlogs/auth.log.1.gz
92.2K	Jan 1 18:45	/var/log/hostlogs/auth.log.2.gz
92.1K	Jan 4 17:30	/var/log/hostlogs/auth.log.3.gz
92.2K	Jan 1 18:45	/var/log/hostlogs/auth.log.4.gz
79.0K	Jan 12 01:59	/var/log/hostlogs/daemon.log.1.gz
78.8K	Jan 11 23:15	/var/log/hostlogs/daemon.log.2.gz
78.7K	Jan 11 20:30	/var/log/hostlogs/daemon.log.3.gz


```

79.1K Jan 11 17:44 /var/log/hostlogs/daemon.log.4.gz
59.1K Jan 11 21:59 /var/log/hostlogs/debug.1.gz
59.2K Jan 11 17:44 /var/log/hostlogs/debug.2.gz
59.2K Jan 11 13:29 /var/log/hostlogs/debug.3.gz
59.3K Jan 11 09:14 /var/log/hostlogs/debug.4.gz
186.6K Oct 20 16:31 /var/log/hostlogs/kern.log.1.gz
238.3K Jan 11 23:15 /var/log/hostlogs/lcmd.log.1.gz
238.4K Jan 11 17:30 /var/log/hostlogs/lcmd.log.2.gz
238.6K Jan 11 11:45 /var/log/hostlogs/lcmd.log.3.gz
238.5K Jan 11 06:00 /var/log/hostlogs/lcmd.log.4.gz
372.5K Jan 11 17:00 /var/log/hostlogs/syslog.1.gz
372.5K Jan 11 04:45 /var/log/hostlogs/syslog.2.gz
371.9K Jan 10 16:30 /var/log/hostlogs/syslog.3.gz
372.7K Jan 10 04:15 /var/log/hostlogs/syslog.4.gz
10.1K Jan 12 02:03 /var/log/messages.0.gz
55.1K Jan 6 21:25 /var/log/messages.1.gz
81.5K Dec 1 21:30 /var/log/messages.2.gz

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

```

2. Enter the option **yes** to proceed with deleting of the files.

Understanding Junos OS with Upgraded FreeBSD for SRX5400, SRX5600, and SRX5800 Devices

Junos OS with an upgraded FreeBSD kernel provides a clean-slate implementation of Junos OS on top of a pristine (minimally modified) and current version of the FreeBSD OS.

The SRX devices currently running Junos OS with upgraded FreeBSD are listed in [Table 21 on page 165](#).

Table 21: Upgraded FreeBSD Kernel Support by Hardware Platform

Platforms	CPU Type	Release Introduced
SRX5400	Intel	17.3
SRX5600	Intel	17.3
SRX5800	Intel	17.3

The major processing changes are as follows:

- Interactions between Junos OS and the upgraded FreeBSD kernel use well-established interfaces because Junos OS is now layered on a minimally modified and current version of FreeBSD.
- Symmetric multiprocessing is enabled and compiled for 64-bit kernel.
- FreeBSD provides a consistent runtime environment for platforms that run Junos OS.
- Upgraded FreeBSD image also has FIPS mode as a configuration mode and eliminates the need for a separate FIPS package.

Limitations

The following are the limitations of the upgraded FreeBSD support on SRX5400, SRX5600, and SRX5800 devices:

- Upgraded FreeBSD is supported only on RE-1800X4 type.
- The underlying FreeBSD is 64 bit, while there are specific 32-bit processes and utilities.
- ISSU is not supported from older version of FreeBSD to upgraded FreeBSD. However, it is supported between upgraded FreeBSDs.

There are also major changes in file structures and software packages. These changes are as follows:

- New packages use XML description files instead of scripts.
- Multiple package sets (a collection of installed packages) are stored on the device at the same time. Sets can be active (the currently used set), pending (the set that should be used at the next reboot), or previous (a formerly active set). Non-recovery snapshots (but not recoverable image snapshots) are available for the package sets to preserve package content lists.

Understanding Changes in Operations and Management for Junos OS with Upgraded FreeBSD for SRX5400, SRX5600, and SRX5800 devices

There is now a separate Operations, Administration, and Maintenance (OAM) volume (**oam**) distinct from the Junos OS volume (**junos**).

One major change is the distinction between recovery snapshots and non-recovery snapshots.

The major characteristics of the recovery snapshots are as follows:

- Recovery snapshots are full copies of the packages and configuration taken at the time the snapshot command is issued.
- Recovery snapshots reside on the OAM volume or USB storage device.

The major characteristics of the non-recovery snapshots are as follows:

- Non-recovery snapshots are snapshots residing on the Junos OS volume that refer to the current running set of packages and a copy of the configuration at the time the snapshot command is issued.
- Non-recovery snapshots do not need to copy the whole Junos OS installation and so are very fast.
- Non-recovery snapshots can be requested as the boot image for the next reboot.

The upgraded FreeBSD kernel requires changes to several commands and statements and their related parameters. The new and changed actions are summarized in [Table 6 on page 19](#). For details of the changes, see the topics covering the specific command or statement.

Table 22: New and Changed Commands and Statements for Junos OS with Upgraded FreeBSD

Command or Statement	Release Introduced	Change
<code>request system snapshot delete <i>snapshot</i></code>	17.3	New action
<code>request system snapshot recovery</code>	17.3	New action
<code>request system snapshot load <i>snapshot</i></code>	17.3	New action
<code>request system recover <i>volume</i></code>	17.3	New action: <i>volume</i> is either junos-volume or oam-volume
<code>request system snapshot</code>	17.3	Changed action
<code>show system snapshot</code>	17.3	Changed action

The new FreeBSD kernel also requires that several commands and statements are now deprecated. In some cases, these commands and statements generate an error, and, in other cases, the result is appropriate for the new kernel. The deprecated commands and statements are summarized in [Table 7 on page 20](#). For details, see the topics covering the specific command or statement.

Table 23: Deprecated Commands and Statements for Junos OS with Upgraded FreeBSD

Deprecated Command or Configuration Statement	Release Deprecated
Deprecated Command	
<code>request system partition abort</code>	17.3
<code>request system partition compact-flash</code>	17.3
<code>request system partition hard-disk</code>	17.3
<code>request system snapshot <config-partition></code>	17.3
<code>request system snapshot <root-partition></code>	17.3
<code>request system snapshot <slice></code>	17.3
<code>request system software delete-backup</code>	17.3
<code>request system software rollback <force></code>	17.3
<code>show system processes providers</code>	17.3
<code>show system snapshot <slice></code>	17.3
Deprecated Configuration Statement	

Table 23: Deprecated Commands and Statements for Junos OS with Upgraded FreeBSD (*continued*)

Deprecated Command or Configuration Statement	Release Deprecated
<code>set system mirror-flash-on-disk</code>	17.3

Understanding Junos OS with Upgraded FreeBSD Package Names

Starting with Junos OS Release 17.3, certain hardware platforms run a Junos OS based on an upgraded FreeBSD kernel (hereafter called Junos OS with upgraded FreeBSD). In releases earlier than Junos OS Release 17.3, software packages came in several major software package categories, such as domestic, worldwide, or Federal Information Processing Standard (FIPS). However, Junos OS with upgraded FreeBSD has a new naming convention: There is only one category, and FIPS, instead of being a separate category, is an option you select on installation. This topic describes the simplified naming convention for Junos OS with upgraded FreeBSD.

If your hardware platform is listed in the table in [“Understanding Junos OS with Upgraded FreeBSD” on page 17](#), then you must use the new package names for download and installation.

The components of the new package naming conventions are as follows:

- **Prefix**—This is **junos-install**. This prefix takes the place of the prefix **junos-srx5000**.
- **Media keyword**—Added to the prefix, a media keyword is only used when the image is not for use with the **request system software add** command. Values for the **media** keyword include **usb** for images installed from a USB drive or **net** for images installed from the loader prompt; for example, the entire prefix of your package might be **junos-install-media-net** or **junos-install-media-usb**.
- **Architecture**—This field indicates the CPU architecture of the platforms. Values include **x86** for Intel and **arm** for Advanced RISC Machines CPUs.
- **Application binary interface (ABI)**—This field indicates the “word length” of the CPU architecture. Values include **32** for 32-bit architectures and **64** for 64-bit architectures.
- **Release**—This field indicates the release number, such as **17.3**.
- **Edition**—The edition field is null (empty) for the standard (domestic) images. For jurisdictions with limits on dataplane encryption, this field is set to **limited**.

As before, all images are in .tar and .tgz(.tar and .tgz) format.



NOTE: There are no longer “export” worldwide images or separate FIPS images. The keyword “signed” no longer appears because all Junos OS images are signed for validation.

Examples of valid Junos OS software package names include the following:

- **junos-install-srx-x86-64-17.3-20170725.0.tgz**—An image for a SRX5400, SRX5600, and SRX5800 devices.
- **junos-install-media-usb-srx-x86-64-17.3-20170725.0.img.gz**—An image stored on and installed from a USB flash drive for SRX5400, SRX5600, and SRX5800 devices.

Upgrading Junos OS with Upgraded FreeBSD

Starting in Junos OS Release 17.3, certain hardware platforms run an upgraded FreeBSD kernel instead of older versions of FreeBSD.

Upgrading to Junos OS with upgraded FreeBSD reformats the file system. Only specific files and directories are preserved unless precautions are taken. By default, the upgrade process only preserves the following directories:

- **/config**
- **/etc/localtime**
- **/var/db**
- **/var/etc/master.passwd**
- **/var/etc/inetd.conf**
- **/var/etc/pam.conf**
- **/var/etc/resolv.conf**
- **/var/etc/syslog.conf**
- **/var/etc/localtime**
- **/var/etc/exports**
- **/var/etc/extensions.allow**
- **/var/preserve**
- **/var/tmp/baseline-config.conf**
- **/var/tmp/preinstall_boot_loader.conf**

Table 24: Upgrade Path to Junos OS with the Upgraded FreeBSD

Current Security Junos OS Release	Upgrade Path	Example
12.3X48 to 17.3	Upgrade in a single step.	To upgrade from Release 12.3X48, upgrade directly to Release 17.3.
15.1X49 to 17.3	Upgrade in a single step.	To upgrade from Release 15.1X49, upgrade directly to Release 17.3.



NOTE: To upgrade from any prior Releases to 12.3X48 (such as 12.1X47), it is required to do two step upgrade. First upgrade to either Junos OS Release 12.3X48 or 15.1X49 and then upgrade to Release 17.3.

Installing Junos OS with Upgraded FreeBSD Over Plain Junos OS

To install Junos OS with upgraded FreeBSD over plain Junos OS:

1. Enter the **request system software add package-name no-validate** command from the operational mode in the CLI:

Use the **no-validate** option with the **request system software add** command. If you leave out the **no-validate** option, the command uses the **validate** option by default, and direct validation of the running configuration does not work for upgrading to Junos OS with upgraded FreeBSD from Junos OS based on older versions of the FreeBSD kernel.



NOTE: You can also use the **reboot** option along with the **request system software add** command, but it is not recommended to do this in a single step while upgrading from a FreeBSD 6.1 based Junos OS to FreeBSD 10 based Junos OS.



NOTE: To validate the current configuration on an upgrade to Junos OS with upgraded FreeBSD from Junos OS, use the **request system software validate on (Junos OS with Upgraded FreeBSD)** command.

```
user@host>request system software add no-copy no-validate
junos-install-srx-x86-64-17.3R1.tgz reboot
Verified junos-install-srx-x86-64-17.3R1 signed by PackageProductionEc_2017
method ECDSA256+SHA256
Verified manifest signed by PackageProductionEc_2017 method ECDSA256+SHA256
Checking PIC combinations
Verified fips-mode signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding fips-mode-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jail-runtime signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jail-runtime-x86-32-20170725.352915_builder_stable_10 ...
Verified jdocs signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding jdocs-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jfirmware signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding jfirmware-x86-32-17.3R1 ...
Verified jpfe-X signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding jpfe-X-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jpfe-X960 signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding jpfe-X960-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jpfe-common signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jpfe-common-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jpfe-fips signed by PackageProductionEc_2017 method ECDSA256+SHA256
Verified jpfe-wrlinux signed by PackageProductionEc_2017 method
```

```
ECDSA256+SHA256
Adding jpfe-wrlinux-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jsd-jet-1 signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding jsd-x86-32-17.3R1-jet-1 ...
Verified jservices-crypto signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-crypto-x86-32-17.3R1 ...
Verified jservices-crypto-base signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-crypto-base-x86-32-20170728.153050_builder_junos_173_r1
...
Verified jservices-ipsec signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-ipsec-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-ssl signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-ssl-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding jservices-x86-32-17.3R1 ...
Verified jservices-aac1 signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-aac1-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-alg signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-alg-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-appid signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-appid-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-cos signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-cos-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-cpcd signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-cpcd-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-hcm signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-hcm-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-idp signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-idp-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-ids signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-ids-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-jdpi signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-jdpi-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-jflow signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-jflow-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-llpdf signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-llpdf-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-lrf signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-lrf-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-mobile signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-mobile-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-mss signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-mss-x86-32-20170728.153050_builder_junos_173_r1 ...
```

```
Verified jservices-nat signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-nat-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-pcef signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-pcef-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-ptsp signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-ptsp-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-rpm signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-rpm-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-sfw signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-sfw-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jservices-traffic-dird signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-traffic-dird-x86-32-20170728.153050_builder_junos_173_r1
...
Verified jservices-ur1f signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding jservices-ur1f-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jsim-pfe signed by PackageProductionEc_2017 method ECDSA256+SHA256
Verified junos-daemons-srx signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-daemons-srx-x86-64-20170728.153050_builder_junos_173_r1 ...
Verified junos-daemons signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-daemons-x86-64-20170728.153050_builder_junos_173_r1 ...
Verified junos-dp-crypto-support-srx signed by PackageProductionEc_2017
method ECDSA256+SHA256
Adding
junos-dp-crypto-support-srx-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified junos-libs-compat32-srx signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-libs-compat32-srx-x86-64-20170728.153050_builder_junos_173_r1
...
Verified junos-libs-compat32 signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-libs-compat32-x86-64-20170728.153050_builder_junos_173_r1 ...
Verified junos-libs-srx signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-libs-srx-x86-64-20170728.153050_builder_junos_173_r1 ...
Verified junos-libs signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding junos-libs-x86-64-20170728.153050_builder_junos_173_r1 ...
Verified junos-modules-srx signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-modules-srx-x86-64-20170728.153050_builder_junos_173_r1 ...
Verified junos-modules signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-modules-x86-64-20170728.153050_builder_junos_173_r1 ...
Verified junos-net-prd signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-net-prd-x86-64-20170728.153050_builder_junos_173_r1 ...
Verified junos-platform-srx signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-platform-srx-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified junos-platform signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-platform-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified junos-runtime-srx signed by PackageProductionEc_2017 method
```



```

ECDSA256+SHA256
Adding junos-runtime-srx-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified junos-runtime signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding junos-runtime-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified junos-vmguest-mtx signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Verified jweb-ex signed by PackageProductionEc_2017 method ECDSA256+SHA256
Verified jweb-srx signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding jweb-srx-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified jweb signed by PackageProductionEc_2017 method ECDSA256+SHA256
Verified oam signed by PackageProductionEc_2017 method ECDSA256+SHA256
Verified os-compatible32 signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding os-compatible32-x86-64-20170725.352915_builder_stable_10 ...
Verified os-crypto signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding os-crypto-x86-64-20170725.352915_builder_stable_10 ...
Verified os-kernel-prd signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding os-kernel-prd-x86-64-20170725.352915_builder_stable_10 ...
Verified os-libs-compatible32 signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding os-libs-compatible32-10-x86-64-20170725.352915_builder_stable_10 ...
Verified os-libs signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding os-libs-10-x86-64-20170725.352915_builder_stable_10 ...
Verified os-runtime signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding os-runtime-x86-64-20170725.352915_builder_stable_10 ...
Verified os-vmguest signed by PackageProductionEc_2017 method ECDSA256+SHA256
Verified os-zoneinfo signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding os-zoneinfo-x86-64-20170725.352915_builder_stable_10 ...
Verified py-base signed by PackageProductionEc_2017 method ECDSA256+SHA256
Adding py-base-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified py-extensions signed by PackageProductionEc_2017 method
ECDSA256+SHA256
Adding py-extensions-x86-32-20170728.153050_builder_junos_173_r1 ...
Verified vrr-mx signed by PackageProductionEc_2017 method ECDSA256+SHA256
NOTICE: 'pending' set will be activated at next reboot...
Rebooting. Please stop cron.
Waiting for PIDS: 19772.

```

The new Junos OS image is installed on the router.

2. Reboot the device to start the new software using the **request system reboot** command:

```

user@host> request system reboot
Reboot the system? [yes, no] (no) yes

```

The software is loaded when you reboot the system. Installation can take between 5 and 10 minutes. The device then reboots from the boot device on which the software was just installed. When the reboot is complete, the device displays the login prompt.

While the software is being upgraded, the Routing Engine on which you are performing the installation does not route traffic.

3. Log in and issue the **show version** command to verify the version of the software installed.

```

user@host> show version

```

```
Hostname: dpidev-srx5k-03
Model: srx5400
Junos: 17.3R1
JUNOS OS Kernel 64-bit [20170725.352915_builder_stable_10]
JUNOS OS libs [20170725.352915_builder_stable_10]
JUNOS OS runtime [20170725.352915_builder_stable_10]
JUNOS OS time zone information [20170725.352915_builder_stable_10]
JUNOS OS libs compat32 [20170725.352915_builder_stable_10]
JUNOS OS 32-bit compatibility [20170725.352915_builder_stable_10]
JUNOS py extensions [20170728.153050_builder_junos_173_r1]
JUNOS py base [20170728.153050_builder_junos_173_r1]
JUNOS OS crypto [20170725.352915_builder_stable_10]
JUNOS network stack and utilities [20170728.153050_builder_junos_173_r1]
JUNOS modules [20170728.153050_builder_junos_173_r1]
JUNOS srx modules [20170728.153050_builder_junos_173_r1]
JUNOS libs [20170728.153050_builder_junos_173_r1]
JUNOS libs compat32 [20170728.153050_builder_junos_173_r1]
JUNOS runtime [20170728.153050_builder_junos_173_r1]
JUNOS Web Management Platform Package [20170728.153050_builder_junos_173_r1]
JUNOS srx libs compat32 [20170728.153050_builder_junos_173_r1]
JUNOS srx runtime [20170728.153050_builder_junos_173_r1]
JUNOS common platform support [20170728.153050_builder_junos_173_r1]
JUNOS srx platform support [20170728.153050_builder_junos_173_r1]
JUNOS srx libs [20170728.153050_builder_junos_173_r1]
JUNOS srx Data Plane Crypto Support [20170728.153050_builder_junos_173_r1]
JUNOS daemons [20170728.153050_builder_junos_173_r1]
JUNOS srx daemons [20170728.153050_builder_junos_173_r1]
JUNOS Services URL Filter package [20170728.153050_builder_junos_173_r1]
JUNOS Services TLB Service PIC package [20170728.153050_builder_junos_173_r1]
JUNOS Services SSL [20170728.153050_builder_junos_173_r1]
JUNOS Services Stateful Firewall [20170728.153050_builder_junos_173_r1]
JUNOS Services RPM [20170728.153050_builder_junos_173_r1]
JUNOS Services PTSP Container package [20170728.153050_builder_junos_173_r1]
JUNOS Services PCEF package [20170728.153050_builder_junos_173_r1]
JUNOS Services NAT [20170728.153050_builder_junos_173_r1]
JUNOS Services Mobile Subscriber Service Container package
[20170728.153050_builder_junos_173_r1]
JUNOS Services MobileNext Software package
[20170728.153050_builder_junos_173_r1]
JUNOS Services Logging Report Framework package
[20170728.153050_builder_junos_173_r1]
JUNOS Services LL-PDF Container package
[20170728.153050_builder_junos_173_r1]
JUNOS Services Jflow Container package [20170728.153050_builder_junos_173_r1]
JUNOS Services Deep Packet Inspection package
[20170728.153050_builder_junos_173_r1]
JUNOS Services IPSec [20170728.153050_builder_junos_173_r1]
JUNOS Services IDS [20170728.153050_builder_junos_173_r1]
JUNOS IDP Services [20170728.153050_builder_junos_173_r1]
JUNOS Services HTTP Content Management package
[20170728.153050_builder_junos_173_r1]
JUNOS Services Crypto [20170728.153050_builder_junos_173_r1]
JUNOS Services Captive Portal and Content Delivery Container package
[20170728.153050_builder_junos_173_r1]
JUNOS Services COS [20170728.153050_builder_junos_173_r1]
JUNOS AppId Services [20170728.153050_builder_junos_173_r1]
JUNOS Services Application Level Gateways
[20170728.153050_builder_junos_173_r1]
JUNOS Services ACL Container package [20170728.153050_builder_junos_173_r1]
JUNOS Extension Toolkit [20170728.153050_builder_junos_173_r1]
JUNOS Packet Forwarding Engine Support (wrlinux)
```

```
[20170728.153050_builder_junos_173_r1]
JUNOS Packet Forwarding Engine Support (MX/EX92XX Common)
[20170728.153050_builder_junos_173_r1]
JUNOS Packet Forwarding Engine Support (M/T Common)
[20170728.153050_builder_junos_173_r1]
JUNOS Packet Forwarding Engine Support (MX Common)
[20170728.153050_builder_junos_173_r1]
JUNOS jfirmware [20170728.153050_builder_junos_173_r1]
JUNOS Online Documentation [20170728.153050_builder_junos_173_r1]
JUNOS jail runtime [20170725.352915_builder_stable_10]
JUNOS FIPS mode utilities [20170728.153050_builder_junos_173_r1]
```



NOTE: The output shows the OS kernel, OS runtime, and other packages installed on the router.

Understanding Junos OS with Upgraded FreeBSD Snapshots

One major change between Junos OS and Junos OS with Upgraded FreeBSD is the distinction between recovery snapshots and non-recovery snapshots.

The major characteristics of the recovery snapshots are as follows:

- Recovery snapshots are full copies of the packages and configuration taken at the time the snapshot command is issued.
- Recovery snapshots reside on the OAM volume or USB medium.

The major characteristics of the non-recovery snapshots are as follows:

- Non-recovery snapshots are snapshots residing on the Junos OS volume that refer to the current running set of packages and a copy of the configuration at the time the snapshot command is issued.
- Non-recovery snapshots do not need to copy the whole Junos OS installation and so are very fast.
- Non-recovery snapshots can be requested as the boot image for the next reboot.

Recovery snapshots are full copies of the packages and configuration taken at the time the snapshot command is issued. Recovery snapshots reside on the OAM volume or USB medium. Recovery snapshots take some time to complete because of the level of detail captured. Recovery snapshots can be used to recover the Junos OS volume. There is only one recovery snapshot on the system.

On the other hand, non-recovery snapshots are snapshots residing on the Junos OS volume that refer to the current running set of packages and a copy of the configuration at the time the snapshot command is issued. Non-recovery snapshots do not need to copy the whole Junos OS installation and so are very fast. They also consume little space, except for the **config.tgz** file. Non-recovery snapshots can be requested as the boot image for the next reboot. You can rename non-recovery snapshots and retain more than one. You rename the non-recovery snapshots with the same procedure used to rename any other file on the system.



NOTE: We recommend that you generate both a non-recovery and a recovery snapshot after you successfully upgrade to Junos OS with upgraded FreeBSD. These snapshots should be refreshed periodically.

Package sets relate to non-recovery and recovery snapshots. The **/active**, **/pending**, and **/previous** sets are all package sets. A non-recovery snapshot is also a package set in a sense, with the addition of a copy of the configuration at the time that the non-recovery snapshot is taken.

Packages that are no longer referenced by any package set or non-recovery snapshot are automatically deleted. We recommend deleting any old non-recovery snapshots after an upgrade so that old packages can be deleted and space recovered.

Some helpful commands for non-recovery snapshots are:

- **request system snapshot**—Use this command to create a non-recovery snapshot.
- **show system snapshot**—Use this command to list all the available non-recovery snapshots.
- **request system snapshot delete**—Use this command to delete a non-recovery snapshot.

A helpful command for recovery snapshots is:

- **request system snapshot recovery**—Use this command to create a recovery snapshot. You can use other parameters to determine the details of the recovery snapshot created. There is only one recovery snapshot on the system.

Understanding Junos OS with Upgraded FreeBSD Disk Volumes for SRX5400, SRX5600, and SRX5800 devices

Starting in Junos OS Release 17.3, certain hardware platforms have a new disk naming convention. These platforms run a Junos OS based on an upgraded FreeBSD kernel instead of older versions of FreeBSD.

The hardware platforms listed in the table in “[Understanding Junos OS with Upgraded FreeBSD](#)” on page 17 have two volumes. Technically, these volumes are **dev/gpt/oam** and **dev/gpt/junos**, but the short forms (**/junos** and **/oam**) are used in this topic. Because the **/junos** and **/oam** volumes have very different purposes, their content is different. Essentially, the **/junos** volume is used for running device software and holds configuration information and logs, whereas the **/oam** volume is used for backup copies of everything needed in the event that the **/junos** volume fails.

The **/junos** volume contains a directory named **/packages/db** that has all the components present on the device, such as **os-kernel-123**, **os-kernel-456**, and so on. A sibling directory named **/packages/sets** is also present. Package sets are an important concept in Junos OS with upgraded FreeBSD.

The **/packages/sets** directory contains a package listing that gathers all the components of the running Junos OS into an XML format in the **/active** subdirectory. So **os-kernel-123** could be a component in the **/packages/sets/active** subdirectory, but then **os-kernel-456** could not be in the same XML package. Package sets do not contain the kernel software itself (for example), but tell the device where to find the kernel component needed for the software package. The same kernel can be present in several package listings, but only one package can be active and running on the device at any given time.

There are several directories on the **/junos** volume where a particular software package listing can be found:

- **/previous**—The package set in this directory contains the list of all the components that ran on the device before the last upgrade.
- **/active**—The package set in this directory contains the list of all the software components currently running on the device.
- **/pending**—The package set in this directory contains the list of all the software components on the device that will run after the next reboot.



NOTE: After a successful reboot, the package set in the **/pending** directory becomes the active package set, and the package set in the **/active** directory becomes the previous set.

The **/junos** volume also contains non-recovery snapshots taken with the **request system snapshot** command. These types of snapshots are new to Junos OS with upgraded FreeBSD and cannot be used for recovery of a failed system. Non-recovery snapshots are a special type of package set that includes a copy of the configuration. There can be many non-recovery snapshots on the device, and the files can be renamed. Multiple non-recovery snapshots, essentially lists of software components and configuration files, can be helpful when major software or configuration changes are occurring and establishment of a known stable system baseline is required.

The compact flash drive is the **/oam** volume and stores recovery snapshot backup information. In case of failure of the main drive (that is, the **/junos** volume), the **/oam** volume can be used to boot the system. In order to perform this reboot, the **/oam** volume needs to have all of the information required to provide the system with a running configuration. This information is provided by the recovery snapshot, created with the **request system snapshot recovery** command. Although it can take a while to perform, the recovery snapshot establishes an **.izo** or **.iso** image of the running Junos OS.

In the case of a total failure of the **/junos** volume, the system can be booted from the **/oam** volume. The recovery snapshot can then restore the repaired system.

Downloading Software Packages from Juniper Networks

To download Junos OS upgrades from Juniper Networks:

1. Using a Web browser, follow the links to the download URL on the Juniper Networks webpage. Depending on your location, select the Canada and U.S. version (domestic) or the Worldwide version (ww):
 - <https://www.juniper.net/support/downloads/junos.html>
 - <https://www.juniper.net/support/downloads/junos.html>
2. Log in to the Juniper Networks website using the username (generally your e-mail address) and password supplied by your Juniper Networks representative.
3. Select the appropriate software image for your platform.
4. Download Junos OS to a local host or to an internal software distribution site.

Related Documentation

- [Understanding Junos OS Upgrades for SRX Series Devices on page 160](#)
- [Preparing Your SRX Series Device for Junos OS Upgrades on page 162](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
- [Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server on page 181](#)

Example: Installing Junos OS Upgrade Packages on SRX Series Devices

This example shows how to install Junos OS upgrades on SRX Series devices.

- [Requirements on page 178](#)
- [Overview on page 178](#)
- [Configuration on page 179](#)
- [Verification on page 180](#)

Requirements

Before you begin:

- Verify the available space on the internal media. See “[Preparing Your SRX Series Device for Junos OS Upgrades](#)” on page 162 and the *Junos OS Release Notes*
- Download the software package. See “[Downloading Software Packages from Juniper Networks](#)” on page 177.
- Copy the software package to the device if you are installing the software package from a local directory on the device. We recommend that you copy it to the `/var/tmp` directory.

Overview

By default, the **request system software add *package-name*** command uses the `validate` option to validate the software package against the current configuration as a prerequisite

to adding the software package. This validation ensures that the device can reboot successfully after the software package is installed. This is the default behavior when you are adding a software package.

In this example, add the software package `junos-srxsme-10.0R2-domestic.tgz` (for SRX Series devices) with the following options:

- **no-copy** option to install the software package but do not save the copies of package files. You must include this option if you do not have enough space on the internal media to perform an upgrade that keeps a copy of the package on the device.
- **no-validate** option to bypass the compatibility check with the current configuration before installation starts.
- **reboot** option to reboots the device after installation is completed.

Configuration

CLI Quick Configuration

To quickly configure this section of the example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

From operational mode, enter:

```
user@host> request system software add /var/tmp/junos-srxsme-10.0R2-domestic.tgz
no-copy no-validate reboot
```

GUI Step-by-Step Procedure

To install Junos OS upgrades on SRX Series devices:

1. In the J-Web user interface, select **Maintain>Software>Upload Package**.
2. On the Upload Package page, specify the software package to upload. Click **Browse** to navigate to the software package location and select `junos-srxsme-10.0R2-domestic.tgz`.
3. Select the **Reboot If Required** check box to set the device to reboot automatically when the upgrade is complete.
4. Select the **Do not save backup** check box to bypass saving the backup copy of the current Junos OS package (SRX Series).
5. Click **Upload Package**. The software is activated after the device has rebooted.
6. Click **OK** to check your configuration and save it as a candidate configuration.
7. If you are done configuring the device, click **Commit Options>Commit**.

Step-by-Step Procedure The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

From operational mode, install the new package on the device with the `no-copy` and `no-validate` options, and format and re-partition the media before installation, and reboot the device after installation is completed.

To install Junos OS upgrades on SRX Series devices:

1. From operational mode, install the new package on the device

```
user@host> request system software add /var/tmp/junos-srxsme-10.0R2-domestic.tgz  
no-copy no-validate
```

2. Reboot the device.

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

When the reboot is complete, the device displays the login prompt.

Results From configuration mode, confirm your configuration by entering the `show system` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

If you are done configuring the device, enter `commit` from configuration mode.

Verification

Confirm that the configuration is working properly.

- [Verifying the Junos OS Upgrade Installation on page 180](#)

[Verifying the Junos OS Upgrade Installation](#)

Purpose Verify that the Junos OS upgrade was installed.

Action From operational mode, enter the `show system` command.

Related Documentation

- [Understanding Junos OS Upgrades for SRX Series Devices on page 160](#)
- [Preparing the USB Flash Drive to Upgrade Junos OS on SRX Series Devices on page 75](#)
- [Preparing Your SRX Series Device for Junos OS Upgrades on page 162](#)
- [Downloading Software Packages from Juniper Networks on page 177](#)
- [Configuring Administration User Accounts on page 213](#)

Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server

You can use the J-Web user interface to install Junos OS packages that are retrieved with FTP or HTTP from the specified location.



NOTE: This procedure applies only to upgrading from one Junos OS release to another.

Before installing the Junos OS upgrade:

- Verify the available space on the internal media. See “[Preparing Your SRX Series Device for Junos OS Upgrades](#)” on page 162 and the *Junos OS Release Notes*
- Download the software package. See “[Downloading Software Packages from Juniper Networks](#)” on page 177.

To install Junos OS upgrades from a remote server:

1. In the J-Web user interface, select **Maintain>Software>Install Package**.
2. On the Install Remote page, enter the required information in the fields described in [Table 25 on page 181](#).

Table 25: Install Package Summary

Field	Function	Your Action
Package Location (required)	Specifies the FTP or HTTP server, file path, and Junos OS package name.	Type the full address of the Junos OS package location on the FTP or HTTP server—one of the following: <i>ftp://hostname/pathname/package-name</i> <i>http://hostname/pathname/package-name</i>
User	Specifies the username, if the server requires one.	Type the username.
Password	Specifies the password, if the server requires one.	Type the password.
Reboot If Required	Specifies that the device is automatically rebooted when the upgrade is complete.	Check the box if you want the device to reboot automatically when the upgrade is complete.
Do not save backup (SRX Series devices)	Specifies that the backup copy of the current Junos OS package is not saved.	Check the box if you want to save the backup copy of the Junos OS package.

Table 25: Install Package Summary (*continued*)

Field	Function	Your Action
Format and re-partition the media before installation (SRX Series devices)	Specifies that the storage media is formatted and new partitions are created.	Check the box if you want to format the internal media with dual-root partitioning.

- Click **Fetch and Install Package**. Junos OS is activated after the device reboots.

Related Documentation

- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
- [Configuring Administration User Accounts on page 213](#)

Understanding BIOS Upgrades on SRX Series Devices

Understanding Manual BIOS Upgrade Using the Junos CLI

For these SRX Series devices, the BIOS consists of a U-boot and the Junos loader. The SRX240, SRX300, and SRX320, and SRX650 Service Gateways also include a U-shell binary as part of the BIOS. Additionally, on SRX100, SRX110, SRX210, SRX220 and SRX240, SRX300, SRX320, SRX340, and SRX345 Service Gateways, a backup BIOS is supported which includes a backup copy of the U-boot in addition to the active copy from which the system generally boots up.

Table 26 on page 182 Lists the CLI commands used for manual BIOS upgrade.

Table 26: CLI Commands for Manual BIOS Upgrade

Active BIOS	Backup BIOS
<code>request system firmware upgrade re bios</code>	<code>request system firmware upgrade re bios backup</code>

BIOS upgrade procedure:

- Install the **jloader-srxsme** package.

- Copy the jloader-srxsme signed package to the device.



NOTE: The version of the jloader-srxsme package you install must match the version of Junos OS.

- Install the package using the **request system software add <path to jloader-srxsme package> no-copy no-validate** command.



NOTE: Installing the `jloader-srxsme` package places the necessary images under `directory/boot`.

2. Verify that the required images for upgrade are installed. Use the **show system firmware** to verify that the correct BIOS image version is available for upgrade.
3. Upgrade the BIOS (Active and backup) image.

Active BIOS:

1. Initiate the upgrade using the **request system firmware upgrade re bios** command.
2. Monitor the upgrade status using the **show system firmware** command.



NOTE: The device must be rebooted for the upgraded active BIOS to take effect.

Backup BIOS:

1. Initiate the upgrade using the **request system firmware upgrade re bios backup** command.
2. Monitor the upgrade status using the **show system firmware** command.

Understanding Auto BIOS Upgrade Methods on SRX Series Devices

The BIOS version listed in the `bios-autoupgrade.conf` file is the minimum supported version. If the current device has a BIOS version earlier than the minimum compatible version, then the auto BIOS upgrade feature upgrades the BIOS automatically to the latest version.

The BIOS upgrades automatically in the following scenarios:

- During Junos OS upgrade through either the J-Web user interface or the CLI (using the **request system software add no-copy no-validate software-image**). In this case, only the active BIOS is upgraded.
- During loader installation using TFTP or USB (using the **install tftp:///software-image** command). In this case, only the active BIOS is upgraded.
- During system boot-up. In this case, both the active BIOS and the backup BIOS are upgraded.

Related Documentation

- [Understanding Junos OS Upgrades for SRX Series Devices on page 160](#)
- [Installing Junos OS on SRX Series Devices Using a USB Flash Drive on page 77](#)
- [Installing Junos OS on SRX Series Devices from the Boot Loader Using a TFTP Server on page 79](#)

- [Installing Junos OS on SRX Series Devices from the Boot Loader Using a USB Storage Device on page 82](#)
- [Disabling Auto BIOS Upgrade on SRX Series Devices on page 184](#)

Disabling Auto BIOS Upgrade on SRX Series Devices

The auto BIOS upgrade feature is enabled by default. You can disable the feature using the CLI in operational mode.

To disable the automatic upgrade of the BIOS on an SRX Series device, use the **chassis routing-engine bios** command as following:

```
user@host> set chassis routing-engine bios no-auto-upgrade
```



NOTE: The command disables automatic upgrade of the BIOS only during Junos OS upgrade or system boot-up. It does not disable automatic BIOS upgrade during loader installation.

Starting in Junos OS Release 15.1X49-D70 and in Junos OS Release 17.3R1, the **set chassis routing-engine bios uninterrupt** command is introduced on SRX300, SRX320, SRX340, and SRX345 devices to disable user inputs at U-boot and boot loader stage.

To disable the user inputs at u-boot and bootloader stage, use the **chassis routing-engine bios** command as following:

```
user@host> set chassis routing-engine bios uninterrupt
```

Release History Table

Release	Description
15.1X49-D70	Starting in Junos OS Release 15.1X49-D70 and in Junos OS Release 17.3R1, the set chassis routing-engine bios uninterrupt command is introduced on SRX300, SRX320, SRX340, and SRX345 devices to disable user inputs at U-boot and boot loader stage.

Related Documentation

- [Understanding Junos OS Upgrades for SRX Series Devices on page 160](#)
- [Understanding BIOS Upgrades on SRX Series Devices on page 182](#)

Reverting the Junos OS Software Image Back to the Previous Version

This example shows how to downgrade Junos OS on the SRX Series devices.

- [Requirements on page 185](#)
- [Overview on page 185](#)
- [Configuration on page 185](#)
- [Verification on page 186](#)

Requirements

No special configuration beyond device initialization is required before configuring this feature.

Overview

When you upgrade your software, the device creates a backup image of the software that was previously installed in addition to installing the requested software upgrade.

To downgrade the software, you can revert to the previous image using the backup image. You can use this method to downgrade to only the software release that was installed on the device before the current release. To downgrade to an earlier version, follow the procedure for upgrading, using the software image labeled with the appropriate release. This example returns software to the previous Junos OS version.



NOTE: This procedure applies only to downgrading from one Junos OS software release to another or from one Junos OS services release to another.

Configuration

CLI Quick Configuration

To quickly configure this section of the example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

From operational mode, enter:

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

GUI Step-by-Step Procedure

To downgrade Junos OS on SRX Series devices:

1. In the J-Web user interface, select **Maintain>Software>Downgrade**. The image of the previous version (if any) appears on this page.



NOTE: After you perform this operation, you cannot undo it.

2. Select **Downgrade** to downgrade to the previous version of the software or **Cancel** to cancel the downgrade process.
3. Click **Maintain>Reboot** from the J-Web user interface to reboot the device.



NOTE: To downgrade to an earlier version, follow the procedure for upgrading, using the software image labeled with the appropriate release.

4. Click **OK** to check your configuration and save it as a candidate configuration.
5. If you are done configuring the device, click **Commit Options>Commit**.

Step-by-Step Procedure The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To downgrade Junos OS on SRX Series devices:

1. From operational mode, return to the previous Junos OS version.

```
user@host> request system software rollback
```

2. Reboot the device.

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

The device is now running the previous version of Junos OS. To downgrade to an earlier version, follow the procedure for upgrading, using the software image labeled with the appropriate release.

Results From configuration mode, confirm your configuration by entering the **show system** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- [Verifying the Junos OS Downgrade Installation on page 186](#)

Verifying the Junos OS Downgrade Installation

Purpose Verify that the Junos OS downgrade was installed.

Action From operational mode, enter the **show system** command.

- Related Documentation**
- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 201](#)
 - [Understanding Junos OS Upgrades for SRX Series Devices on page 160](#)
 - [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
 - [Restarting and Halting SRX Series Devices on page 253](#)

Overview of CoS Upgrade Requirements (Junos OS Release 11.1 or 11.2 to a Later Release)

Before you upgrade to Junos OS Release 11.3, you must deactivate the CoS configuration if the CoS configuration includes any of the following features:

- **excess-rate** option
- **strict-high** or **high** priority queues
- Any of the Junos OS Release 11.1 or 11.2 default multidestination forwarding classes



CAUTION: If your CoS configuration contains any of the features listed above and you attempt to upgrade from Junos OS Release 11.1 or 11.2 to a later version without first editing the configuration, the Junos OS might not restart.

Junos OS Release 11.3 and later for QFX Series no longer supports the **excess-rate** statement, the **strict** priority option, or the default multidestination forwarding classes used in Junos OS Release 11.1 and 11.2. In addition, Junos OS Release 11.3 introduces new restrictions on how to configure and use **strict-high** priority queues.

This topic does not describe how to perform the software upgrade procedure. It describes how to deactivate your CoS configuration, edit your CoS configuration, and reactivate your CoS configuration at the appropriate times.

Use the following procedure to upgrade safely from Junos OS Release 11.1 or 11.2 to a later release:

1. Deactivate the CoS configuration *before* you upgrade the software:

```
user@switch# deactivate class-of-service
```

2. Follow the upgrade procedure to Junos OS Release 11.3 or later software.
3. Make the following changes to the CoS configuration while the CoS configuration is still deactivated:
 - Remove the **excess-rate** statement from the CoS configuration if you have used it at the **[edit class-of-service schedulers]** or **[edit class-of-service traffic-control-profiles]** hierarchy level.
 - Remove the **strict-high** and **strict** priority queue configurations if you have used them at the **[edit class-of-service schedulers]** hierarchy level.

- Remove the default multidestination forwarding classes (**mcast-be**, **mcast-af**, **mcast-ef**, and **mcast-nc**) if you have used them at the `[edit class-of-service schedulers]`, `[edit class-of-service rewrite-rules]`, `[edit class-of-service classifiers]`, `[edit class-of-service scheduler-maps]`, or `[edit class-of-service forwarding-class-sets]` hierarchy level. Alternatively, you can change the mapping of the multidestination traffic to use the new default multidestination forwarding class (**mcast**).
4. If desired, configure **strict-high** priority queues in accordance with the Junos OS Release 11.3 or later configuration rules, and map multidestination traffic to the default multidestination forwarding class (**mcast**).

5. Activate the CoS configuration:

```
user@switch# activate class-of-service
```

6. Commit the CoS configuration:

```
user@switch# commit
```



NOTE: If you configured the **transmit-rate** option for any queues under the `[edit class-of-service schedulers]` hierarchy level, if the rate is configured as an exact rate in Mbps, we recommend that you reconfigure the **transmit-rate** option as a percentage. This is because the scheduler converts exact rates to percentages, and when the exact rate is below 1 Gbps, some granularity may be lost in the conversion. You can avoid this potential issue by specifying the **transmit-rate** option as a percentage.

Related Documentation

- *Upgrading Software*
- *Understanding CoS Classifiers*
- *Understanding CoS Output Queue Schedulers*
- *Understanding CoS Traffic Control Profiles*
- *Overview of CoS Upgrade Requirements to Junos OS Release 12.2*
- *Overview of CoS Upgrade Requirements to Junos OS Release 12.3 (QFX3500 and QFX3600 Switches) or to Junos OS Release 13.1 (QFabric Systems)*
- *Example: Configuring Unicast Classifiers*
- *Example: Configuring Queue Schedulers*
- *Example: Configuring Traffic Control Profiles (Priority Group Scheduling)*

Before You Begin Installing or Upgrading the Firmware

Before you begin installing or upgrading the firmware on the MIC or PIC, complete the following steps:

1. Verify that a previous version of the firmware package is installed on the router by using the **show version** command.

```
user@host> show version
Hostname: mxHost
Model: mx480
Junos: 15.1I20160816_2117_yyin
JUNOS OS Kernel 64-bit (WITNESS) [20160723.102341_fbsd-builder_stable_10]
JUNOS OS libs [20160723.102341_fbsd-builder_stable_10]
JUNOS OS runtime [20160723.102341_fbsd-builder_stable_10]
JUNOS OS time zone information [20160723.102341_fbsd-builder_stable_10]
...
JUNOS jfirmware [20160628.005233_builder_release_151_f_throttle]
JUNOS Online Documentation [20160812.205759_yyin_release_151_f_throttle]
JUNOS FIPS mode utilities [20160816.211724_yyin_release_151_f_throttle]
....
```

```
user@host> show version
Hostname: ptxHost
Model: ptx3000
Junos: 15.1F-20160720.0
JUNOS Base OS boot [15.1F-20160720.0]
JUNOS Base OS Software Suite [15.1F-20160720.0]
JUNOS platform Software Suite [15.1F-20160720.0]
JUNOS Web Management [15.1F-20160720.0]
JUNOS Runtime Software Suite [15.1F-20160720.0]
JUNOS Online Documentation [15.1F-20160720.0]
...
JUNOS jfirmware [20160628.005233_builder_release_151_f_throttle]
JUNOS 64-bit Runtime Software Suite [15.1F-20160720.0]
JUNOS Packet Forwarding Engine Simulation Package [15.1F-20160720.0]
JUNOS Packet Forwarding Engine Support (M/T/EX Common) [15.1F-20160720.0]
JUNOS Packet Forwarding Engine Support (T-Series) [15.1F-20160720.0]
JUNOS Routing Software Suite [15.1F-20160720.0]
```

If the output of the **show version** command displays **JUNOS jfirmware..** among the list of packages that are installed on the router, then a previous version of the firmware package is installed on the router. If the output of the **show version** command does not display **JUNOS jfirmware..** among the list of packages that are installed on the router, the firmware package is not installed on the router.

2. If a previous version of the firmware package is installed on the router, delete the firmware package from the router by using the **request system firmware delete** command. If a previous version of the firmware package is not installed on the router, then proceed to install the firmware package. For information about how to install the firmware package, see [“Installing Firmware on the 100-Gigabit DWDM OTN MIC \(MIC3-100G-DWDM\)” on page 193](#) or [“Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\)” on page 190](#).

```
user@host> request system software delete jfirmware
/packages/db/jfirmware-x86-32-15.1F-20160625.0
```

3. To verify that the firmware package is removed from the router, use the **show version** command.

```
user@host> show version
Hostname: mxHost
Model: mx240
Junos: 15.1F6-S1.3
JUNOS OS Kernel 64-bit [20160724.331042_builder_stable_10]
JUNOS OS libs [20160724.331042_builder_stable_10]
JUNOS OS runtime [20160724.331042_builder_stable_10]
JUNOS OS time zone information [20160724.331042_builder_stable_10]
....
JUNOS IDP Services [20160812.205945_builder_junos_151_f6_s1]
....
JUNOS Packet Forwarding Engine Support (M/T Common)
[20160812.205945_builder_junos_151_f6_s1]
JUNOS Online Documentation [20160812.205945_builder_junos_151_f6_s1]
JUNOS FIPS mode utilities [20160812.205945_builder_junos_151_f6_s1]
```

If the firmware package is uninstalled successfully, the output of the **show version** command does not display **JUNOS jfirmware..** among the list of packages that are installed on the router.

Related Documentation

- *Configuring OTN Interfaces on MIC3-100G-DWDM MIC*
- *Configuring OTN Interfaces on PTX-5-100G-WDM PIC*
- [Installing Firmware on the 100-Gigabit DWDM OTN MIC \(MIC3-100G-DWDM\) on page 193](#)
- [Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\) on page 190](#)
- *Understanding the MIC3-100G-DWDM MIC*
- *Understanding the PTX-5-100G-WDM PIC*
- [Upgrading Firmware on the 100-Gigabit DWDM OTN MIC \(MIC3-100G-DWDM\) on page 194](#)
- [Upgrading Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\) on page 191](#)

Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC (PTX-5-100G-WDM)

Before you install the firmware package, ensure that a previous version is not installed on the router. For more information, see [“Before You Begin Installing or Upgrading the Firmware” on page 189](#).

To install the firmware package, complete the following steps:

1. Upgrade Junos OS on the router to the version that supports the firmware package. See [“Installing the Software Package on a Router with Redundant Routing Engines” on page 60](#) or [“Installing the Software Package on a Router with a Single Routing Engine” on page 59](#) for more information.
2. Download the firmware package from <http://www.juniper.net/support/>. For information about downloading software packages, see [“Downloading Software” on page 48](#).



NOTE: Download the firmware package specific to your router. The firmware package for PTX Series routers is different from the firmware package for the MX Series routers.

3. Save the firmware package to the `/var/path/package-name` directory on the router. For example, you can save the firmware package to the `/var/tmp` directory.
4. Install the firmware package by using the `request system firmware add path/package-name` command. For example, to install the `jfirmware-15.1F6.9.tgz` package:

```
user@host> request system firmware add jfirmware-15.1F6.9.tgz
```

5. Run the `show version` command to verify that the firmware package is installed.

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

After the firmware package is installed successfully, the output of the `show version` command displays **Junos jfirmware..** among the list of packages that are installed on the router.

Related Documentation

- [Before You Begin Installing or Upgrading the Firmware on page 189](#)
- [Configuring OTN Interfaces on PTX-5-100G-WDM PIC](#)
- [show system firmware](#)
- [Understanding the PTX-5-100G-WDM PIC](#)
- [Upgrading Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\) on page 191](#)

Upgrading Firmware on the 5-Port 100-Gigabit DWDM OTN PIC (PTX-5-100G-WDM)

Before you upgrade the firmware package, ensure that a previous version is not installed on the router. For more information, see [“Before You Begin Installing or Upgrading the Firmware” on page 189](#).

To upgrade the version of your firmware, complete the following steps:

1. Run the **show system firmware** command to view the list of components installed on the router and the firmware version for each component.

```
user@host> show system firmware
```

Part	Type	Tag	Current	Available	Status
	version	version			
FPC 0	ROM Monitor	0	0	10.4.1	OK
FPC 1	ROM Monitor	0	0	10.4.1	OK
FPC 2	ROM Monitor	0	0	10.4.1	OK
PIC 0	CMIC LTC 2/0	1	.0	1.0	OK
FPC 3	ROM Monitor	0	0	10.4.1	OK
FPC 4	ROM Monitor	0	0	13.3.1	OK
FPC 4	MPCS(0)	2	0.24.0		OK
Routing Engine 0	RE BIOS		0	1.18	OK
Routing Engine 1			0	1.18	OK

The output of the **show system firmware** command displays the current firmware version of the PIC as **.0** and the available firmware version as **1.0**.

2. To upgrade the firmware of the PIC, use the **request system firmware upgrade pic** command. For example, to upgrade the firmware version of the PIC from **.0** to **1.0**, specify the FPC slot and PIC slot in the command.

```
user@host> request system firmware upgrade pic pic-slot 0 fpc-slot 2
```

Part	Type	Tag	Current	Available	Status
	version	version			
FPC 2					
PIC 0	CMIC LTC 2/0	1	.0	1.0	OK

Perform indicated firmware upgrade ? [yes,no] (no) yes

Confirm that you want to perform the firmware upgrade by typing **Yes** so the firmware upgrade is initiated.

3. To monitor the progress of the upgrade, use the **show system firmware** command. During the installation process, the status of the PIC changes to **PROGRAMMING**. When the installation process is complete, the status of the PIC changes to **UPGRADED SUCCESSFULLY**.



NOTE: The amount of time it takes to upgrade firmware varies depending on the component.

```
user@host> request system firmware
```

Part	Type	Tag	Current	Available	Status
	version	version			
FPC 0	ROM Monitor	0	0	10.4.1	OK
FPC 1	ROM Monitor	0	0	10.4.1	OK
FPC 2	ROM Monitor	0	0	10.4.1	OK
PIC 0	CMIC LTC 2/0	1	1.0	1.0	UPGRADED SUCCESSFULLY
FPC 3	ROM Monitor	0	0	10.4.1	OK
FPC 4	ROM Monitor	0	0	13.3.1	OK
FPC 4	MPCS(0)	2	0.24.0		OK

Routing Engine 0 RE BIOS	0	1.18	OK
Routing Engine 1	0	1.18	OK



NOTE: If the installation process fails, delete the firmware package by using the `request system software delete firmware-package-name` command. Reinstall the firmware package by following the procedure for installing the firmware package and then upgrade the firmware package.

- Restart the FPC that the PIC is installed in by using the `request chassis fpc fpc-slot restart` command.
- (Optional) After the firmware upgrade is successfully completed, uninstall the firmware package from the router by using the `request system software delete` command.

Related Documentation

- [Before You Begin Installing or Upgrading the Firmware on page 189](#)
- [Configuring OTN Interfaces on PTX-5-100G-WDM PIC](#)
- [Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\) on page 190](#)
- `show system firmware`
- [Understanding the PTX-5-100G-WDM PIC](#)

Installing Firmware on the 100-Gigabit DWDM OTN MIC (MIC3-100G-DWDM)

Before you install the firmware package, ensure that a previous version is not installed on the router. For more information, see [“Before You Begin Installing or Upgrading the Firmware” on page 189](#).

To install the firmware package, complete the following steps:

- Upgrade Junos OS on the router to the version that supports the firmware package. See [“Installing the Software Package on a Router with Redundant Routing Engines” on page 60](#) or [“Installing the Software Package on a Router with a Single Routing Engine” on page 59](#) for more information.
- Download the firmware package from <http://www.juniper.net/support/>. For information about downloading software packages, see [“Downloading Software” on page 48](#).



NOTE: Download the firmware package specific to your router. The firmware package for MX Series routers is different from the firmware package for the PTX Series routers.

3. Save the firmware package to the `/var/path/package-name` directory on the router. For example, you can save the firmware package to the `/var/tmp` directory.
4. Install the firmware package by using the **request system firmware add** `/var/path/package-name` command. For example, to install the `jfirmware-x86-32-15.1F6.9.tgz` package:

```
user@host> request system firmware add jfirmware-x86-32-15.1F6.9.tgz
```
5. Run the **show version** command to verify that the firmware package is installed.

```
user@host> show version
Hostname: Host1
Model: mx480
Junos: 15.1I20160816_2117_yyin
JUNOS OS Kernel 64-bit (WITNESS) [20160723.102341_fbsd-builder_stable_10]
JUNOS OS libs [20160723.102341_fbsd-builder_stable_10]
JUNOS OS runtime [20160723.102341_fbsd-builder_stable_10]
JUNOS OS time zone information [20160723.102341_fbsd-builder_stable_10]
...
JUNOS jfirmware [20160628.005233_builder_release_151_f_throttle]
JUNOS Online Documentation [20160812.205759_yyin_release_151_f_throttle]
JUNOS FIPS mode utilities [20160816.211724_yyin_release_151_f_throttle]
....
```

After the firmware package is installed successfully, the output of the **show version** command displays **JUNOS jfirmware..** among the list of packages that are installed on the router.

Related Documentation

- [Before You Begin Installing or Upgrading the Firmware on page 189](#)
- [Configuring OTN Interfaces on MIC3-100G-DWDM MIC](#)
- [show system firmware](#)
- [Understanding the MIC3-100G-DWDM MIC](#)
- [Upgrading Firmware on the 100-Gigabit DWDM OTN MIC \(MIC3-100G-DWDM\) on page 194](#)

Upgrading Firmware on the 100-Gigabit DWDM OTN MIC (MIC3-100G-DWDM)

Before you upgrade the firmware package, ensure that a previous version is not installed on the router. For more information, see [“Before You Begin Installing or Upgrading the Firmware” on page 189](#).

To upgrade the version of your firmware package, complete the following steps:

1. Run the **show system firmware** command to view the list of components installed on the router and the firmware version for each component.

```
user@host> show system firmware
```

Part	Type	Tag	Current	Available	Status
	version	version			
FPC 0	ROM Monitor	0	0	10.4.1	OK
FPC 1	ROM Monitor	0	0	10.4.1	OK
FPC 2	ROM Monitor	0	0	10.4.1	OK
PIC 0	CMIC LTC 2/0	1	.0	1.0	OK
FPC 3	ROM Monitor	0	0	10.4.1	OK
FPC 4	ROM Monitor	0	0	13.3.1	OK
FPC 4	MPCS(0)	2	0.24.0		OK
Routing Engine 0	RE BIOS	0	1.18		OK
Routing Engine 1		0	1.18		OK

The output of the **show system firmware** command displays the current firmware version of the MIC as **.0** and the available firmware version as **1.0**.

2. To upgrade the firmware of the MIC, use the **request system firmware upgrade pic** command. For example, to upgrade the firmware version of the MIC from **.0** to **1.0**, specify the MPC slot and MIC slot in the command.

```
user@host> request system firmware upgrade pic pic-slot 0 fpc-slot 2
```

Part	Type	Tag	Current	Available	Status
	version	version			
FPC 2					
PIC 0	CMIC LTC 2/0	1	.0	1.0	OK

Perform indicated firmware upgrade ? [yes,no] (no) yes

Confirm that you want to perform the firmware upgrade by typing **Yes** so the firmware upgrade is initiated.

3. To monitor the progress of the upgrade, use the **show system firmware** command. During the installation process, the status of the MIC changes to **PROGRAMMING**. When the installation process is complete, the status of the MIC changes to **UPGRADED SUCCESSFULLY**.



NOTE: The amount of time it takes to upgrade firmware varies depending on the component.

```
user@host> request system firmware
```

Part	Type	Tag	Current	Available	Status
	version	version			
FPC 0	ROM Monitor	0	0	10.4.1	OK
FPC 1	ROM Monitor	0	0	10.4.1	OK
FPC 2	ROM Monitor	0	0	10.4.1	OK
PIC 0	CMIC LTC 2/0	1	.0	1.0	OK
FPC 3	ROM Monitor	0	0	10.4.1	OK
FPC 4	ROM Monitor	0	0	13.3.1	OK
FPC 4	MPCS(0)	2	0.24.0		OK

Routing Engine 0 RE BIOS	0	1.18	OK
Routing Engine 1	0	1.18	OK



NOTE: If the installation process fails, delete the firmware package by using the `request system software delete firmware-package-name` command. Reinstall the firmware package by following the procedure for installing the firmware package and then upgrade the firmware package.

- Restart the MPC that the MIC is installed in by using the `request chassis fpc fpc-slot restart` command.
- (Optional) After the firmware upgrade is successfully completed, uninstall the firmware package from the router by using the `request system software delete` command.

**Related
Documentation**

- [Before You Begin Installing or Upgrading the Firmware on page 189](#)
- [Installing Firmware on the 100-Gigabit DWDM OTN MIC \(MIC3-100G-DWDM\) on page 193](#)
- `show system firmware`
- [Understanding the MIC3-100G-DWDM MIC](#)

CHAPTER 9

Booting a Device Using a System Snapshot

- [Understanding System Snapshot on EX Series Switches on page 197](#)
- [Creating a Snapshot and Using It to Boot an EX Series Switch on page 198](#)
- [Verifying That a System Snapshot Was Created on an EX Series Switch on page 199](#)
- [Booting an EX Series Switch Using a Software Package Stored on a USB Flash Drive on page 200](#)
- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 201](#)

Understanding System Snapshot on EX Series Switches

You can create copies of the software running a Juniper Networks EX Series Ethernet Switch using the system snapshot feature. The system snapshot feature takes a “snapshot” of the files currently used to run the switch and copies them to an alternate storage location. You can then use this snapshot to boot the switch at the next bootup or as a backup boot option.

The switch can boot from either internal flash media or external (USB) flash media. The contents of the snapshot vary depending on whether you create the snapshot on the media that the switch booted from or on the media that it did not boot from.

Snapshots are particularly useful for moving files onto USB flash drives. You cannot use the **copy** command or any other file-moving technique to move files from an internal memory source to USB memory on the switch.

- If you create the snapshot on the media that the switch did not boot from, the following partitions on the boot media are included in the snapshot: **root**, **altroot**, **var**, **var/tmp**, and **config**.

The **root** partition is the primary boot partition, and the **altroot** partition is the backup boot partition.

- If you create the snapshot on the media that the switch booted from, the root partition that the switch booted from is copied to the alternate root partition. The **var**, **var/tmp**, and **config** partitions are not copied as part of the snapshot because they already exist on the boot media.

The system snapshot feature has the following limitations:

- You cannot use snapshots to move files to any destination outside the switch other than an installed external USB flash drive or switches that are members of the same Virtual Chassis as the switch on which you created the snapshot.
- Snapshot commands, like all commands executed on a Virtual Chassis, are executed on the local member switch. If different member switches request the snapshot, the snapshot command is pushed to the Virtual Chassis member creating the snapshot and is executed on that member, and the output is then returned to the switch that initiated the process. For instance, if the command to create an external snapshot on member 3 is entered on member 1, the snapshot of internal memory on member 3 is taken on external memory on member 3. The output of the process is seen on member 1. No files move between the switches.

**Related
Documentation**

- [Understanding Software Installation on EX Series Switches on page 44](#)
- [Creating a Snapshot and Using It to Boot an EX Series Switch on page 198](#)

Creating a Snapshot and Using It to Boot an EX Series Switch

The system snapshot feature takes a “snapshot” of the files currently used to run the switch and copies them to an alternate storage location. You can then use this snapshot to boot the switch at the next bootup or as a backup boot option.

This topic includes the following tasks:

- [Creating a Snapshot on a USB Flash Drive and Using It to Boot the Switch on page 198](#)

Creating a Snapshot on a USB Flash Drive and Using It to Boot the Switch

You can create a snapshot on USB flash memory after a switch is booted by using files stored in internal memory.

Ensure that you have the following tools and parts available before creating a snapshot on a USB flash drive:

- A USB flash drive that meets the switch USB port specifications. See *USB Port Specifications for an EX Series Switch*.

To create a snapshot on USB flash memory and use it to boot the switch:

1. Place the snapshot into USB flash memory:

```
user@switch> request system snapshot partition media usb
```

2. (Optional) Perform this step if you want to boot the switch now using the snapshot stored on the USB flash drive.

```
user@switch> request system reboot (EX Series) media usb
```

- Related Documentation**
- [Verifying That a System Snapshot Was Created on an EX Series Switch on page 199](#)
 - [Understanding System Snapshot on EX Series Switches on page 197](#)

Verifying That a System Snapshot Was Created on an EX Series Switch

Purpose Verify that a system snapshot was created with the proper files on an EX Series switch.

Action View the snapshot:

```
user@switch> show system snapshot media external
Information for snapshot on      external (/dev/dals1a) (backup)
Creation date: Mar 19 03:37:18 2012
JUNOS version on snapshot:
  jbase : ex-12.1I20120111_0048_user
  jcrypto-ex: 12.1I20120111_0048_user
  jdocs-ex: 12.1I20120111_0048_user
  jroute-ex: 12.1I20120111_0048_user
  jswitch-ex: 12.1I20120111_0048_user
  jweb-ex: 12.1I20120111_0048_user
Information for snapshot on      external (/dev/dals2a) (primary)
Creation date: Mar 19 03:38:25 2012
JUNOS version on snapshot:
  jbase : ex-12.2I20120305_2240_user
  jcrypto-ex: 12.2I20120305_2240_user
  jdocs-ex: 12.2I20120305_2240_user
  jroute-ex: 12.2I20120305_2240_user
  jswitch-ex: 12.2I20120305_2240_user
  jweb-ex: 12.2I20120305_2240_user
```

Meaning The output shows the date and time when the snapshot was created and the packages that are part of the snapshot. Check to see that the date and time match the time when you created the snapshot.

You can compare the output of this command to the output of the **show system software** command to ensure that the snapshot contains the same packages as the software currently running the switch.

- Related Documentation**
- [Creating a Snapshot and Using It to Boot an EX Series Switch on page 198](#)

Booting an EX Series Switch Using a Software Package Stored on a USB Flash Drive

There are two methods of getting Junos OS stored on a USB flash drive before using the software to boot the switch. You can pre-install the software onto the USB flash drive before inserting the USB flash drive into the USB port, or you can use the system snapshot feature to copy files from internal switch memory to the USB flash drive.

To move files into USB flash memory by using a system snapshot and use those files to boot the switch, see [“Creating a Snapshot and Using It to Boot an EX Series Switch” on page 198](#). We recommend that you use this method to boot the switch from a USB flash drive if your switch is running properly.

If you need to pre-install the software onto the USB flash drive, you can use the method described in this topic. Pre-installing Junos OS onto a USB flash drive to boot the switch can be done at any time and is particularly useful when the switch boots to the loader prompt because the switch cannot locate the Junos OS in internal flash memory.

Ensure that you have the following tools and parts available to boot the switch from a USB flash drive:

- A USB flash drive that meets the EX Series switch USB port specifications. See *USB Port Specifications for an EX Series Switch*.
- A computer or other device that you can use to download the software package from the Internet and copy it to the USB flash drive.

To download a Junos OS package onto a USB flash drive before inserting the USB flash drive:

1. Download the Junos OS package that you want to place onto the EX Series switch from the Internet onto the USB flash drive by using your computer or other device. See [“Downloading Software Packages from Juniper Networks” on page 51](#).
2. Remove the USB flash drive from the computer or other device.
3. Insert the USB flash drive into the USB port on the switch.
4. This step can be performed only when the prompt for the loader script (**loader>**) is displayed. The loader script starts when the Junos OS loads but the CLI is not working for any reason or if the switch has no software installed.

Install the software package onto the switch:

```
loader> install source
```

where **source** represents the name and location of the Junos OS package on the USB flash drive. The Junos OS package on a flash drive is commonly stored in the root drive as the only file—for example, **file:///jinstall-ex-4200-9.4R1.5-domestic-signed.tgz**.

- Related Documentation**
- [Installing Software on an EX Series Switch with a Single Routing Engine \(CLI Procedure\)](#)
 - [Installing Software on EX Series Switches \(J-Web Procedure\) on page 66](#)
 - [Understanding Software Installation on EX Series Switches on page 44](#)
 - See *EX2200 Switches Hardware Overview* for USB port location.
 - See *Rear Panel of an EX3200 Switch* for USB port location.
 - See *Rear Panel of an EX3300 Switch* for USB port location.
 - See *Rear Panel of an EX4200 Switch* for USB port location.
 - See *EX4300 Switches Hardware Overview* for USB port location.
 - See *Front Panel of an EX4500 Switch* for USB port location.
 - See *EX4550 Switches Hardware Overview* for USB port location.
 - See *Switch Fabric and Routing Engine (SRE) Module in an EX6200 Switch* for USB port location.
 - See *Switch Fabric and Routing Engine (SRE) Module in an EX8208 Switch* for USB port location.
 - See *Routing Engine (RE) Module in an EX8216 Switch* for USB port location.

Example: Creating a Snapshot and Using It to Boot an SRX Series Device

This example shows how to configure a boot device.

- [Requirements on page 201](#)
- [Overview on page 201](#)
- [Configuration on page 202](#)
- [Creating a Snapshot on a USB Flash Drive and Using It to Boot the SRX Series Device on page 203](#)
- [Verification on page 204](#)

Requirements

Before you begin, ensure that the backup device has a storage capacity of at least 1 GB. See [“Preparing Your SRX Series Device for Junos OS Upgrades” on page 162](#).

Overview

You can configure a boot device to replace the primary boot device on your SRX Series device or to act as a backup boot device. Use either the J-Web user interface or the CLI to take a snapshot of the configuration currently running on the device, or of the original factory configuration and a rescue configuration, and save it to an alternate medium.



NOTE: For media redundancy, we recommend that you keep a secondary storage medium attached to the SRX Series device and updated at all times.

If the primary storage medium becomes corrupted and no backup medium is in place, you can recover the primary internal media from the TFTP installation.

You can also configure a boot device to store snapshots of software failures for use in troubleshooting.



NOTE: You cannot copy software to the active boot device.



NOTE: After a boot device is created with the default factory configuration, it can operate only in an internal media slot.

This example configures a boot device to back up the currently running and active file system partitions by rebooting from internal media and including only files shipped from the factory.

Configuration

CLI Quick Configuration

To quickly configure this section of the example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

From operational mode, enter:

```
user@host> request system snapshot partition media internal factory
```

GUI Step-by-Step Procedure

To configure a boot device:

1. In the J-Web user interface, select **Maintain>Snapshot**.
2. On the Snapshot page, specify the boot device to copy the snapshot to. From the Target Media list, select the **internal** boot device.
3. Select the Factory check box to copy only default files that were loaded on the internal media when it was shipped from the factory, plus the rescue configuration if one has been set.
4. Select the Partition check box to partition the medium that you are copying the snapshot to. This process is usually necessary for boot devices that do not already have software installed on them.
5. Click **Snapshot**.

6. Click **OK** to check your configuration and save it as a candidate configuration.

7. If you are done configuring the device, click **Commit Options>Commit**.

Step-by-Step Procedure The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure a boot device:

```
user@host> request system snapshot partition media internal factory
```

Results From configuration mode, confirm your configuration by entering the **show system snapshot media internal** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host> show system snapshot media internal
```

```
Information for snapshot on      internal (/dev/ad0s1a) (backup)
Creation date: Oct 9 13:30:06 2009
JUNOS version on snapshot:
  junos : 10.0B3.10-domestic
Information for snapshot on      internal (/dev/ad0s2a) (primary)
Creation date: Jan 6 15:45:35 2010
JUNOS version on snapshot:
  junos : 10.2-20091229.2-domestic
```

If you are done configuring the device, enter **commit** from configuration mode.

Creating a Snapshot on a USB Flash Drive and Using It to Boot the SRX Series Device

Step-by-Step Procedure You can create a snapshot on a USB flash drive and use it to boot the SRX series device.

To create a snapshot on a USB flash memory and use it to boot the SRX series device:

1. Place the snapshot into USB flash memory:

```
user@host> request system snapshot partition media USB
```

2. (Optional) Perform this step, if you want to boot the SRX now using the newly created snapshot on the USB flash drive. If you created the snapshot as a backup, do not perform this step.

- To reboot the SRX using the most recently created snapshot:

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

Verification

Confirm that the configuration is working properly.

- [Verifying the Snapshot Information on page 204](#)

Verifying the Snapshot Information

Purpose Verify that the snapshot information for both root partitions on SRX Series devices were configured.

Action From operational mode, enter the **show system snapshot media** command.

The command output displays the snapshot creation time and Junos OS Release version on a media for both the primary and backup roots.



.....
NOTE: With the dual-root partitioning scheme, performing a snapshot to a USB storage device that is less than 1 GB is not supported.
.....



.....
NOTE: You can use the **show system snapshot media internal** command to determine the partitioning scheme present on the internal media. Information for only one root is displayed for single-root partitioning, whereas information for both roots is displayed for dual-root partitioning.
.....



.....
NOTE: Any removable media that has been formatted with dual-root partitioning will not be recognized correctly by the **show system snapshot CLI** command on systems that have single-root partitioning. Intermixing dual-root and single-root formatted media on the same system is strongly discouraged.
.....

CHAPTER 10

Performing a Recovery Installation

- [Creating an Emergency Boot Device on page 205](#)
- [Understanding Integrity Check and Autorecovery of Configuration, Licenses, and Disk Information on SRX Series Devices on page 207](#)
- [Creating an Emergency Boot Device for RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX Routing Engines on page 209](#)
- [Performing a Recovery Installation on page 211](#)
- [Creating a New Configuration on a Single Routing Engine on page 212](#)
- [Creating a New Configuration with Redundant Routing Engines on page 217](#)
- [Saving a Rescue Configuration File on page 222](#)
- [Restoring a Saved Configuration on page 223](#)
- [Reverting to the Default Factory Configuration by Using the request system zeroize Command on page 225](#)
- [Reverting to the Rescue Configuration on page 225](#)

Creating an Emergency Boot Device

If the device's Junos OS software is damaged in some way that prevents Junos OS software from loading completely, you can use the emergency boot device to revive the device. The emergency boot device repartitions the primary disk and reloads a fresh installation of Junos OS software.

The procedures outlined in this section discuss how to create an emergency boot device for any ACX Series, M Series, MX Series, T Series, TX Matrix, and TX Matrix Plus router.

To create an emergency boot device:

1. Use FTP to copy the installation media into the router's **/var/tmp** directory.
2. Insert the PC Card into the external PC Card slot or USB storage device into the USB port.
3. In the UNIX shell, navigate to the **/var/tmp** directory:

```
start shell
cd /var/tmp
```

4. Log in as **su**:

```
su [enter]
password: [enter SU password]
```

5. For Junos OS with upgraded FreeBSD only, expand the image, for example:

```
gzip -d installMedia.img.gz
```

where **installMedia** refers to the installation media downloaded into the **/var/tmp** directory. For example, for Junos OS with upgraded FreeBSD, the filename might be **junos-install-media-usb-mx-x86-64-16.1R2.11.img.gz**. (To determine which platforms use Junos OS with upgraded FreeBSD, see [“Understanding Junos OS with Upgraded FreeBSD” on page 17.](#))

6. Issue the following commands:

- For Junos OS with upgraded FreeBSD:

```
dd if=/dev/zero of=/dev/externalDrive count=20
dd if=installMedia.img of=/dev/externalDrive bs=256k
```

- For Junos OS:

```
dd if=/dev/zero of=/dev/externalDrive count=20
dd if=installMedia of=/dev/externalDrive bs=64k
```

where:

- externalDrive**—Refers to the removable media name of the emergency boot device. For example, the removable media name for an emergency boot device on the M120 router is **da0** for both Routing Engines. For the names of the removable media, see the table in [“Routing Engines and Storage Media Names \(ACX Series, M Series, MX Series, PTX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers\)” on page 33.](#)
- installMedia**—Refers to the installation media downloaded into the **/var/tmp** directory. For example, the filename might be **install-media-9.0R2.10-domestic** for Junos OS or, for Junos OS with upgraded FreeBSD, **junos-install-media-usb-mx-x86-64-16.1R2.11.img** (unzipped). (To determine which platforms use Junos OS with upgraded FreeBSD, see [“Understanding Junos OS with Upgraded FreeBSD” on page 17.](#))

7. Log out as **su**:

```
exit
```

**Related
Documentation**

- [Routing Engines and Storage Media Names \(ACX Series, M Series, MX Series, PTX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers\) on page 33](#)
- [Understanding Junos OS with Upgraded FreeBSD on page 17](#)

Understanding Integrity Check and Autorecovery of Configuration, Licenses, and Disk Information on SRX Series Devices

This topic includes the following sections:

- [Overview on page 207](#)
- [How Autorecovery Works on page 207](#)
- [How to Use Autorecovery on page 208](#)
- [Data That Is Backed Up in an Autorecovery on page 208](#)
- [Troubleshooting Alarms on page 208](#)
- [Considerations on page 209](#)

Overview

The autorecovery feature is supported on dual-partitioned SRX Series devices. With this feature, information on disk partitioning, configuration, and licenses is recovered automatically in the event it becomes corrupted.

Autorecovery provides the following functions:

- Detect corruption in disk partitioning during system bootup and attempt to recover partitions automatically
- Detect corruption in the Junos OS rescue configuration during system bootup and attempt to recover the rescue configuration automatically
- Detect corruption in Junos OS licenses during system bootup and attempt to recover licenses automatically

How Autorecovery Works

The feature works in the following ways:

- The feature provides the **request system autorecovery state save** command, which backs up important data such as disk partitioning information, licenses, and Junos OS rescue configuration.
- Once the backup copies are saved, they are used to check the integrity of the working copies of the data on every bootup.
- The working copies are automatically recovered if any corruption is detected.

How to Use Autorecovery

You use autorecovery in the following ways:

- Prepare the router for deployment with the necessary licenses and configuration.
- After you finalize the state, execute the **request system autorecovery state save** command to back up the state.
- After you save the state, integrity check and recovery actions (if any) occur automatically on every bootup.
- If subsequent maintenance activities change the state of the router by adding licenses or updating the configuration, you need to execute the **request system autorecovery state save** command again to update the saved state.
- Execute the **show system autorecovery state** command any time to view the status of the saved information and the integrity check status of each saved item.
- Execute the **request system autorecovery state clear** command to delete all backed up data and disable autorecovery, if required.

Data That Is Backed Up in an Autorecovery

The following data is backed up during the autorecovery process:

- Rescue configuration (regenerated from the current configuration)
- License keys
- BSD labels (disk-partitioning information)

Data is backed up only when you execute the **request system autorecovery state save** command. Disk-partitioning information is backed up automatically from factory defaults (for new systems), on installation from the boot loader, and on snapshot creation.

Troubleshooting Alarms

Table 27 on page 208 lists types of autorecovery alarms, descriptions, and required actions.

Table 27: Autorecovery Alarms

Alarm	Alarm Type	Description	Action Required
Autorecovery information needs to be saved	Minor	This alarm indicates: <ul style="list-style-type: none"> • Unsaved data needs to be saved, or saved data contains problems and another save is required. 	<ul style="list-style-type: none"> • Ensure that the system has all required licenses and configuration. • Execute the request system autorecovery state save command.
Autorecovery has recovered corrupted information	Minor	This alarm indicates: <ul style="list-style-type: none"> • Boot time integrity check failed for certain items; however, the items have been recovered successfully. 	<ul style="list-style-type: none"> • No action is required. • Alarm is cleared on next bootup.

Table 27: Autorecovery Alarms (*continued*)

Alarm	Alarm Type	Description	Action Required
Autorecovery was unable to recover data completely	Major	This alarm indicates: <ul style="list-style-type: none"> • Boot time integrity check failed for certain items, which could not be recovered successfully. 	<ul style="list-style-type: none"> • The system might be experiencing a fatal malfunction.

Considerations

- Devices must have dual-root partitioning for autorecovery to work.
- The **request system configuration rescue save** command regenerates the rescue configuration from the current Junos OS configuration and then saves it. Therefore, executing the **save** command overwrites any existing rescue configuration.
- In general, the saved contents of the rescue configuration are not updated automatically. If you add licenses, you must execute the **request system autorecovery state save** command again.



NOTE: The rescue configuration is backed up. If `/config` is corrupted, the system boots from the rescue configuration.

Related Documentation

- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 201](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 178](#)
- [Reverting the Junos OS Software Image Back to the Previous Version on page 184](#)

Creating an Emergency Boot Device for RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX Routing Engines

If Junos OS on your router is damaged during loading in a way that prevents it from loading completely, you can use the emergency boot device to revive the router. The emergency boot device repartitions the primary disk and reloads a fresh installation of Junos OS. For RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX Routing Engines, you can use a USB storage device with at least 8 GB of free space to create an emergency boot device.

To create an emergency boot device on a router with RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX Routing Engines:

1. Copy the installation media into the router's `/var/tmp` directory.
2. Insert the USB storage device into the router's USB port.
3. In the UNIX shell, navigate to the `/var/tmp` directory:

```
start shell
cd /var/tmp
```

4. Log in as **su**:

```
su [enter]
password: [enter SU password]
```

5. Gunzip the copied file.

For example, to convert `junos-vmhost-install-usb-mx-x86-64-15.1F6.8.img.gz` to `junos-vmhost-install-usb-mx-x86-64-15.1F6.8.img`, use the following command:

```
gunzip junos-vmhost-install-usb-mx-x86-64-15.1F6.8.img.gz
```

6. Issue the following command:

```
dd if=/path/to/downloaded.img of=/dev/devicenode bs=4M
```

where:

- **devicenode**—Refers to the name of the removable media of the emergency boot device. For names of storage media, see [“Routing Engines and Storage Media Names \(ACX Series, M Series, MX Series, PTX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers\)”](#) on page 33.
- **downloaded.img**—Refers to the installation media copied to the `/var/tmp` directory. For example, `junos-vmhost-install-usb-ptx-x86-64-15.1F6.8.img`.

The following code example can be used to create an emergency boot device by using a USB storage device:

```
dd if=/path/to/junos-vmhost-install-usb-mx-x86-64-15.1F6.8.img of=/dev/da0
bs=4M
```



NOTE: In the `dd` command, use `junos-vmhost-install-usb-mx-86` for RE-MX-X6 and RE-MX-X8 Routing Engines and `junos-vmhost-install-ptx-86` for RE-PTX-X8 Routing Engine respectively.

7. Log out as **su**:

```
exit
```

Related Documentation

- [Boot Process Overview on page 267](#)

Performing a Recovery Installation

If the device's software is corrupted or otherwise damaged, you may need to perform a recovery installation, using the emergency boot device to restore the default factory installation. Once you have recovered the software, you will need to restore the router or switch's configuration. You can either create a new configuration as you did when the device was shipped from the factory, or if you saved the device's previous configuration, you can simply restore that file to the system.

Depending on the situation, you should try to perform the following steps before you perform the recovery installation:

1. Ensure you have an emergency recovery disk to use during the installation. When the router or switch is first shipped, an emergency recovery disk is provided with it. For instructions on creating an emergency boot device, see [“Creating an Emergency Boot Device” on page 205](#)
2. Copy the existing configuration in the file `/config/juniper.conf.gz` from the device to a remote system. For extra safety, you can also copy the backup configurations (the files named `/config/juniper.conf.n`, where *n* is a number from 0 through 9).



WARNING: The recovery installation process completely overwrites the entire contents of the fixed storage media.

3. Copy any other stored files to a remote system as desired.

To reinstall Junos OS:

1. Insert the removable media emergency boot device into the device.



NOTE: You can store a configuration on installation media such as a PC Card or USB stick.

2. Reboot the device.

If the CLI is still active, issue the **request system reboot** command from command mode to reboot the device.

If the CLI is not working, manually power off the device using the main power switch, wait 10 seconds, and then power the device back on.

3. When the software prompts you with the following question, type **y**:



NOTE: Introduced in Junos OS Release 15.1, Junos OS with upgraded FreeBSD does not display the following warning. To determine which platforms use Junos OS with upgraded FreeBSD, see [“Understanding Junos OS with Upgraded FreeBSD” on page 17](#).

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

The device copies the software from the removable media emergency boot device onto your system, occasionally displaying status messages. Copying the software can take up to 45 minutes, depending on the device. When the process is complete, the router boots into Amnesiac state and the login prompt is displayed.

4. Remove the removable media emergency boot device.
5. Log in as root on the device's console port and issue the **request system reboot** command from command mode to reboot the device.

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

6. Create a new configuration as you did when the device was shipped from the factory, or restore a previously saved configuration file to the system. For more information, see [“Creating a New Configuration on a Single Routing Engine” on page 212](#), [“Creating a New Configuration with Redundant Routing Engines” on page 217](#), and [“Restoring a Saved Configuration” on page 223](#).

Creating a New Configuration on a Single Routing Engine

To create a new base configuration on a single Routing Engine, perform the following tasks:

- [Logging In to the Router Console on page 212](#)
- [Configuring Administration User Accounts on page 213](#)
- [Adding the Management Console to the Network on page 213](#)
- [Committing Changes on page 215](#)

Logging In to the Router Console

To log in to the device's console interface and open the CLI in configuration mode:

1. Verify the device is powered on.
2. Log in through the console port as root.

```
Amnesiac <tttyd0>
```

```
login: root
```



NOTE: From the factory, the root administration user account is not associated with a password. However, you must add a password to the root administration account before you can successfully commit a configuration.

3. Start the CLI, which initially opens in operational mode. Note the command prompt ends with **>** in the CLI operational mode.

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

4. Enter the CLI configuration mode. Note the command prompt ends with **#** in the CLI configuration mode.

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

Configuring Administration User Accounts

Set the root administration user account password. You also need to set up one or more administration user accounts. These administration user accounts are used to log in to the device through the management console. To configure administration user accounts:

1. Add a password to the root (superuser) administration user account.

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

2. Create a management console user account.

```
[edit]
root# set system login user user-name authentication plain-text-password
New Password: password
Retype new password: password
```

3. Set the user account class to **super-user**.

```
[edit]
root# set system login user user-name class super-user
```

Adding the Management Console to the Network

To add the management console to the network:

1. Specify the device hostname.



NOTE: The hostname specified in the device configuration is not used by the DNS server to resolve to the correct IP address. This hostname is used to display the name of the Routing Engine in the CLI. For example, this hostname appears on the command line prompt when the user is logged in to the CLI:

```
user-name@host-name>
```

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

2. Configure the IP address of the DNS server.

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

3. Configure the router or switch domain name.

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

4. Configure the IP address and prefix length for the router or switch Ethernet interface.

- For all devices *except* the TX Matrix Plus router, T1600 or T4000 routers in a routing matrix, and PTX Series Packet Transport Routers:

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

- For the TX Matrix Plus router, T1600 or T4000 routers in a routing matrix only, and PTX Series Packet Transport Routers:

```
[edit]
root@# set interfaces em0 unit 0 family inet address address/prefix-length
```

To use **em0** as an out-of-band management Ethernet interface, you must configure its logical port, **em0.0**, with a valid IP address.

- For a T1600 standalone router (not connected to a TX Matrix Plus router and not in a routing matrix):

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

5. Configure the IP address of a backup router. The backup router is used while the local router is booting and if the routing process fails to start. Once the routing process starts, the backup router address is removed from the local routing and forwarding tables. For more information about the backup router, see the *Getting Started Guide for Routing Devices*.

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

6. (Optional) Configure the static routes to remote subnets with access to the management port. Access to the management port is limited to the local subnet. To access the management port from a remote subnet, you need to add a static route to that subnet within the routing table.

```
[edit]
root# set routing-options static route remote-subnet next-hop destination-IP
retain no-readvertise
```

7. Configure telnet service at the **[edit system services]** hierarchy level.

```
[edit]
root# set system services telnet
```

Committing Changes

Now that you have completed your changes to the configuration file, commit the configuration changes.

1. Before committing the configuration, you can review your changes to the configuration with the **show** command.

```

root# show
## Last changed: 2008-08-27 22:30:42 UTC
version 9.3B1.5;
system {
    host-name tp8;
    domain-name subnet.device1.example.com;
    backup-router 192.168.71.254;
    root-authentication {
        encrypted-password "$ABC123"; ## SECRET-DATA
    }
    name-server {
        192.168.5.68;
        172.17.28.101;
    }
    login {
        user PE1 {
            class super-user;
            authentication {
                encrypted-password "$ABC123"; ## SECRET-DATA
            }
        }
    }
    services {
        telnet;
    }
    syslog {
        user * {
            any emergency;
        }
        file messages {
            any notice;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
}
interfaces {
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.69.205/21;
            }
        }
    }
}
routing-options {
    static {
        route 172.16.0.0/12 {
            next-hop 192.168.71.254;
            retain;
        }
    }
}

```

```

        no-readvertise;
    }
    route 192.168.0.0/16 {
        next-hop 192.168.71.254;
        retain;
        no-readvertise;
    }
}

```

On a TX Matrix Plus router and PTX Series Packet Transport Routers, the management Ethernet interface is **em0** and not **fxp0**. Therefore, when you issue the **show** command in the configuration mode, the configuration statements would be:

```

[edit]
root@ show
system {
    host-name hostname;
    domain-name domain.name;
    backup-router address ;
    root-authentication {
        (encrypted-password "password" | public-key);
        ssh-rsa "public-key";
        ssh-dsa "public-key";
    }
    name-server {
        address;
    }
    interfaces {
        em0 {
            unit 0 {
                family inet {
                    address address ;
                }
            }
        }
    }
}

```

2. Commit the configuration.

```

[edit]
root# commit
commit complete

```



NOTE: If you receive an error message after you issue the **commit** statement, you can review the configuration using the **show** command to find the errors in your configuration. You can delete incorrect entries using the **delete** command. For example, to delete a hostname from the configuration, issue the following statement:

```

[edit]
root# delete system host-name host-name

```

3. Exit configuration mode.

```
[edit]
root# exit
Exiting configuration mode

root>
```

Creating a New Configuration with Redundant Routing Engines

To create a new base configuration on a router with redundant Routing Engines, perform the following tasks:

- [Configuring Administration User Accounts on page 217](#)
- [Setting Up Routing Engine Configuration Groups on page 218](#)
- [Completing the Management Console Configuration on page 220](#)
- [Committing and Synchronizing Changes on page 220](#)

Configuring Administration User Accounts

Set the root administration user account password. You also need to set up one or more administration user accounts. These administration user accounts are used to log in to the device through the management console. To configure administration user accounts:

1. Add a password to the root (superuser) administration user account.

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

2. Create a management console user account.

```
[edit]
root# set system login user user-name authentication plain-text-password
New Password: password
Retype new password: password
```

3. Set the user account class to **super-user**.

```
[edit]
root# set system login user user-name class super-user
```

Setting Up Routing Engine Configuration Groups

In a router with two Routing Engines, one configuration should be shared between both Routing Engines. This ensures that both Routing Engine configurations are identical. Within this configuration, create two Routing Engine groups, one for each Routing Engine. Within these groups, you specify the Routing Engine–specific parameters.

For more information about creating configuration groups, see the *CLI User Guide*.

For more information about the initial configuration for redundant Routing Engine systems and the **re0** group, see *Junos OS High Availability Library for Routing Devices*.

1. Create the configuration group **re0**. The **re0** group is a special group designator that is only used by **RE0** in a redundant routing platform.

```
[edit]
root# set groups re0
```

2. Navigate to the **groups re0** level of the configuration hierarchy.

```
[edit]
root# edit groups re0
```

3. Specify the router hostname.

```
[edit groups re0]
root# set system host-name host-name
```



NOTE: The hostname specified in the router configuration is not used by the DNS server to resolve to the correct IP address. This hostname is used to display the name of the Routing Engine in the CLI. For example, the hostname appears at the command-line prompt when the user is logged in to the CLI:

```
user-name@host-name>
```

4. Configure the IP address and prefix length for the router Ethernet interface.
 - For all devices *except* the TX Matrix Plus router, T1600 or T4000 routers in a routing matrix, and PTX Series Packet Transport Routers:

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

- For the TX Matrix Plus router, T1600 or T4000 routers in a routing matrix only, and PTX Series Packet Transport Routers:

```
[edit]
root@# set interfaces em0 unit 0 family inet address address/prefix-length
```

To use **em0** as an out-of-band management Ethernet interface, you must configure its logical port, **em0.0**, with a valid IP address.

- For a T1600 standalone router (not connected to a TX Matrix Plus router and not in a routing matrix):

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

5. Return to the top level of the hierarchy.

```
[edit groups re0]
root# top
```

6. Create the configuration group **re1**.

```
[edit]
root# set groups re1
```

7. Navigate to the **groups re1** level of the configuration hierarchy.

```
[edit]
root# edit groups re1
```

8. Specify the router hostname.

```
[edit groups re1]
root# set system host-name host-name
```

9. Configure the IP address and prefix length for the router Ethernet interface.

- For all devices *except* the TX Matrix Plus router, T1600 or T4000 routers in a routing matrix, and PTX Series Packet Transport Routers:

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

- For the TX Matrix Plus router and T1600 or T4000 routers in a routing matrix only:

```
[edit]
root@# set interfaces em0 unit 0 family inet address address/prefix-length
```

To use **em0** as an out-of-band management Ethernet interface, you must configure its logical port, **em0.0**, with a valid IP address.

- For a T1600 standalone router (not connected to a TX Matrix Plus router, and not in a routing matrix):

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

10. Return to the top level of the hierarchy.

```
[edit groups re0]
root# top
```

11. Specify the group application order.

```
[edit]
root# set apply-groups [ re0 re1 ]
```

Completing the Management Console Configuration

To configure the global management console parameters:

1. Configure the IP address of the DNS server.

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

2. Configure the router domain name.

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

3. Configure the IP address of a backup router. The backup router is used while the local router is booting and if the routing process fails to start. Once the routing process starts, the backup router address is removed from the local routing and forwarding tables. For more information about the backup router, see the *Getting Started Guide for Routing Devices*.

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

4. (Optional) Configure the static routes to remote subnets with access to the management port. Access to the management port is limited to the local subnet. To access the management port from a remote subnet, you need to add a static route to that subnet within the routing table.

```
[edit]
root# set routing-options static route remote-subnet next-hop destination-IP
retain no-readvertise
```

5. Configure telnet service at the `[edit system services]` hierarchy level.

```
[edit]
root# set system services telnet
```

Committing and Synchronizing Changes

Commit the configuration changes. When you issue the **synchronize** command, the configuration is shared between both Routing Engines and committed on both Routing Engines simultaneously.

1. Before committing the configuration, you can review the configuration entries using the **show** command.

```
root# show
## Last changed: 2008-10-17 18:32:25 UTC
version 9.1R1.8;
groups {
  re0 {
    system {
      host-name spice-re0;
    }
    interfaces {
```



```

        fxp0 {
            unit 0 {
                family inet {
                    address 192.168.69.155/21;
                }
            }
        }
    }
}
re1 {
    system {
        host-name spice-re1;
    }
    interfaces {
        fxp0 {
            unit 0 {
                family inet {
                    address 192.168.70.72/21;
                }
            }
        }
    }
}
}
global;
}
apply-groups [ re0 re1 ];
system {
    domain-name englab.juniper.net;
    backup-router 192.168.71.254;
    root-authentication {
        encrypted-password "$ABC123"; ## SECRET-DATA
    }
    name-server {
        192.168.1.1;
    }
    login {
        user user {
            uid 2001;
            class super-user;
            authentication {
                encrypted-password "$ABC123"; ## SECRET-DATA
            }
        }
    }
    services {
        telnet;
    }
    syslog {
        user * {
            any emergency;
        }
        file messages {
            any notice;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
}
routing-options {

```

```
static {  
    /* corporate office */  
    route 172.16.0.0/12 {  
        next-hop 192.168.71.254;  
        retain;  
        no-readvertise;  
    }  
}
```

2. Commit and synchronize the configuration. The **commit synchronize** command commits this new configuration on both Routing Engines simultaneously.

```
[edit]  
root# commit synchronize  
re0:  
configuration check succeeds  
re1:  
commit complete  
re0:  
commit complete
```

If you receive an error message after you issue the **commit** statement, you can review the configuration using the **show** command to find the errors in your configuration. You can delete incorrect entries using the **delete** command. For example, to delete a hostname from the configuration, issue the following command:

```
[edit]  
root# delete system host-name host-name
```

3. Exit configuration mode.

```
[edit]  
root# exit  
Exiting configuration mode  
  
root>
```

Saving a Rescue Configuration File

A rescue configuration file is helpful in the event that your device's configuration file has been misconfigured. You can restore the device to this rescue configuration to bring the device back online. If you save this file off the device, the rescue configuration can also be used to restore your device in the event of a software failure.

To save a current device configuration as a rescue configuration file:

1. Edit the configuration file on the device to reflect the base configuration you wish to use.

For more information about editing the configuration, see *Overview for Routing Devices*.

2. In the CLI operational mode, save this edited base configuration as the rescue configuration file:

```
user@host> request system configuration rescue save
```

The rescue configuration file is automatically saved under **/config** directory.

3. Copy the rescue configuration to a remote server:

```
user@host1% cd /config/
user@host1% ls -lrt rescue.conf.gz

user@host1%ftp host2
Name: username
Password: password
User user logged in.
ftp> cd /var/tmp
ftp> lcd /config
ftp> bin
ftp> put rescue.conf.gz
local: rescue.conf.gz remote: rescue.conf.gz

Transfer complete.
ftp> bye
Goodbye.
```

To roll back to the rescue configuration, use the **rollback rescue** command.

```
user@host# rollback rescue

load complete
```



NOTE: After rolling back to the rescue configuration, you must commit the configuration to activate it:

```
user@host#commit
```

Restoring a Saved Configuration

To restore a saved configuration, perform the following tasks:

1. [Copy Saved Files to the Router on page 223](#)
2. [Loading and Committing the Configuration File on page 224](#)

Copy Saved Files to the Router

To copy the saved configuration to the router:

1. Log in to the console as **root**. There is no password.

```
Escape character is '^]'.
[Enter]
router (ttyd0)

login: root
Password: [Enter]
```

Initially, access to the router is limited to the console port after a recovery installation. Access through the management ports and interfaces is set in the configuration. For information about accessing the router through the console port, see the administration guide for your particular router.

2. Start the CLI:

```
# cli
```

3. Copy the configuration file on the remote server to the router's `/var/tmp` directory:

```
root@host> ftp remote-server
user: username
password: password
ftp> bin
Type set to I.
ftp> get /path/file
ftp> bye
Goodbye.
```

Loading and Committing the Configuration File

Once the saved configuration file is copied to the router, you load and commit the file:

1. Start the CLI configuration mode.

```
user@routername> configure
Entering configuration mode
```

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

2. Load the file into the current configuration. You should override the existing file.

```
user@host#
load override /var/tmp/filename
load complete
```

3. Commit the file.

```
user@host# commit
commit complete
```

4. Exit the CLI configuration mode.

```
user@host# exit
user@host>
```

5. Back up Junos OS.

After you have installed the software on the router, committed the configuration, and are satisfied that the new configuration is successfully running, issue the **request system snapshot** command to back up the new software to the `/altconfig` file system. If you do not issue the **request system snapshot** command, the configuration on the alternate boot drive will be out of sync with the configuration on the primary boot drive.

The **request system snapshot** command causes the root file system to be backed up to `/altroot`, and `/config` to be backed up to `/altconfig`. The root and `/config` file systems are on the router's CompactFlash card, and the `/altroot` and `/altconfig` file systems are on the router's hard disk or solid-state drive (SSD).

Reverting to the Default Factory Configuration by Using the `request system zeroize` Command

The `request system zeroize` command is a standard Junos OS operational mode command that removes all configuration information and resets all key values. The operation unlinks all user-created data files, including customized configuration and log files, from their directories. The switch then reboots and reverts to the factory-default configuration.

To completely erase user-created data so that it is unrecoverable, use the `request system zeroize media` command.



CAUTION: Before issuing `request system zeroize`, use the `request system snapshot` command to back up the files currently used to run the switch to a secondary device.

To revert to the factory-default configuration by using the `request system zeroize` command:

1. `user@switch> request system zeroize`
warning: System will be rebooted and may not boot without configuration
Erase all data, including configuration and log files? [yes,no] (yes)
2. Type **yes** to remove configuration and log files and revert to the factory default configuration.
3. Complete the initial configuration of the switch.

Related Documentation • [request system zeroize on page 466](#)

Reverting to the Rescue Configuration

If someone inadvertently commits a configuration that denies management access to a device and the console port is not accessible, you can overwrite the invalid configuration and replace it with the rescue configuration. The rescue configuration is a previously committed, valid configuration.

To revert the switch to the rescue configuration:

1. Enter the `load override` command.

[edit]
`user@switch# load override filename`
2. Commit your changes.

[edit]
`user@switch# commit filename`

- Related Documentation**
- *Reverting to the Default Factory Configuration*

CHAPTER 11

Reinstalling Software

- Checklist for Reinstalling Junos OS on page 227
- Log the Software Version Information on page 229
- Log the Hardware Version Information on page 230
- Log the Chassis Environment Information on page 231
- Log the System Boot-Message Information on page 231
- Log the Active Configuration on page 233
- Log the Interfaces on the Router on page 233
- Log the BGP, IS-IS, and OSPF Adjacency Information on page 234
- Log the System Storage Information on page 235
- Back Up the Currently Running and Active File System on page 236
- Reinstall Junos OS on page 237
- Reconfigure Junos OS on page 237
- Configure Host Names, Domain Names, and IP Addresses on page 241
- Protecting Network Security by Configuring the Root Password on page 243
- Check Network Connectivity on page 245
- Copy Backup Configurations to the Router on page 245
- After You Reinstall Junos OS on page 245
- Compare Information Logged Before and After the Reinstall on page 246
- Back Up the New Software on page 247

Checklist for Reinstalling Junos OS

Table 28 on page 227 provides links and commands for reinstalling Junos OS.

Table 28: Checklist for Reinstalling Junos OS

Tasks	Command or Action
Before You Reinstall Junos OS	
1. Log the Software Version Information on page 229	<code>show version</code> <code>save filename</code>

Table 28: Checklist for Reinstalling Junos OS (*continued*)

Tasks	Command or Action
2. Log the Hardware Version Information on page 230	<code>show chassis hardware save filename</code>
3. Log the Chassis Environment Information on page 231	<code>show chassis environment save filename</code>
4. Log the System Boot-Message Information on page 231	<code>show system boot-messages save filename</code>
5. Log the Active Configuration on page 233	<code>show configuration save filename</code>
6. Log the Interfaces on the Router on page 233	<code>show interface terse save filename</code>
7. Log the BGP, IS-IS, and OSPF Adjacency Information on page 234	<code>show bgp summary save filename</code> <code>show isis adjacency brief save filename</code> <code>show ospf neighbor brief save filename</code>
8. Log the System Storage Information on page 235	<code>show system storage save filename</code>
9. Back Up the Currently Running and Active File System on page 236	<code>request system snapshot</code>
10.	http://www.juniper.net/support
“Reinstall Junos OS” on page 237	
“Reconfigure Junos OS” on page 237	
1. Configure Host Names, Domain Names, and IP Addresses on page 237	Log in as root. Start the CLI. Enter configuration mode: <code>configure</code> <code>set system host-name host-name</code> <code>set system domain-name domain-name</code> <code>set interfaces fxp0 unit 0 family inet address address/prefix-length</code> <code>set system backup-router address</code> <code>set system name-server address</code>
2. Protecting Network Security by Configuring the Root Password on page 239	<code>set system root-authentication plain-text-password</code> <code>set system root-authentication encrypted-password password</code> <code>set system root-authentication ssh-rsa key</code> <code>commit</code> <code>exit</code>
3. Check Network Connectivity on page 240	<code>ping address</code>
4. Copy Backup Configurations to the Router on page 241	<code>file copy var/tmp</code> <code>configure</code> <code>[edit]</code> <code>load merge /config/filename or load replace /config/filename</code> <code>[edit]</code> <code>commit</code>

Table 28: Checklist for Reinstalling Junos OS (*continued*)

Tasks	Command or Action
“After You Reinstall Junos OS” on page 245	
1. Compare Information Logged Before and After the Reinstall on page 245	show version save <i>filename</i> show chassis hardware save <i>filename</i> show chassis environment save <i>filename</i> show system boot-messages save <i>filename</i> show configuration save <i>filename</i> show interfaces terse save <i>filename</i> show bgp summary show isis adjacency brief show ospf neighbor brief save <i>filename</i> show system storage save <i>filename</i>
2. Back Up the New Software on page 246	request system snapshot

Log the Software Version Information

Purpose The purpose of this action is to log the Junos OS version information.

Action Use the following Junos OS CLI operational mode command:

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

Sample Output

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

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

```
RMOPD release 5.0R5 built by builder on 2002-03-02 04:46:11 UTC
jkernel-dd release 5.0R5 built by builder on 2002-03-02 04:41:07 UTC
jroute-dd release 5.0R5 built by builder on 2002-03-02 04:41:21 UTC
jdocs-dd release 5.0R5 built by builder on 2002-03-02 04:39:11 UTC
```

Meaning The sample output shows the hostname, router model, and the different Junos OS packages, processes, and documents.

Related Documentation •

Log the Hardware Version Information

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

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

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

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

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

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

Item	Version	Part number	Serial number	Description
Chassis			101	M160
Midplane	REV 02	710-001245	S/N AB4107	
FPM CMB	REV 01	710-001642	S/N AA2911	
FPM Display	REV 01	710-001647	S/N AA2999	
CIP	REV 02	710-001593	S/N AA9563	
PEM 0	Rev 01	740-001243	S/N KJ35769	DC
PEM 1	Rev 01	740-001243	S/N KJ35765	DC
PCG 0	REV 01	710-001568	S/N AA9794	
PCG 1	REV 01	710-001568	S/N AA9804	
Host 1			da000004f8d57001	teknor
MCS 1	REV 03	710-001226	S/N AA9777	
SFM 0 SPP	REV 04	710-001228	S/N AA2975	
SFM 0 SPR	REV 02	710-001224	S/N AA9838	Internet Processor I
SFM 1 SPP	REV 04	710-001228	S/N AA2860	
SFM 1 SPR	REV 01	710-001224	S/N AB0139	Internet Processor I
FPC 0	REV 03	710-001255	S/N AA9806	FPC Type 1
CPU	REV 02	710-001217	S/N AA9590	
PIC 1	REV 05	750-000616	S/N AA1527	1x OC-12 ATM, MM

PIC 2	REV 05	750-000616	S/N AA1535	1x OC-12 ATM, MM
PIC 3	REV 01	750-000616	S/N AA1519	1x OC-12 ATM, MM
FPC 1	REV 02	710-001611	S/N AA9523	FPC Type 2
CPU	REV 02	710-001217	S/N AA9571	
PIC 0	REV 03	750-001900	S/N AA9626	1x STM-16 SDH, SMIR
PIC 1	REV 01	710-002381	S/N AD3633	2x G/E, 1000 BASE-SX
FPC 2				FPC Type OC192
CPU	REV 03	710-001217	S/N AB3329	
PIC 0	REV 01			1x OC-192 SM SR-2

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

Log the Chassis Environment Information

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

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

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

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

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

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

Log the System Boot-Message Information

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

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

```

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

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

JUNOS 4.1-20000216-Zf8469 #0: 2000-02-16 12:57:28 UTC

tlim@device1.example.com:/p/build/20000216-0905/4.1/release_kernel/sys/compile/GENERIC
CPU: Pentium Pro (332.55-MHz 686-class CPU)
    Origin = "GenuineIntel" Id = 0x66a Stepping=10

Features=0x183f9ff<FPU,VME,DE,PSE,TSC,MSR,PAE,MCE,CX8,SEP,MTRR,PGE,MCA,CMOV,<b16>,<b17>,MMX,<b24>>
Teknor CPU Card Recognized
real memory = 805306368 (786432K bytes)
avail memory = 786280448 (767852K bytes)
Probing for devices on PCI bus 0:
chip0 <generic PCI bridge (vendor=8086 device=7192 subclass=0)> rev 3 class 60000
    on pci0:0:0
chip1 <Intel 82371AB PCI-ISA bridge> rev 1 class 60100 on pci0:7:0
chip2 <Intel 82371AB IDE interface> rev 1 class 10180 on pci0:7:1
chip3 <Intel 82371AB USB interface> rev 1 class c0300 int d irq 11 on pci0:7:2
smb0 <Intel 82371AB SMB controller> rev 1 class 68000 on pci0:7:3
pcic0 <TI PCI-1131 PCI-CardBus Bridge> rev 1 class 60700 int a irq 15 on pci0:13:0
TI1131 PCI Config Reg: [pci only][FUNC0 pci int]
pcic1 <TI PCI-1131 PCI-CardBus Bridge> rev 1 class 60700 int b irq 12 on pci0:13:1
TI1131 PCI Config Reg: [pci only][FUNC1 pci int]
fxp0 <Intel EtherExpress Pro 10/100B Ethernet> rev 8 class 20000 int a irq 12 on
    pci0:16:0
chip4 <generic PCI bridge (vendor=1011 device=0022 subclass=4)> rev 4 class 60400
    on pci0:17:0
fxp1 <Intel EtherExpress Pro 10/100B Ethernet> rev 8 class 20000 int a irq 10 on
    pci0:19:0
Probing for devices on PCI bus 1:mcs0 <Miscellaneous Control Subsystem> rev 12
class ff0000 int a irq 12 on pci1:13:0
fxp2 <Intel EtherExpress Pro 10/100B Ethernet> rev 8 class 20000 int a irq 10 on
    pci1:14:0
Probing for devices on the ISA bus:
sc0 at 0x60-0x6f irq 1 on motherboard
sc0: EGA color <16 virtual consoles, flags=0x0>
ed0 not found at 0x300
ed1 not found at 0x280
ed2 not found at 0x340
psm0 not found at 0x60
sio0 at 0x3f8-0x3ff irq 4 flags 0x20010 on isa
sio0: type 16550A, console
sio1 at 0x3e8-0x3ef irq 5 flags 0x20000 on isa
sio1: type 16550A
sio2 at 0x2f8-0x2ff irq 3 flags 0x20000 on isa
sio2: type 16550A
pcic0 at 0x3e0-0x3e1 on isa
PC-Card ctlr(0) TI PCI-1131 [CardBus bridge mode] (5 mem & 2 I/O windows)
pcic0: slot 0 controller I/O address 0x3e0
npx0 flags 0x1 on motherboard
npx0: INT 16 interface
fdc0: direction bit not set

```

```

fdc0: cmd 3 failed at out byte 1 of 3
fdc0 not found at 0x3f0
wdc0 at 0x1f0-0x1f7 irq 14 on isa
wdc0: unit 0 (wd0): <SunDisk SDCFB-80>, single-sector-i/o
wd0: 76MB (156672 sectors), 612 cyls, 8 heads, 32 S/T, 512 B/S
wdc0: unit 1 (wd1): <IBM-DCXA-210000>
wd1: 8063MB (16514064 sectors), 16383 cyls, 16 heads, 63 S/T, 512 B/S
wdc1 not found at 0x170
wdc2 not found at 0x180
ep0 not found at 0x300
fxp0: Ethernet address 00:a0:a5:12:05:5a
fxp1: Ethernet address 00:a0:a5:12:05:59
fxp2: Ethernet address 02:00:00:00:00:01
swapon: adding /dev/wd1s1b as swap device
Automatic reboot in progress...
/dev/rwd0s1a: clean, 16599 free (95 frags, 2063 blocks, 0.1% fragmentation)
/dev/rwd0s1e: clean, 9233 free (9 frags, 1153 blocks, 0.1% fragmentation)
/dev/rwd0s1a: clean, 16599 free (95 frags, 2063 blocks, 0.1% fragmentation)
/dev/rwd1s1f: clean, 4301055 free (335 frags, 537590 blocks, 0.0% fragmentation)

```

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

Log the Active Configuration

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

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

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

```

user@host> show configuration
system {
  host-name lab8;
  domain-name device1.example.com;
  backup-router 10.1.1.254;
    time-zone America/Los_Angeles;
  default-address-selection;
    dump-on-panic;
  name-server {
    [...]
  }
}
[...Output truncated...]

```

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

Log the Interfaces on the Router

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

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

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

```
user@host> show interfaces terse
Interface      Admin Link Proto Local Remote
at-1/3/0       up    up    inet  203.0.113.1    --> 203.0.113.2
at-1/3/0.0     up    up    inet  203.0.113.1    --> 203.0.113.2
                iso
fxp0           up    up
fxp0.0         up    up    inet  10.168.5.59/24
gre            down  up
ipip           down  up
lo0            up    up
lo0.0          up    up    inet  127.0.0.1      --> 0/0
                iso 47.0005.80ff.f800.0000.0108.0001.1921.6800.5059.00
so-1/2/0       up    down
so-1/2/1       down  down
so-1/2/2       down  down
so-1/2/3       down  down
so-2/0/0       up    up
so-2/0/0.0     up    up    inet  192.2.3.4      --> 192.2.3.5
                iso

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

Meaning The sample output displays summary information about the physical and logical interfaces on the router.

Log the BGP, IS-IS, and OSPF Adjacency Information

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

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

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

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

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

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

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

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

```
user@host> show ospf neighbor brief
Address      Intf      State      ID          Pri  Dead
10.168.254.225 fxp3.0    2Way      10.250.240.32 128  36
10.168.254.230 fxp3.0    Full      10.250.240.8  128  38
10.168.254.229 fxp3.0    Full      10.250.240.35 128  33
10.1.1.129      fxp2.0    Full      10.250.240.12 128  37
10.1.1.131      fxp2.0    Full      10.250.240.11 128  38
10.1.2.1        fxp1.0    Full      10.250.240.9  128  32
10.1.2.81       fxp0.0    Full      10.250.240.10 128  33
```

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

Log the System Storage Information

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

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

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

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

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

Back Up the Currently Running and Active File System

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

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

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

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



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

Reinstall Junos OS

Action To reinstall Junos OS, follow these steps:

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

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

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

4. Remove the removable medium when prompted.

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

Reconfigure Junos OS

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

To reconfigure the software, follow these steps:

1. [Configure Host Names, Domain Names, and IP Addresses on page 237](#)
2. [Protecting Network Security by Configuring the Root Password on page 239](#)
3. [Check Network Connectivity on page 240](#)
4. [Copy Backup Configurations to the Router on page 241](#)

Configure Host Names, Domain Names, and IP Addresses

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

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

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

3. Enter configuration mode:

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

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

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

5. Configure the machine's domain name:

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

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

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

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

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

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

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

Protecting Network Security by Configuring the Root Password

Configuring the root password on your Junos OS-enabled router helps prevent unauthorized users from making changes to your network. The root user (also referred to as superuser) has unrestricted access and full permissions within the system, so it is crucial to protect these functions by setting a strong password when setting up a new router.

After a new router is initially powered on, you log in as the user **root** with no password. Junos OS requires configuration of the root password before it accepts a commit operation. On a new device, the root password must always be a part of the configuration submitted with your initial commit.

To set the root password, you have a few options as shown in Step 1 of the following procedure.

- Enter a plain-text password that Junos OS encrypts.
- Enter a password that is already encrypted.
- Enter a secure shell (ssh) public key string.

The most secure options of these three are using an already encrypted password or an ssh public key string. Pre-encrypting your password or using a ssh public key string means the plain-text version of your password will never be transferred over the internet, protecting it from being intercepted by a man-in-the-middle attack.



BEST PRACTICE: Optionally, instead of configuring the root password at the **[edit system]** hierarchy level, you can use a configuration group to strengthen security, as shown in Step 2 of this procedure. This step uses a group called **global** as an example.

To set the root password:

1. Use one of these methods to configure the root password:

- To enter a plain-text password that the system encrypts for you:

```
[edit groups global system]
root@# set root-authentication plain-text-password
New Password: type password here
Retype new password: retry password here
```

If you use a plain-text password, Junos OS displays the password as an encrypted string so that users viewing the configuration cannot see it. As you enter the password in plain text, Junos OS encrypts it immediately. You do not have to configure Junos OS to encrypt the password as in some other systems. Plain-text passwords are hidden and marked as **## SECRET-DATA** in the configuration.

- To enter a password that is already encrypted:



CAUTION: Do not use the `encrypted-password` option unless the password is *already* encrypted, and you are entering the encrypted version of the password.

If you accidentally configure the `encrypted-password` option with a plain-text password or with blank quotation marks (" "), you will not be able to log in to the device as root, and you will need to complete the root password recovery process.

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

- To enter an ssh public key string:

```
[edit groups global system]
root@# set root-authentication (ssh-dsa | ssh-eccdsa | ssh-rsa key)
```

2. (Optional) Strengthen security by only allowing root access from the console port.

```
[edit groups global system]
root@# set services ssh root-login deny
```

3. If you used a configuration group in Step 2, apply the configuration group, substituting **global** with the appropriate group name.

```
[edit]
user@host# set apply-groups global
```

4. Commit the changes.

```
root@# commit
```

Check Network Connectivity

Purpose Establish that the router has network connectivity.

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

```
root@> ping address
```

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

Copy Backup Configurations to the Router

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

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

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

2. Load and activate the desired configuration:

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

Configure Host Names, Domain Names, and IP Addresses

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

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

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

3. Enter configuration mode:

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

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

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

5. Configure the machine's domain name:

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

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

[edit]

root@# **set interfaces fxp0 unit 0 family inet address *address* / *prefix-length***

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

[edit]

root@# **set system backup-router *address***

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

[edit]

root@# **set system name-server *address***

Protecting Network Security by Configuring the Root Password

Configuring the root password on your Junos OS-enabled router helps prevent unauthorized users from making changes to your network. The root user (also referred to as superuser) has unrestricted access and full permissions within the system, so it is crucial to protect these functions by setting a strong password when setting up a new router.

After a new router is initially powered on, you log in as the user **root** with no password. Junos OS requires configuration of the root password before it accepts a commit operation. On a new device, the root password must always be a part of the configuration submitted with your initial commit.

To set the root password, you have a few options as shown in Step 1 of the following procedure.

- Enter a plain-text password that Junos OS encrypts.
- Enter a password that is already encrypted.
- Enter a secure shell (ssh) public key string.

The most secure options of these three are using an already encrypted password or an ssh public key string. Pre-encrypting your password or using a ssh public key string means the plain-text version of your password will never be transferred over the internet, protecting it from being intercepted by a man-in-the-middle attack.



BEST PRACTICE: Optionally, instead of configuring the root password at the **[edit system]** hierarchy level, you can use a configuration group to strengthen security, as shown in Step 2 of this procedure. This step uses a group called **global** as an example.

To set the root password:

1. Use one of these methods to configure the root password:

- To enter a plain-text password that the system encrypts for you:

```
[edit groups global system]
root@# set root-authentication plain-text-password
New Password: type password here
Retype new password: retry password here
```

If you use a plain-text password, Junos OS displays the password as an encrypted string so that users viewing the configuration cannot see it. As you enter the password in plain text, Junos OS encrypts it immediately. You do not have to configure Junos OS to encrypt the password as in some other systems. Plain-text passwords are hidden and marked as **## SECRET-DATA** in the configuration.

- To enter a password that is already encrypted:



CAUTION: Do not use the encrypted-password option unless the password is *already* encrypted, and you are entering the encrypted version of the password.

If you accidentally configure the encrypted-password option with a plain-text password or with blank quotation marks (" "), you will not be able to log in to the device as root, and you will need to complete the root password recovery process.

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

- To enter an ssh public key string:

```
[edit groups global system]
root@# set root-authentication (ssh-dsa | ssh-eccdsa | ssh-rsa key)
```

2. (Optional) Strengthen security by only allowing root access from the console port.

```
[edit groups global system]
root@# set services ssh root-login deny
```

3. If you used a configuration group in Step 2, apply the configuration group, substituting **global** with the appropriate group name.

```
[edit]
user@host# set apply-groups global
```

4. Commit the changes.

```
root@# commit
```

Related Documentation

- [Accessing a Junos OS Device the First Time on page 55](#)
- [Understanding User Accounts](#)
- [Recovering the Root Password](#)

Check Network Connectivity

Purpose Establish that the router has network connectivity.

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

```
root@> ping address
```

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

Copy Backup Configurations to the Router

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

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

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

2. Load and activate the desired configuration:

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

After You Reinstall Junos OS

To verify that the new version of the Junos OS is running as expected after the reinstall, follow these steps:

1. [Compare Information Logged Before and After the Reinstall on page 245](#)
2. [Back Up the New Software on page 246](#)

Compare Information Logged Before and After the Reinstall

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

Action To obtain system information, use the following commands:

```
user@host> show version
user@host> show chassis hardware
user@host> show chassis environment
user@host> show system boot-messages
```

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

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

Back Up the New Software

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

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

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

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



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

Compare Information Logged Before and After the Reinstall

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

Action To obtain system information, use the following commands:

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

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

Back Up the New Software

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

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

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

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



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

Downgrading Software

- [Downgrading Junos OS from Upgraded FreeBSD on page 249](#)

Downgrading Junos OS from Upgraded FreeBSD

Starting in Junos OS Release 15.1, certain hardware platforms run a Junos OS based on an upgraded FreeBSD kernel instead of older versions of FreeBSD. If you have previously upgraded to Junos OS with upgraded FreeBSD, you can downgrade to earlier versions of Junos OS, as long as the downgrade conforms to the Junos OS policy of skipping at most two earlier releases.

Before you begin:

1. Verify that you have previously upgraded to Junos OS with the upgraded FreeBSD kernel, as described in [“Upgrading Junos OS with Upgraded FreeBSD” on page 153](#).
2. Download the Junos OS package.

Select and perform the procedure that matches your conditions:

- [Downgrading from Junos OS with Upgraded FreeBSD to Junos OS on page 249](#)
- [Installing Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Different Release on page 250](#)

Downgrading from Junos OS with Upgraded FreeBSD to Junos OS

This example uses the package `/var/tmp/jinstall-13.3R2.7-domestic-signed.tgz` to install Junos OS with a pre-upgraded FreeBSD kernel on the master Routing Engine (`re0`).



NOTE: The following procedure refers to routers, but it also applies to switches.

To downgrade from Junos OS with upgraded FreeBSD to Junos OS:

1. Enter the **`request system software add package-name no-validate reboot`** command from the operational mode in the CLI.

Use the **`no-validate`** and **`reboot`** options with the **`request system software add`** command. If you leave out the **`no-validate`** option, the command uses the **`validate`** option by

default, and direct validation of running configuration does not work for downgrading to Junos OS from Junos OS with upgraded FreeBSD.



NOTE: To validate current configuration on an downgrade to Junos OS from Junos OS with upgraded FreeBSD, use the `request system software validate on (Junos OS with Upgraded FreeBSD)` command.

If you leave out the **reboot** option, you can take care of that in a separate reboot step.

The following example uses the **re0** option:

```
user@host>request system software add
/var/tmp/jinstall-13.3R2.7-domestic-signed.tgz re0 no-validate reboot
THIS IS A SIGNED PACKAGE Saving the config files ...
NOTICE: uncommitted changes have been saved in
/var/db/config/juniper.conf.pre-install Rebooting. Please wait ...
shutdown: [pid 11001] Shutdown NOW! *** FINAL System shutdown message
from root@host *** System going down IMMEDIATELY Shutdown NOW! System
shutdown time has arrived\x07\x07 users@host> Connection to
device1.example.com closed by remote host. Connection to
device1.example.com closed. ... user@router> show version
Hostname: host
Model: mx240
Junos: 13.3R2.7
JUNOS Base OS boot [13.3R2.7]
JUNOS Base OS Software Suite [13.3R2.7]
JUNOS Kernel Software Suite [13.3R2.7]
JUNOS Crypto Software Suite [13.3R2.7]
JUNOS Packet Forwarding Engine Support (M/T/EX Common) [13.3R2.7]
JUNOS Packet Forwarding Engine Support (MX Common) [13.3R2.7]
JUNOS Online Documentation [13.3R2.7]
JUNOS Services AACL Container package [13.3R2.7]
...
```

2. Verify the downgrade of the software package.

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



NOTE: The output shows the OS kernel, OS runtime, and other packages installed on the router.

Installing Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Different Release



NOTE: If you have important files in other directories, copy them from the router or switch to a secure location before upgrading the router or switch.



NOTE: The following procedure refers to routers, but it also applies to switches.

To install Junos OS with upgraded FreeBSD over Junos OS with upgraded FreeBSD of a different release:

1. Enter the **request system software add *package-name* validate reboot** command from the operational mode in the CLI:



NOTE: The **no-copy** option is enabled by default.

Use the **validate** and **reboot** options with the **request system software add** command. The command uses the **validate** option by default. We encourage users to validate using the **validate** option when upgrading from Junos OS to Junos OS, or from Junos OS with upgraded FreeBSD to Junos OS with upgraded FreeBSD.

If you leave out the **reboot** option, you can take care of that in a separate reboot step.

The new Junos OS image is installed on the router.

2. Verify the installation of Junos OS with upgraded FreeBSD:

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



NOTE: The output shows the OS kernel, OS runtime, and other packages installed on the router.

Related Documentation

- [Upgrading Junos OS with Upgraded FreeBSD on page 153](#)
- [Understanding Junos OS with Upgraded FreeBSD on page 17](#)
- [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 414](#)
- [request system reboot \(Junos OS with Upgraded FreeBSD\) on page 398](#)

CHAPTER 13

Rebooting or Halting Software Processes on a Device

- [Restarting and Halting SRX Series Devices on page 253](#)
- [Rebooting or Halting the EX Series Switch \(J-Web Procedure\) on page 258](#)

Restarting and Halting SRX Series Devices

This topic includes the following sections:

- [Rebooting SRX Series Devices on page 253](#)
- [Halting SRX Series Devices on page 255](#)
- [Bringing Chassis Components Online and Offline on SRX Series Devices on page 257](#)
- [Restarting the Chassis on SRX Series Devices on page 258](#)

Rebooting SRX Series Devices

This example shows how to reboot a SRX Series device.

- [Requirements on page 253](#)
- [Overview on page 253](#)
- [Configuration on page 253](#)
- [Verification on page 255](#)

Requirements

Before rebooting the device, save and commit any Junos OS updates.

Overview

This example shows how to reboot a device fifty minutes from when you set the time from the internal media while sending a text message of 'stop' to all system users before the device reboots.

Configuration

CLI Quick Configuration To quickly configure this section of the example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your

network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

From operational mode, enter:

```
user@host> request system reboot at 5 in 50 media internal message stop
```

GUI Step-by-Step Procedure

To reboot a device:

1. In the J-Web user interface, select **Maintain>Reboot**.
2. Select **Reboot in 50 minutes** to reboot the device fifty minutes from the current time.
3. Select the **internal** (for SRX Series devices) boot device from the Reboot From Media list.
4. In the Message box, type **stop** as the message to display to any user on the device before the reboot occurs.
5. Click **Schedule**. The J-Web user interface requests confirmation to perform the reboot.
6. Click **OK** to confirm the operation.
 - If the reboot is scheduled to occur immediately, the device reboots. You cannot access J-Web until the device has restarted and the boot sequence is complete. After the reboot is complete, refresh the browser window to display the J-Web login page.
 - If the reboot is scheduled to occur in the future, the Reboot page displays the time until reboot. You have the option to cancel the request by clicking **Cancel Reboot** on the J-Web user interface Reboot page.
7. Click **OK** to check your configuration and save it as a candidate configuration.
8. If you are done configuring the device, click **Commit Options>Commit**.

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To reboot a device:

From operational mode, schedule a reboot of the device to occur fifty minutes from when you set the time from the internal media while sending a text message of 'stop' to all system users before the device reboots.

Enter:

```
user@host> request system reboot at 5 in 50 media internal message stop
```

Results From configuration mode, confirm your configuration by entering the **show system** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- [Verifying the Device Reboot on page 255](#)

Verifying the Device Reboot

Purpose Verify that the device rebooted.

Action From operational mode, enter the **show system** command.

Halting SRX Series Devices

This example shows how to halt a device.

- [Requirements on page 255](#)
- [Overview on page 255](#)
- [Configuration on page 256](#)
- [Verification on page 257](#)

Requirements

Before halting the device, save and commit any Junos OS updates.

Overview

When the device is halted, all software processes stop and you can access the device through the console port only. Reboot the device by pressing any key on the keyboard.



NOTE: If you cannot connect to the device through the console port, shut down the device by pressing and holding the power button on the front panel until the **POWER LED** turns off. After the device has shut down, you can power on the device by pressing the power button again. The **POWER LED** turns on during startup and remains steadily green when the device is operating normally.

This example shows how to halt the system and stop software processes on the device immediately.

Configuration

CLI Quick Configuration

To quickly configure this section of the example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

From operational mode, enter:

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



NOTE: The **request system halt** command used for halting the system and stopping software processes on the device is not supported on SRX1500, SRX4100, and SRX4200 devices.

GUI Step-by-Step Procedure

To halt a device immediately:

1. In the J-Web user interface, select **Maintain>Reboot**.
2. Select **Halt Immediately**. After the software stops, you can access the device through the console port only.
3. Click **Schedule**. The J-Web user interface requests confirmation to halt.
4. Click **OK** to confirm the operation. If the device halts, all software processes stop and you can access the device through the console port only. Reboot the device by pressing any key on the keyboard.
5. Click **OK** to check your configuration and save it as a candidate configuration.
6. If you are done configuring the device, click **Commit Options>Commit**.

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To halt a device:

From operational mode, halt the SRX Series device immediately.

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

Results From configuration mode, confirm your configuration by entering the **show system** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- [Verifying the Device Halt on page 257](#)

Verifying the Device Halt

Purpose Verify that the device halted.

Action From operational mode, enter the **show system** command.

Bringing Chassis Components Online and Offline on SRX Series Devices

You can use the **request** commands to bring chassis components (except Power Entry Modules and fans) online and offline.

To bring chassis components online and offline, enter these **request chassis** commands:

```
user@host> request chassis <fru> slot <slot#> pic <pic#> online
```

```
user@host> request chassis <fru> slot <slot#> pic <pic#> offline
```

Where **<fru>** in the request chassis command can be any of the following (for SRX300, SRX320, SRX340, SRX345, and SRX550M devices):

- **fpc**—Changes the Flexible PIC Concentrator (FPC) status.

Where **<fru>** in the request chassis command can be any of the following (for SRX5800, SRX5600, and SRX5400 devices):

- **cb**—Changes the control board status.
- **fabric**—Changes the fabric status.
- **fpc**—Changes the Flexible PIC Concentrator (FPC) status.
- **fpm**—Changes the craft interface status.
- **pic**—Changes the physical interface card status.
- **routing-engine**—Changes the routing engine status.



NOTE: The **request chassis** command is not supported for bringing SPCs online and offline.

Example:

To bring specific pic and the corresponding fpc slot online, from operational mode enter the following **request chassis** command:

```
user@host> request chassis pic pic-slot 1 fpc-slot 1 online
```

Restarting the Chassis on SRX Series Devices

You can restart the chassis using the **restart chassis-control** command with the following options:

- To restart the process gracefully:

```
user@host> restart chassis-control gracefully
```

- To restart the process immediately:

```
user@host> restart chassis-control immediately
```

- To restart the process softly:

```
user@host> restart chassis-control soft
```

Rebooting or Halting the EX Series Switch (J-Web Procedure)

You can use the J-Web interface to schedule a reboot or to halt the switching platform.

To reboot or halt the switching platform by using the J-Web interface:

1. In the J-Web interface, select **Maintain > Reboot**.
2. Select one:
 - **Reboot Immediately**—Reboots the switching platform immediately.
 - **Reboot in *number of minutes***—Reboots the switch in the number of minutes from now that you specify.
 - **Reboot when the system time is *hour:minute***—Reboots the switch at the absolute time that you specify, on the current day. You must select a 2-digit hour in 24-hour format and a 2-digit minute.
 - **Halt Immediately**— Stops the switching platform software immediately. After the switching platform software has stopped, you can access the switching platform through the console port only.
3. (Optional) In the Message box, type a message to be displayed to any users on the switching platform before the reboot occurs.
4. Click **Schedule**. The J-Web interface requests confirmation to perform the reboot or halt.
5. Click **OK** to confirm the operation.
 - If the reboot is scheduled to occur immediately, the switch reboots. You cannot access the J-Web interface until the switch has restarted and the boot sequence

is complete. After the reboot is complete, refresh the browser window to display the J-Web interface login page.

- If the reboot is scheduled to occur in the future, the Reboot page displays the time until reboot. You have the option to cancel the request by clicking **Cancel Reboot** on the J-Web interface Reboot page.
- If the switch is halted, all software processes stop and you can access the switching platform through the console port only. Reboot the switch by pressing any key on the keyboard.

Related Documentation • *Starting the J-Web Interface*

RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Overview

- [RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX with VM Host Support on page 261](#)
- [Architecture of RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines on page 262](#)
- [Salient Features of the RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines on page 263](#)
- [Boot Process Overview on page 267](#)
- [VM Host Installation on page 268](#)
- [Copying VM Host Installation Package to the PXE Boot Server on page 272](#)
- [Upgrading the SSD Firmware on RE-S-X6-64G and RE-MX2K-X8-64G Routing Engines on page 275](#)
- [Disabling Autorecovery on RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines on page 277](#)
- [VM Host Operations and Management on page 277](#)

RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX with VM Host Support

The Routing Engines RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX not only provide increased control plane scalability and performance but also provide virtualization capabilities to the Junos OS infrastructure to support greater computing demands.

Virtualization enables multiple instances of operating systems, called guests, to run concurrently on the host and share virtualized hardware resources. A guest is a virtual machine (VM) that runs on a hypervisor-based host and shares its resources. A host is a virtualized software whose hypervisor allows multiple guest VMs to run on it concurrently and share its resources. A VM can be an instance of Junos OS or any compatible third-party VM. Each VM runs its own operating system image and applications that can be different from that of another VM running on the same host.

On the RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX Routing Engines, one instance of Junos OS runs as a VM over a Linux-based host (VM host) and serves as the VM operating in the administrative context. Junos OS manages all configurations, chassis control, communication with the host OS, and user interface command execution, thus providing near-native Junos OS experience to the end user.

Table 29 on page 262 lists the hardware specifications of the Routing Engines.

Table 29: Hardware Specifications of the RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX, Routing Engines

Model Number	Supported on Router	Specifications
RE-S-X6-64G	MX240, MX480, and MX960	<ul style="list-style-type: none"> 6-core Haswell CPU Wellsburg PCH-based Routing Engine with 64-GB DRAM and two 64-GB solid-state drives (SSDs)
REMX2K-X8-64G	MX2020 and MX2010	<ul style="list-style-type: none"> 8-core Haswell CPU Wellsburg PCH-based Routing Engine with 64-GB DRAM and two 64-GB SSDs
RE-PTX-X8-64G	PTX5000	<ul style="list-style-type: none"> 8-core Haswell CPU Wellsburg PCH-based Routing Engine with 64-GB DRAM and two 64-GB SSDs New Control Board CB2-PTX
RCBPTX	PTX3000	<ul style="list-style-type: none"> Wellsburg PCH-based Routing Engine with 64-GB DRAM and two 64-GB SSDs Multi-core Haswell CPU <p>RCB combines the functionality of a Routing Engine, Control Board, and Centralized Clock Generator (CCG)</p>



NOTE: Platform support depends on the Junos OS release in your installation.

Related Documentation

- Supported Routing Engines by Router*

Architecture of RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines

Figure 12 on page 263 illustrates the architecture of RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines. It comprises the following components:

- The hardware layer
- The operating system and hypervisor layer.
- The host utilities and Junos VM guest layer.

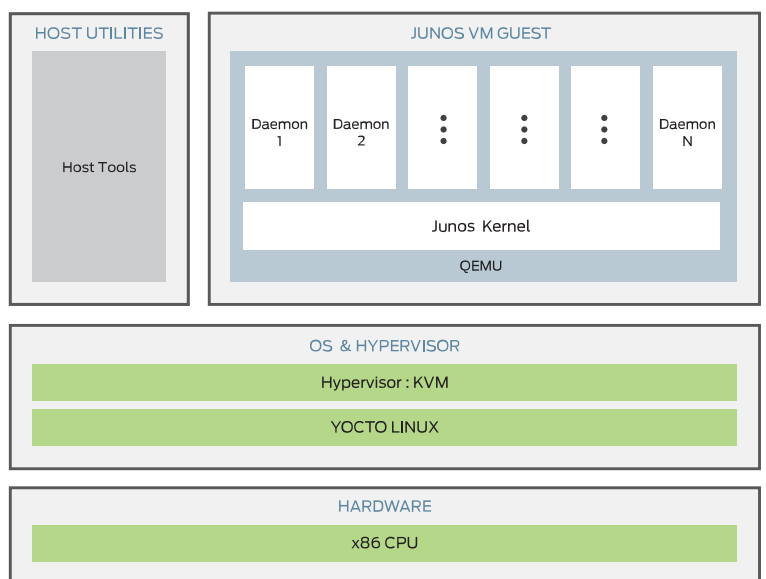
The server at the hardware layer contains the physical network interface cards (NICs), CPUs, memory, and Ethernet management port. The NICs support hardware virtualization based on single root I/O virtualization (SR-IOV). With SR-IOV, the physical NICs (known as a physical functions) are managed by the host, while the virtual functions are managed by the guest OS. Over the hardware layer, a Linux-based OS provides the host environment along with the kernel-based virtual machine (KVM) and Quick Emulator (QEMU). This host OS manages the boot complex, CPU memory storage, and various other hardware components such as the physical functions. Junos OS runs as guest OS, manages the

virtual functions, and serves as the administrative framework. Additionally, it also provides the interface for managing the host and the hypervisor.

The additional applications and utilities running on the host OS assist in providing the following functionality:

- Facilitating communication between host OS and guest OS.
- Triggering appropriate execution of the host OS based on the command and configuration on the guest Junos OS.
- Extending the VM management functionality to provide features such as autorecovery.

Figure 12: Architecture of RE-MX-X6, RE-MX-X8 and RE-PTX-X8 Routing Engines



Related Documentation • [RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX with VM Host Support on page 261](#)

Salient Features of the RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines

While continuing to provide the same end-user experience, the new architecture provides a better performing Routing Engine.

The following are the salient features of the Routing Engines:

Platform Virtualization

Platform virtualization by the introduction of a middle layer that comprises the host OS and the KVM (or the hypervisor).

- Enables support for multiple instances of Junos OS to be run concurrently.

- Enables support for third-party software to be run directly.

Hardware Assisted Paravirtualized Guest Junos OS

Provides the user with the benefits of platform virtualization along with the default performance and functionality. Paravirtualization is a virtualization technique in which a software component similar to the underlying hardware component resides in the VM and interacts with the hypervisor to execute many operations. In contrast to full virtualization, this technique reduces the overhead of virtualization in the VM.

Guest Junos OS to Serve as the Administrative Framework

The configurations, chassis control, communication with the host OS, and user interface command execution are managed by the guest Junos OS.

Storage Partitioning and Redundancy

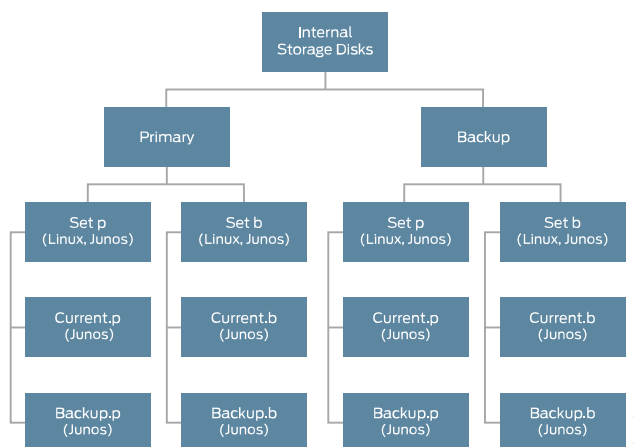
An Internal solid-state drive (SSD) is used as boot media for operating the Routing Engine. Additional options such as USB storage and network boot are available for installation and recovery purposes. A set of two 64-GB SSDs is available for normal functioning of the Routing Engine. The Routing Engine requires both the SSDs to be functional. Storage partitioning is important for debugging the Routing Engine, for new installations, and for SSD replacement.

Of the two SSDs, one SSD is the primary disk and the other SSD is the backup disk. Two sets of software boot images are available on the primary disk. You can use one set of images for booting and the other set of images for upgrade purposes. Until a software upgrade or a software rollback is performed, the BIOS is programmed to boot from the same set of images on the SSD.

Both the SSDs are partitioned to provide host boot partition, root partition, and partition for the guest image storage. The host boot partition contains the boot loader, which is the software responsible for booting the OS, Linux kernel, and RAM file system. The root partition contains the root file system for the host OS.

[Figure 13 on page 265](#) shows the partitioning of SSDs.

Figure 13: SSD Partitioning



Each SSD partition contains more than one set of fully functional host software. In case of a boot failure on the primary SSD, the router can boot by using the snapshot available on the alternate SSD. This snapshot can be generated by a fresh installation or by using the **request vmhost snapshot** command.

Related Documentation

- [request vmhost snapshot on page 532](#)
- [request vmhost reboot on page 474](#)
- [request vmhost power-off on page 527](#)
- [Upgrading the SSD Firmware on RE-S-X6-64G and RE-MX2K-X8-64G Routing Engines on page 275](#)

NTP and Time Zone

The date and time zones are synchronized from the administrative guest Junos OS to the host OS. Therefore, the timestamps in system log files of Junos OS and the host OS are synchronized.

Autorecovery

The automatic recovery (autorecovery) feature provides the following functions:

- Detecting corruptions in disk partitioning during system startup and attempting to recover partitions automatically
- Detecting corruptions in the Junos OS configuration during system startup and attempting to recover the configuration automatically, thereby ensuring that the operations and management are not disrupted.
- Detecting corruptions in Junos OS licenses during system startup and attempting to recover licenses automatically.

During the process of recovery, the host OS tries to launch the Junos VM from the image available on the primary disk. However, if the Junos VM fails to launch, the host OS

attempts to launch the Junos VM from the snapshot of the host OS image and Junos OS image available in the backup disk, provided **request vmhost snapshot** was the last operation performed. If the backup disk does not contain the snapshot, the host OS attempts to launch the Junos VM from the software available in the alternate set in the primary disk, provided **request vmhost upgrade** was the last operation performed.

The autorecovery feature is enabled by default on the guest OS. If you need to disable autorecovery—for example, to examine the failure state for debugging—use the following command:

```
user@host> set vmhost no-auto-recovery
```

- Related Documentation**
- [Disabling Autorecovery on RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines on page 277](#)

Handling Reboot and Power Off

You can reboot the Routing Engine by using the **request vmhost reboot** command. This command reboots the Routing Engine by rebooting both the guest Junos OS and the host OS. However, reboot of the Routing Engine can be triggered because of various reasons. The events or the reasons that trigger a host OS reboot are different from those that trigger a guest OS reboot.

Guest OS reboot implies that only the Junos OS is rebooted, and that the host OS is up and running. The following are a few of the reasons that trigger a guest OS reboot:

- Reboot due to panic
- VJUNOS reboot—Guest OS reboot after a shutdown.
- VJUNOS watchdog from host—Guest reboot due to emulated watchdog timer expiry

Host OS reboot implies that both the host OS and the guest OS (here, Junos OS) are rebooted. The following are a few reasons that trigger a host OS and guest OS reboot:

- Hypervisor reboot
- Power cycle or power failure
- Reboot due to exception.
- Reset-button reset—Reboot triggered by the pressing of the reset button on the front panel.
- Thermal shutdown
- Watchdog—Reboot due to PCH watchdog timer expiry

You can find the reason for the reboot by using the **show chassis routing-engine** command or the **show vmhost uptime** command.

For example:

```
host@router> show chassis routing-engine 0 | match "Last reboot reason"
Last reboot reason 0x4000:VJUNOS reboot
```

```
host@router> show vmhost uptime re0 | match "Vmhost last reboot reason"
Vmhost last reboot reason: 0x2000:hypervisor reboot
```

If the Routing Engine finishes booting and if you need to power off the router again, run the **request vmhost power-off** command. If you want the Routing Engine to reboot, use the **request vmhost reboot** command.

Related Documentation

- [request vmhost snapshot on page 532](#)
- [request vmhost reboot on page 474](#)
- [request vmhost power-off on page 527](#)
- [Disabling Autorecovery on RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines on page 277](#)

Boot Process Overview

The boot process involves configuring the basic parameters through the console port and filename synchronization.

Booting for the First Time

When you power on a router for the first time, the router initiates the boot process.

After hardware and field-programmable gate array (FPGA) level initialization is complete, the Unified Extensible Firmware Interface (UEFI) selects the boot device to launch the host OS. The host OS launches the default guest Junos OS, which is the administrative context for the user. After the router has powered on completely, a login prompt is displayed on the console port.

Boot Sequence

The Routing Engine boots from the storage media in the following sequence:

- USB
- Solid-state Drive 1 (SSD1)
- Solid-state Drive 1 (SSD2)
- Preboot Execution Environment (PXE)

Related Documentation

- [Creating an Emergency Boot Device for RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX Routing Engines on page 209](#)

Understanding Console Port

To perform the initial configuration, you need to connect a terminal or laptop computer to the router through the console port, which is a serial port on the front of the router. The console port is the management port used by administrators to log in to Junos OS directly—that is, without using a network connection.

Two universal asynchronous receiver/transmitter (UART) ports are connected to the midplane to provide CTY access to line cards. At any time, two ports can be active for the CTY application. These ports are available to the Junos VMs for configuration.

For more information about configuring the router's basic properties, see [“Accessing a Junos OS Device the First Time” on page 55](#).

Understanding Hostnames Synchronization

A hostname provides a unique identification for a router on the network. Junos OS uses the configured hostname as part of the command prompt, to prepend log files and other accounting information, as well as in other places where knowing the device identity is useful. Although Junos OS supports a maximum hostname length of 255 characters, the host OS supports hostnames that have only 64 characters or less. Therefore, hostnames need to be synchronized between Junos OS and the host OS. Keep in mind the following conditions when you synchronize the hostname configured on Junos OS with that on the host OS:

- If the Junos OS-configured hostname has less than or equal to 58 characters, then the hostname supported by the host OS (Linux) has the format *Junos hostname-node*.

For example, if the Junos OS-configured hostname is *xx.xx*, the hostname is *xx.xx-node*.

- If the Junos OS-configured hostname is greater than 58 characters in length, then the synchronization process truncates characters from the 59th character onward and replaces the truncated characters with *-node*.

Related Documentation

- [vmhost on page 361](#)
- [request vmhost reboot on page 474](#)
- [request vmhost power-off on page 527](#)
- [Creating an Emergency Boot Device for RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX Routing Engines on page 209](#)

VM Host Installation

You can install the Junos OS software package and host software package on the router. The following installation options are available:

- Fresh installation—This installation method can be used for factory installation as well as for recovery after corruption. Fresh installation can be done using Preboot Execution Environment (PXE)/NetBoot or a USB install media package. This method of installation installs the host OS, tools, and the Junos VMs.

A PXE boot is an environment to boot devices using a network interface independent of available data storage devices or installed operating systems. The PXE environment is built on a foundation of Internet protocols and services. These include TCP/IP, DHCP, and TFTP. This method of installation mostly used for installing the operating system on a device, without depending on the state of the internal media. The required software for network installation is stored on a TFTP server. PXE boot method supports remote

installation thereby overcoming the need for an in-person assistance for installation. For more information, see [“Copying VM Host Installation Package to the PXE Boot Server” on page 272](#)

You can choose to use the USB disk installation method when the router fails to reboot because of internal media failure or when there is no installed Junos OS. For more information, see [“Creating an Emergency Boot Device for RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX Routing Engines” on page 209](#).

On a fresh installation using USB, the following directories are populated with the Junos OS image on both the SSDs:

- Current.p
- Backup.p
- Backup.b
- Regular installation—This installation method is generally for an upgrade or a downgrade. This procedure can be used to install the runtime installation package on the currently running Junos VM to upgrade or downgrade relevant components. Junos VM performs the dependency check to identify the software components that require an upgrade or a downgrade to ensure compatibility.

VM Host Upgrade

Every Junos OS release is a group of files bundled together. The Routing Engines RE-MX-X6, RE-MX-X8, and RE-PTX-X8 support only the 64-bit version of Junos OS.



NOTE: If you have important files in directories other than `/config` and `/var`, copy the files to a secure location before upgrading the router. The files under `/config` and `/var` (except `/var/etc`) are preserved after the VM host upgrade.

To upgrade the software, you can use the following methods:

- Junos OS upgrade—Use the regular **junos-install-x.tgz** image upgrade. When you use this method, you must specify the regular package in the **request system software add** command. If the host software does not meet the minimum version as specified in the regular upgrade package, then you need to upgrade the host OS as well.
- Host upgrade—Use the **junos-vmhost-install-x.tgz** image upgrade. When you upgrade the host OS, you must specify the regular package in the **request vmhost software add** command. This is the recommended mode of upgrade because this method installs the host image along with the compatible Junos OS.

When the upgrade happens through a VM host package, the build components are compatible with each other. However, during a subcomponent or a package upgrade, such as a Junos OS image upgrade, you must check the installed base version and ensure the compatibility. Each relevant subpackage has the logic to fetch the VM host version and compare it with the supported version. In case of a version compatibility mismatch, a suitable warning is displayed and the installation is aborted.

The following example illustrates the upgrade operation. You can install multiple software packages and software add-on packages at the same time.

```
user@host> > request vmhost software add
/var/tmp/junos-vmhost-install-ptx-x86-64-15.1F5-S2.8.tgz
Initializing...
Verified os-libs-10-x86-64-20160616 signed by PackageProductionEc_2016
Mounting os-libs-10-x86-64-20160616.329709_builder_stable_10
....
Transfer Done
Transfer /packages/db/pkginst.13874/junos-vmhost-install*.tgz
Transfer Done
Starting upgrade ...
Preparing for upgrade...
/tmp/pkg-0mc/unpack/install/
...
...

Cmos Write successfull for Boot_retry
... upgrade complete.
A REBOOT IS REQUIRED TO LOAD THIS SOFTWARE CORRECTLY.
Use the 'request vmhost reboot' command to reboot the system
```

VM Host Rollback

You can revert to the software version that was loaded at the last successful **request vmhost software add** operation. You can roll back to the previous set of software packages, including the host OS packages, by using the **request vmhost software rollback** command.

The following example illustrates the software rollback operation. The Routing Engine that has booted from the primary disk by using the set p had booted using the set b before the upgrade.

```
user@host> show vmhost version
Current root details, Device sda, Label: jrootp_P, Partition: sda3

Current boot disk: Primary
Current root set: p
UEFI Version: NGRE_v00.53.00.01

Primary Disk, Upgrade Time: Wed Feb 24 17:51:53 UTC 2016

Version: set p
VMHost Version: 2.951
VMHost Root: vmhost-x86_64-15.1I20160210_2212_builder
VMHost Core: vmhost-core-x86_64-15.1I20160210_2212_builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F5.5

Version: set b
VMHost Version: 2.953
VMHost Root: vmhost-x86_64-15.1F520160222_1052_builder
VMHost Core: vmhost-core-x86_64-15.1F520160222_1052_builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F5.6
```

```

user@host> request vmhost software rollback
Current root details, Device sda, Label: jrootp_P, Partition: sda3
Finding alternate root for rollback
Rollback to software on jrootb_P ...
sh /etc/install/mk-mtre-rollback.sh jrootb_P b
Mounting device in preparation for rollback...
Updating boot partition for rollback...
Rollback complete, please reboot the node for it to take effect.
Cmos Write successfull
Cmos Write successfull for Boot_retry
Cmos Write successfull for Boot_retry

```

```

user@host> show vmhost version
Current root details, Device sda, Label: jrootp_P, Partition: sda3
Current boot disk: Primary
Current root set: p
UEFI Version: NGRE_v00.53.00.01

```

```

Primary Disk, Upgrade Time: Wed Feb 24 17:51:53 UTC 2016
Pending reboot.

```

```

Version: set p
VMHost Version: 2.951
VMHost Root: vmhost-x86_64-15.1I20160210_2212_builder
VMHost Core: vmhost-core-x86_64-15.1I20160210_2212_builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F5.5

```

```

Version: set b
VMHost Version: 2.953
VMHost Root: vmhost-x86_64-15.1F520160222_1052_builder
VMHost Core: vmhost-core-x86_64-15.1F520160222_1052_builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F5.6

```

```

user@host> request vmhost reboot
Reboot the vmhost ? [yes,no] (no) yes

```

```

warning: Rebooting re1
Initiating vmhost reboot... ok
Initiating Junos shutdown... shutdown: [pid 9733]
Shutdown NOW!
ok
Junos shutdown is in progress...

```

```

*** FINAL System shutdown message ***

```

```

System going down IMMEDIATELY

```

```

user@host> show vmhost version
Current root details, Device sda, Label: jrootb_P, Partition: sda4
Current boot disk: Primary
Current root set: b
UEFI Version: NGRE_v00.53.00.01

```

```

Primary Disk, Upgrade Time: Wed Feb 24 17:51:53 UTC 2016

```

```
Version: set p
VMHost Version: 2.951
VMHost Root: vmhost-x86_64-15.1I20160210_2212_builder
VMHost Core: vmhost-core-x86_64-15.1I20160210_2212_builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F5.5
```

```
Version: set b
VMHost Version: 2.953
VMHost Root: vmhost-x86_64-15.1F520160222_1052_builder
VMHost Core: vmhost-core-x86_64-15.1F520160222_1052_builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F5.6
```

VM Host Snapshot

The snapshot feature enables you to create copies of the currently running and active file system partitions on a router.

On the router, you can back up the snapshot of the host OS image along with the Junos OS image. You can use the **request vmhost snapshot** command to create a VM host recovery snapshot on the backup disk.

Disk Recovery Using the VM Host Snapshot

If the state of the primary disk (disk1) is good and the backup disk (disk2) has to be recovered then use the **request vmhost snapshot** command to recover the backup disk assuming the Routing Engine is booted from the primary disk. If the state of the secondary disk is not known or the file systems in disk are not in a consistent state, then include **partition** option in the command i.e. **request vmhost snapshot partition**.

If the state of the backup disk (disk2) is good and the primary disk (disk1) has to be recovered then use the **request vmhost snapshot recovery** command to recover the primary disk assuming the Routing Engine is booted from the backup disk. If the state of the primary disk is not known or the partition tables are in bad condition, then include **partition** option in the command i.e. **request vmhost snapshot recovery partition**.

To boot from desired disk, you can execute **request vmhost reboot {disk1,disk2}** command.

Related Documentation

- [request vmhost software add on page 535](#)
- [request vmhost software rollback on page 540](#)
- [request vmhost snapshot on page 532](#)
- [show vmhost snapshot on page 565](#)

Copying VM Host Installation Package to the PXE Boot Server

You can install the host OS, tools, and the Junos virtual machines (VMs) on the routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines by using the Preboot Execution Environment (PXE) boot method. This is one of the methods used for a fresh installation. A PXE boot prepares a client/server environment to boot devices by using

a network interface that is independent of available data storage devices or installed operating systems. The image of the operating system is stored on a TFTP server.

To copy the installation packages to the PXE boot server:

1. Copy the downloaded installation media to the **/var/tmp** directory in the PXE boot server.

```
scp
/volume/build/junos/15.1/release/15.1F3.9/ship/junos-vmhost-install-net-x86-64-15.1F3.9.tgz
user@host:/var/tmp/
```

2. Log in to the PXE boot server and verify the installation file.

```
user@host> ls -lh junos-vmhost-install-net-x86-64-15.1F3.9.tgz
-rw-r--r-- 1 root root 1.8G Oct 24 00:42
junos-vmhost-install-net-x86-64-15.1F3.9.tgz
```

3. Extract the **junos-vmhost-install-net** TAR file.

```
user@host> tar xvzf junos-vmhost-install-net-x86-64-15.1F3.9.tgz -C /var/tmp
contents/
contents/junos-vmhost-install.tgz
contents/vmhost-install-net-x86_64-15.1I20151019_1021_builder.tgz
manifest
manifest.certs
manifest.ecerts
manifest.esig
manifest.sig
package.xml
```

4. Remove the previously installed files, if any, from the **/tftpboot** directory.

```
user@host> rm -f
/tftpboot/{vmhost-version.sh,bootpxe64.efi,vmhost-version,grub.cfg,initramfs,vmlinuz}
user@host> ls -lh /tftpboot//
total 45M

rw-r--r-- 1 root root 690K Sep  8 13:22 bootpxe.efi
-rw-rw-r-- 1 930 930 45M Oct 20 01:51
vmhost-install-net-x86_64-15.1I20151019_1021_builder.tgz
```

5. Extract the network installation package.

```
user@host> tar xvzf
/var/tmp/contents/vmhost-install-net-x86_64-15.1I20151019_1021_builder.tgz -C /tftpboot/
./
./vmhost-version.sh
./bootpxe64.efi
./vmhost-version
./grub.cfg
..
...
-rw-rw-r-- 1 930 930 45M Oct 20 01:51
vmhost-install-net-x86_64-15.1I20151019_1021_builder.tgz
-rw-rw-r-- 1 930 930 6 Oct 20 01:51 vmhost-version
-rwxrwxr-x 1 930 930 416 Oct 20 01:51 vmhost-version.sh
```

```
-rw-r--r-- 1 930 930 6.9M Oct 20 01:51 vmlinuz
```

6. Rename or delete the previously installed root file system/scripts from the `/var/install` directory. Create a new `/var/install` directory.

```
user@host>mv /var/install /var/install_old
user@host>mkdir /var/install
```

7. Extract the installation package.

```
user@host>tar xvzf /var/tmp/contents/junos-vmhost-install.tgz -C /var/install
./
./vmhost-pkgs-version
./vm/
./vm/note
./vm/grub.cfg.ngre
./vm/vsmartd-1.0-0.x86_64.rpm
./vm/re_fpga-1.0-0.x86_64.rpm
./vm/veccd-1.0-0.x86_64.rpm
./vmhost-version.sh
./vmhost/
./vmhost/vmhost-x86_64-15.1I20151019_1021_builder.img.gz
...
...
./junos/junos-mtre-upgrade.sh
./vmhost-core-x86_64-15.1I20151019_1021_builder.tgz
./junos/
./junos/junos-install-x86-64-15.1F3.9.img.gz
```

8. Set permissions for the files in the `/var/install` and `/tftpboot` directories.

```
user@host> chown root:root /tftpboot/*
user@host> chmod a+rw /tftpboot/*
user@host> chown -R root:root /var/install
user@host> chmod -R a+rw /var/install
```

9. Exit the PXE boot server.

```
user@host> exit
```

Related Documentation

- [VM Host Installation on page 268](#)
- [Creating an Emergency Boot Device for RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX Routing Engines on page 209](#)

Upgrading the SSD Firmware on RE-S-X6-64G and RE-MX2K-X8-64G Routing Engines

Starting in Junos OS Release 17.2R1, you can upgrade the solid-state drive (SSD) firmware on MX Series routers with the RE-S-X6-64G and RE-MX2K-X8-64G Routing Engines. A set of two SSDs, **disk1** and **disk2**, is available for normal functioning of the Routing Engine. This topic shows how to perform the upgrade.



NOTE: You should only upgrade SSD firmware under the direction of a Juniper Networks support representative.

Before you begin check the current firmware version of the SSD.

```
user@host> show system firmware
```

Part	Type	Tag	Current version	Available version	Status
Routing Engine 0	RE BIOS	0	0.45	0.53	OK
Routing Engine 0	RE FPGA	1	36.0.0	41.0	OK
Routing Engine 0	RE SSD1	4	12028	12029	OK
Routing Engine 0	RE SSD2	5	12028	12029	OK
Routing Engine 1		0	1.4		OK

If the value of **Current version** is less than the value of **Available version** then the following procedure can be used for the SSD firmware upgrade.

To upgrade SSD firmware:

1. Copy the jfirmware package to the device.

If the file has been obtained from JTAC, use FTP or SCP to load the firmware file on the device. Save the file in the **/var/tmp** directory.

```
user@host> request system software add
ftp://ftp.juniper.net/private/system/jfirmware-17.1R2-signed.tgz
```

2. Upgrade the SSD disk1 firmware.



NOTE: You should upgrade the SSD on a master Routing Engine only. For upgrading firmware on the backup Routing Engine, switch mastership using the following command and then log in to the backup Routing Engine.

```
user@host> request chassis routing-engine master switch
```

To initiate the upgrade, use the following command:

```
user@host> request system firmware upgrade re ssd disk1
```

Part	Type	Tag	Current version	Available version	Status
Routing Engine 0	RE SSD1	4	12028	12029	OK

Perform indicated firmware upgrade ? [yes,no] (no) yes

Firmware upgrade initiated, use "show system firmware" to monitor status.

Then, monitor the upgrade status using the **show system firmware** command.

```
user@host> show system firmware
```

Part	Type	Tag	Current version	Available version	Status
Routing Engine 0	RE BIOS	0	0.45	0.53	OK
Routing Engine 0	RE FPGA	1	36.0.0	41.0	OK
Routing Engine 0	RE SSD1	4	12028	12029	OK
Routing Engine 0	RE SSD2	5	12028	12029	OK
Routing Engine 1		0	1.4		OK

```
user@host> show system firmware
```

Part	Type	Tag	Current version	Available version	Status
Routing Engine 0	RE BIOS	0	0.45	0.53	OK
Routing Engine 0	RE FPGA	1	36.0.0	41.0	OK
Routing Engine 0	RE SSD1	4	12029	12029	UPGRADED SUCCESSFULLY
Routing Engine 0	RE SSD2	5	12028	12029	OK
Routing Engine 1		0	1.4		OK

After a successful upgrade, confirm that the **Current version** and **Available version** of the SSD firmware are identical.

3. Upgrade SSD Disk2 firmware.

To initiate the upgrade, use the following command:

```
user@host> request system firmware upgrade re ssd disk2
```

Part	Type	Tag	Current version	Available version	Status
Routing Engine 0	RE SSD2	5	12028	12029	OK

Perform indicated firmware upgrade ? [yes,no] (no) yes

Firmware upgrade initiated, use "show system firmware" to monitor status.

Monitor the upgrade status using the **show system firmware** command.

```
user@host> show system firmware
```

Part	Type	Tag	Current version	Available version	Status
Routing Engine 0	RE BIOS	0	0.45	0.53	OK
Routing Engine 0	RE FPGA	1	36.0.0	41.0	OK
Routing Engine 0	RE SSD1	4	12028	12029	UPGRADED SUCCESSFULLY
Routing Engine 0	RE SSD2	5	12028	12029	PROGRAMMING
Routing Engine 1		0	1.4		OK

```
user@host> show system firmware
```

Part	Type	Tag	Current version	Available version	Status
Routing Engine 0	RE BIOS	0	0.45	0.53	OK

Routing Engine	0	RE FPGA	1	36.0.0	41.0	OK	
Routing Engine	0	RE SSD1	4	12029	12029	UPGRADED	SUCCESSFULLY
Routing Engine	0	RE SSD2	5	12029	12029	UPGRADED	SUCCESSFULLY
Routing Engine	1		0	1.4		OK	

After a successful upgrade, confirm that the **Current version** and **Available version** of the SSD firmware are identical.

Release History Table

Release	Description
17.2R1	Starting in Junos OS Release 17.2R1, you can upgrade the solid-state drive (SSD) firmware on MX Series routers with the RE-S-X6-64G and RE-MX2K-X8-64G Routing Engines.

Disabling Autorecovery on RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines

The autorecovery feature helps recover the Junos OS automatically in the event of a corruption, thereby ensuring that the Junos OS is available for operations and management always. The host Junos OS tries to launch the Junos VM from the image available on the primary disk. However, if the guest Junos OS fails to launch, the host OS attempts to launch the Junos VM from the snapshot of the host OS image and Junos OS image available in the backup disk, provided **request vmhost snapshot** was the last operation performed. If the backup disk does not contain the snapshot, the host OS attempts to launch the Junos VM from the software available in the alternate set in the primary disk, provided **request vmhost upgrade** was the last operation performed.

The autorecovery feature is enabled by default on the guest Junos OS. For debugging purposes, if you do not want the host to recover the Junos VM automatically, you can disable the auto-recovery by the host.

To disable the guest auto-recovery, include the **no-auto-recovery** statement at the **[edit vmhost]** hierarchy level:

```
[edit vmhost]
no-auto-recovery
```

Related Documentation

- [vmhost on page 361](#)

VM Host Operations and Management

With the virtualization of the Routing Engine, Junos OS supports new **request** and **show** commands associated with the host and hypervisor processes. The commands are related to:

- Reboot, halt, and power management for the host.
- Software upgrade for the host.

- Disk snapshot for the host.

The following **request** commands are not available on the RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines:

- **request system halt**
- **request system partition abort**
- **request system power-off**
- **request system power on**

The following commands can be used only for the guest Junos OS:

- [request system reboot](#)
- [request system snapshot](#)
- [request system software add](#)
- [request system zeroize](#)

You can use the following new **request vmhost** commands on the host OS:

- [request vmhost cleanup](#)
- [request vmhost file-copy](#)
- [request vmhost halt](#)
- [request vmhost hard-disk-test](#)
- [request vmhost power-off](#)
- [request vmhost power-on](#)
- [request vmhost reboot](#)
- [request vmhost snapshot](#)
- [request vmhost software abort](#)
- [request vmhost software add](#)
- [request vmhost software in-service-upgrade](#)
- [request vmhost software rollback](#)
- [request vmhost zeroize](#)

**Related
Documentation**

- [RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX with VM Host Support on page 261](#)

PART 3

Installing and Managing Software Licenses

- [Software License Overview on page 281](#)
- [Installing and Managing Licenses on page 317](#)

Software License Overview

- [Junos OS Feature Licenses on page 281](#)
- [License Enforcement on page 282](#)
- [Junos OS Feature License Keys on page 283](#)
- [Software Feature Licenses on page 286](#)
- [Software Features That Require Licenses on MX Series Routers Only on page 301](#)
- [Understanding Software Licenses for EX Series Switches on page 307](#)

Junos OS Feature Licenses

Some Junos OS software features require a license to activate the feature. To enable a licensed feature, you need to purchase, install, manage, and verify a license key that corresponds to each licensed feature. To conform to Junos OS feature licensing requirements, you must purchase one license per feature per device. The presence of the appropriate software license key on your device determines whether you are eligible to configure and use the licensed feature.

To speed deployment of licensed features, Junos OS software implements an honor-based licensing structure and provides you with a 30-day grace period to use a licensed feature without a license key installed. The grace period begins when you configure the feature and your device uses the licensed feature for the first time, but not necessarily when you install the license. After the grace period expires, the system generates system log messages saying that the feature requires a license. To clear the error message and use the licensed feature properly, you must install and verify the required license.

Data center customers, for example those using the QFX platform, use universal licenses. Starting in Junos OS Release 15.1, to ensure that license keys are used properly, Juniper Networks license key generation is enhanced to specify a customer ID in the license key. You can see the customer ID displayed in the output of the **show system license** command.

For information about how to purchase software licenses, contact your Juniper Networks sales representative.

Release History Table

Release	Description
15.1	Starting in Junos OS Release 15.1, to ensure that license keys are used properly, Juniper Networks license key generation is enhanced to specify a customer ID in the license key.

Related Documentation

- [License Enforcement on page 282](#)
- [Junos OS Feature License Keys on page 283](#)
- [Software Feature Licenses on page 286](#)
- [Verifying Junos OS License Installation \(CLI\) on page 324](#)
- [show system license on page 492](#)

License Enforcement

For features or scaling levels that require a license, you must install and properly configure the license to meet the requirements for using the licensable feature or scale level. The device enables you to commit a configuration that specifies a licensable feature or scale without a license for a 30-day grace period. The grace period is a short-term grant that enables you to start using features in the pack or scale up to the system limits (regardless of the license key limit) without a license key installed. The grace period begins when the licensable feature or scaling level is actually used by the device (not when it is first committed). In other words, you can commit licensable features or scaling limits to the device configuration, but the grace period does not begin until the device uses the licensable feature or exceeds a licensable scaling level.



NOTE: Configurations might include both licensed and nonlicensed features. For these situations, the license is enforced up to the point where the license can be clearly distinguished. For example, an authentication-order configuration is shared by both Authentication, Authorization, and Accounting (AAA), which is licensed, and by Layer 2 Tunneling Protocol (L2TP), which is not licensed. When the configuration is committed, the device does not issue any license warnings, because it is not yet known whether AAA or L2TP is using the configuration. However, at runtime, the device checks for a license when AAA authenticates clients, but does not check when L2TP authenticates clients.

The device reports any license breach as a warning log message whenever a configuration is committed that contains a feature or scale limit usage that requires a license. Following the 30-day grace period, the device periodically reports the breach to syslog messages until a license is installed and properly configured on the device to resolve the breach.



NOTE: Successful commitment of a licensable feature or scaling configuration does not imply that the required licenses are installed or not required. If a required license is not present, the system issues a warning message after it commits the configuration.

Related Documentation

- [Junos OS Feature License Keys](#)
- [Software Feature Licenses on page 286](#)
- [Adding New Licenses \(CLI Procedure\) on page 317](#)
- [Deleting License Keys \(CLI\) on page 322](#)
- [Saving License Keys \(CLI\) on page 324](#)
- [Verifying Junos OS License Installation \(CLI\) on page 324](#)

Junos OS Feature License Keys

Some Junos OS software features require a license to be activated. To enable each licensed feature, you must purchase, install, manage, and verify a license key that corresponds to the licensed feature.

Release-Tied License Keys and Upgrade Licenses on MX Series Routers

The Junos OS licensing infrastructure currently associates a license feature with attributes such as date, platform, and validity. In addition to these attributes, for MX Series routers running Junos OS Release 12.2 and later, a licensed feature can be associated with a release number at the time of generating the license key. This type of release-tied license key is used to validate a particular licensed feature while attempting a software upgrade. The upgrade process aborts if the release number in the license key is earlier than the Junos OS release number to which the system is being upgraded.

Additionally, an upgrade license key can be generated for a release-tied licensed feature. An upgrade license key is used for carrying forward a capacity license to the upgrade release. Although an upgrade license might be an acceptable license on the current release, it does not add to the existing capacity limit. The capacity added in the upgrade license key is valid for the upgrade software release only.

The release number embedded in the license key indicates the maximum release number up to which Junos OS can be upgraded.

As an example, assume that your system is running Junos OS Release 12.2 and is using the **scale-subscriber** licensed feature with a later release-tied upgrade license key installed. If you request a software upgrade to the later release of Junos OS, the software upgrade operation fails and the following error message is displayed:

```
mgd: error: No valid upgrade license found for feature 'scale-subscriber'.
Aborting Software upgrade.
Validation failed
```

In this example, to successfully upgrade to the later release of Junos OS, the release number included in the upgrade license key should be greater than or equal to the later release number. Also, you can perform software upgrades up to the previous release without any additional license keys to retain the existing scale limit.



NOTE:

When you install a release-tied license, the following apply:

- You can purchase an upgrade capacity license only if a base capacity license for the same scale-tier has already been generated or purchased.
 - You cannot install an upgrade license if the capacity does not match any of the existing base capacity licenses on the system.
 - The license installation fails when you install a lower release number license key on a higher software release number.
 - A release-tied license can be installed on a Junos OS release number that is lower than or equal to the release number included in the license key. For example, a 12.2 license key is valid on Junos OS Release 12.1.
 - An upgrade license is valid only on the target release number specified in the license key, but can be installed on an earlier Junos OS release. For example, a 4 K scale-tier upgrade license for Junos OS Release 12.2 can be installed on an earlier release, and the installed count of licenses remains unaltered.
 - Release-tied licenses of the previous release are not deleted on upgrading Junos OS to a newer release version.
-

Licensable Ports on MX5, MX10, and MX40 Routers

Starting with Junos OS Release 12.2, license keys are available to enhance the port capacity on MX5, MX10, and MX40 routers up to the port capacity of an MX80 router. The MX5, MX10, and MX40 routers are derived from the modular MX80 chassis with similar slot and port assignments, and provide all functionality available on an MX80 router, but at a lower capacity. Restricting port capacity is achieved by making a set of MIC slots and ports licensable. MICs without a license are locked, and are unlocked or made usable by installing appropriate upgrade licenses.

The base capacity of a router is identified by the Ideeprom assembly ID (I2C ID), which defines the board type. However, the Junos OS licensing infrastructure allows the use of restricted ports without a license for a grace period of 30 days. After the grace period expires, the router reverts back to the base capacity if no upgrade license is purchased and installed for the locked ports. The I2C ID along with an upgrade license determine the final capacity of an MX5, MX10, or MX40 router.

The MX5, MX10, MX40, and MX80 routers support the following types of MICs:

- A built-in 10-Gigabit Ethernet MIC with four 10-Gigabit Ethernet ports
- Two front-pluggable MICs

A feature ID is assigned to every license upgrade for enhancing port capacity.

[Table 30 on page 285](#) displays the chassis types and their associated port capacity, I2C ID, base capacity, feature ID, feature name, and the final capacity after a license upgrade.

Table 30: Upgrade Licenses for Enhancing Port Capacity

Chassis Type	Port Capacity	I2C ID	Base Capacity	Feature ID and Feature Name	Upgrade Capacity
MX5	20G	0x556	Slot 1 <ul style="list-style-type: none"> • 1/MIC0 	f1—MX5 to MX10 upgrade	Slot 1 and 2 <ul style="list-style-type: none"> • 1/MIC0 • 1/MIC1
MX10	40G	0x555	Slot 1 and 2 <ul style="list-style-type: none"> • 1/MIC0 • 1/MIC1 	f2—MX10 to MX40 upgrade	Slot 2 and first 2 ports on Slot 0 <ul style="list-style-type: none"> • 1/MIC1 • First 2 ports on 0/MIC0
MX40	60G	0x554	Slot 1, Slot 2 and first 2 ports on Slot 0 <ul style="list-style-type: none"> • 1/MIC0 • 1/MIC1 • First 2 ports on 0/MIC0 	f3—MX40 to MX80 upgrade	Slot 2 and all ports on Slot 0 <ul style="list-style-type: none"> • 1/MIC1 • All 4 ports on 0/MIC0

When installing an upgrade license for enhancing port capacity on MX5, MX10 and MX40 routers, consider the following:

- To upgrade an MX5 router to MX80 router capacity, licenses for all three features (f1, f2, f3) must be installed. All three features can be provided in a single license key.
- To upgrade an MX10 router to MX40 router capacity, installing a license key with f2 feature is sufficient.
- Non-applicable feature IDs in a license key reject the upgrade license. For example:
 - An f1 feature ID on an MX10 upgrade license key rejects the license.
 - Feature IDs f1 and f2 on an MX40 upgrade license key reject the entire license.

Port Activation on MX104 Routers

Starting with Junos OS Release 13.3, license keys are available to activate the ports on the MX104 router. MX104 routers have four built-in ports. By default, in the absence of valid licenses, all four built-in ports are deactivated. By installing licenses, you can activate any two of the four or all of the four built-in ports. For instance, you can install a license to activate the first two built-in ports (xe-2/0/0 and xe-2/0/1) or you can install a license to activate the next two built-in ports (xe-2/0/2 and xe-2/0/3). You can also install a license to activate all four built-in ports (xe-2/0/0, xe-2/0/1, xe-2/0/2, and xe-2/0/3). If you have already activated two of the built-in ports, you can install an additional license to activate the other two built-in ports on the MX104 router.

A feature ID is assigned to every license for activating the built-in ports on the MX104 router. The port license model with the feature ID is described in [Table 31 on page 286](#).

Table 31: Port Activation License Model for MX104 Routers

Feature ID	Feature Name	Functionality
F1	MX104 2X10G Port Activate (0 and 1)	Ability to activate first two built-in ports (xe-2/0/0 and xe-2/0/1)
F2	MX104 2X10G Port Activate (2 and 3)	Ability to activate next two built-in ports (xe-2/0/2 and xe-2/0/3)

Both the features are also provided in a single license key for ease of use. To activate all four ports, you must either install the licenses for both the features listed in [Table 31 on page 286](#) or the single license key for both features. If you install the single license key when feature IDs F1 and F2 are already installed, the license does not get rejected. Also, MX104 routers do not support the graceful license expiry policy. A graceful license expiry policy allows the use of a feature for a certain period of time (usually a grace period of 30 days), and reverts if the license for that feature is not installed after the grace period.

Related Documentation

- [Junos OS Feature Licenses on page 281](#)
- [License Enforcement on page 282](#)
- [Software Feature Licenses on page 286](#)
- [Verifying Junos OS License Installation \(CLI\) on page 324](#)
- [show system license on page 492](#)

Software Feature Licenses

Each license is tied to one software feature pack, and that license is valid for only one device.



NOTE: This is not a complete list of licenses. Contact your Juniper Networks representative for license information.

For information about how to purchase software licenses, contact your Juniper Networks sales representative at <http://www.juniper.net/in/en/contact-us/>.

- [Software Features That Require Licenses on M Series, MX Series, and T Series Routers on page 287](#)
- [Software Features That Require Licenses on M Series Routers Only on page 290](#)
- [Software Features That Require Licenses on MX Series Routers Only on page 291](#)
- [Software Feature Licenses for SRX Series Devices on page 297](#)

- [Software Features That Require Licenses on EX Series Switches on page 297](#)
- [Software Features That Require Licenses on the QFX Series on page 298](#)

Software Features That Require Licenses on M Series, MX Series, and T Series Routers

Table 32 on page 287 lists the licenses you can purchase for each M Series, MX Series, and T Series software feature. Each license allows you to run the specified software feature on a single device.



NOTE: The DHCP server functionality for Junos OS is part of the subscriber management feature. You must have the S-SA-FP, S-MX80-SA-FP or S-MX104-SA-FP license in order to enable the DHCP server. For service accounting, you must also have S-SSM-FP.

For information about how to purchase a software license, contact your Juniper Networks sales representative at <http://www.juniper.net/in/en/contact-us/>.

Table 32: Junos OS Feature License Model Number for M Series, MX Series, and T Series Routers

Licensed Software Feature	Supported Devices	Model Number
Generalized Multiprotocol Label Switching (GMPLS) Support on Junos OS	M10i, M7i, M120, M160, M20, M320, M40e, T320, T640, and MX Series Routers	JS-GMPLS
IPv6 Support on Junos OS	M120, M160, M20, M320, M40e, T320, T640, and MX Series Routers	JS-IPv6
Logical Router Support for Junos OS	M10i, M120, M160, M20, M320, M40e, M7i, T320, T640, and MX Series Routers	JS-LR
J-Flow accounting license for Adaptive Services (AS) PIC and Multiservices PIC	M10i, M120, M160, M20, M320, M40e, M7i, T320, M10, M5, T640, and T1600	S-ACCT
Chassis license for Application Traffic Optimization service, policy enforcement and application statistics. This license includes S-AI and S-LDPF functionality and 1-year Signature Subscription License	MX104, MX240, MX480, MX960, M Series, and T Series Routers	S-ATO
Software License for Passive Monitoring Flow Collector Application, supporting 100 Kpps throughput; Chassis based license for Multiservices PIC	M320, T640, T320, T1600	S-COLLECTOR-100K
License to use Compressed Real-Time Transport Protocol (CRTP) feature in AS PIC and Multiservices PIC	M10i, M120, M160, M20, M320, M40e, M7i, T320, M10, M5, T640, and T1600	S-CRTP
Software License for Passive Monitoring DFC Application, supporting 100Kpps throughput; Chassis based license for Multiservices PIC	M320, T640, T320, and T1600	S-DFC-100K

Table 32: Junos OS Feature License Model Number for M Series, MX Series, and T Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Security Services license for AS PIC and Multiservices PIC	M10i, M7i, M5, M120, M160, M20, M320, M40e, T320, T640, M10, and T1600	S-ES
Chassis license for IDP service, policy enforcement. This license includes S-AI and S-LDPF functionality and 1-year Signature Subscription License	MX104, MX240, MX480, MX960, M Series, and T Series Routers	S-IDP
Junos-FIPS Software License	M10i, M7i, M320, M40e, T320, and T640	S-JUNOS-FIPS
Link Services Software License—up to 1023 ML bundles per Chassis for Multiservices PIC and Multiservices Dense Port Concentrator (DPC)	M5, M7i, M10, M10i, M20, M40e, M120, M320, T320, T640, T1600, MX240, MX480, and MX960	S-LSSL-1023
Link Services Software Upgrade License—from 255 to 1023 ML bundles per Chassis for Multiservices PIC and Multiservices DPC	M5, M7i, M10, M10i, M20, M40e, M120, M320, T320, T640, T1600, MX240, MX480, and MX960	S-LSSL-1023-UPG
Link Services Software Upgrade License—from 64 to 255 ML bundles per Chassis for AS PIC, Multiservices PIC, and Multiservices DPC	M5, M7i, M10, M10i, M20, M40e, M120, M320, T320, T640, T1600, MX240, MX480, and MX960	S-LSSL-255-UPG
Link Services Software License—up to 255 ML bundles per Chassis for AS PIC, Multiservices PIC, and Multiservices DPC	M10, M7i, M5, M120, M20, M320, M40e, T320, T640, M10i, T1600, MX240, MX480, and MX960	S-LSSL-256
Link Services Software License—up to 4 ML bundles per Chassis for AS PIC, Multiservices PIC, and Multiservices DPC	M10i, M120, M20, M320, M40e, M7i, T320, M10, M5, T640, T1600, MX240, MX480, and MX960	S-LSSL-4
Link Services Software License—up to 64 ML bundles per Chassis for AS PIC, MS PIC and MS DPC	M10, M7i, M5, M120, M20, M320, M40e, T320, T640, M10i, T1600, MX240, MX480, and MX960	S-LSSL-64
Link Services Software Upgrade License—from 4 to 64 ML bundles per Chassis for AS PIC, Multiservices PIC, and Multiservices DPC	M5, M7i, M10, M10i, M20, M40e, M120, M320, T320, T640, T1600, MX240, MX480, and MX960	S-LSSL-64-UPG
Software License for Passive Monitoring Flow Monitor Application, supporting 1M flows. Chassis based license for Multiservices PIC	M320, T640, T320, and T1600	S-MONITOR-1M
Network Address Translation (NAT), FW license on AS PIC and Multiservices PIC: Multi-instance	M10, M7i, M5, M120, M160, M20, M320, M40e, T320, T640, M10i, and T1600	S-NAT-FW-MULTI
NAT, FW license on AS PIC and Multiservices PIC: Single-instance	M10, M7i, M5, M120, M160, M20, M320, M40e, T320, T640, M10i, and T1600	S-NAT-FW-SINGLE

Table 32: Junos OS Feature License Model Number for M Series, MX Series, and T Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Software license for Packet trigger subscriber policy	MX240, MX480, MX960, M120, and M320	S-PTSP
Subscriber Access Feature Pack License Scaling (128000)	MX104, MX240, MX480, MX960, M120, and M320	S-SA-128K
Subscriber Access Feature Pack License Scaling (32000)	MX104, MX240, MX480, MX960, M120, and M320	S-SA-32K
Subscriber Access Feature Pack License Scaling (4000)	MX104, MX240, MX480, MX960, M120, M320, and MX80	S-SA-4K
Subscriber Access Feature Pack License Scaling (64000)	MX104, MX240, MX480, MX960, M120, and M320	S-SA-64K
Subscriber Access Feature Pack License Scaling (8000)	MX104, MX240, MX480, MX960, M120, M320, and MX80	S-SA-8K
Subscriber Access Feature Pack License Scaling (96000)	MX104, MX240, MX480, MX960, M120, and M320	S-SA-96K
Subscriber Access Feature Pack license	MX104, MX240, MX480, MX960, M120, and M320	S-SA-FP
Stateful Failover for Services on AS PIC and Multiservices PIC: Multilink PPP (MLPPP) only	M10, M7i, M5, M120, M160, M20, M320, M40e, T320, T640, M10i, and T1600	S-SERVICES-SFO
Subscriber Service Management Feature Packet License (RADIUS/SRC based Service Activation and Deactivation) Per-Service Accounting Features for Subscribers	MX104, MX240, MX480, MX960, M120, and M320	S-SSM-FP
Subscriber Traffic Lawful Intercept Feature Pack License	MX240, MX480, MX960, M120, M320, and MX80	S-SSP-FP
Software license for application aware traffic direct feature	MX240, MX480, MX960, M120, and M320	S-TFDIRECT-APP
Software license for subscriber aware traffic direct feature	MX240, MX480, MX960, M120, and M320	S-TFDIRECT-SUB
Video Services Feature Pack license	M120, M320, MX80, MX104, MX240, MX480, and MX960	S-VIDEO-FP
Port capacity enhancement Feature Pack License for MX5 routers	MX5	mx5-to-mx10-upgrade
Port capacity enhancement Feature Pack License for MX10 routers	MX10	mx10-to-mx40-upgrade

Table 32: Junos OS Feature License Model Number for M Series, MX Series, and T Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Port capacity enhancement Feature Pack License for MX40 routers	MX40	mx40-to-mx80-upgrade

Software Features That Require Licenses on M Series Routers Only

Table 33 on page 290 lists the licenses you can purchase for each M Series software feature. Each license allows you to run the specified software feature on a single device.

For information about how to purchase a software license, contact your Juniper Networks sales representative at <http://www.juniper.net/in/en/contact-us/>.

Table 33: Junos OS Feature License Model Number for M Series Routers

Licensed Software Feature	Supported Devices	Model Number
J-Flow accounting license on Integrated Adaptive Services Module (ASM) and Integrated Multiservices Module	M7i	S-ACCT-BB
Security Services license on ASM and Integrated Multiservices Module	M7i	S-ES-BB
Layer 2 Tunneling Protocol (L2TP) L2TP Network Server (LNS) license for 16000 sessions on Multiservices PIC	M120	S-LNS-16K
L2TP LNS license Upgrade—from 8000 to 16000 sessions on Multiservices PIC	M120	S-LNS-16K-UPG
L2TP LNS license for 2000 sessions on AS PIC or Integrated Adaptive Services Module and Multiservices PIC	M7i, M10i, and M120	S-LNS-2K
L2TP LNS license for 4000 sessions on AS PIC or Integrated Adaptive Services Module and Multiservices PIC	M7i, M10i, and M120	S-LNS-4K
L2TP LNS license Upgrade—from 2000 to 4000 sessions on AS PIC or Integrated Adaptive Services Module and Multiservices PIC	M7i, M10i, and M120	S-LNS-4K-UPG
L2TP LNS license for 8000 sessions on Multiservices PIC	M7i, M10i, and M120	S-LNS-8K
L2TP LNS license Upgrade—from 4000 to 8000 sessions on AS PIC and Multiservices PIC	M7i, M10i, and M120	S-LNS-8K-UPG
Link services software license on integrated ASM and Integrated Multi Services Module—up to 4 ML bundles	M7i	S-LSSL-BB
NAT, FW license on Integrated ASM and Integrated Multi Services Module: Multi instance	M7i	S-NAT-FW-MULTI-BB
NAT, FW license on Integrated ASM and Integrated Multi Services Module: Single instance	M7i	S-NAT-FW-SINGLE-BB

Table 33: Junos OS Feature License Model Number for M Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Tunnel services software license for AS PIC and Multiservices PIC (chassis license)	M7i and M10i	S-TUNNEL

Software Features That Require Licenses on MX Series Routers Only

Table 34 on page 291 lists the licenses you can purchase for each MX Series software feature. Each license allows you to run the specified software feature on a single device.



NOTE: The DHCP server functionality for Junos OS is part of the subscriber management feature. You must have the S-SA-FP, S-MX80-SA-FP or S-MX104-SA-FP license in order to enable the DHCP server. For service accounting, you must also have S-SSM-FP.



NOTE: This is not a complete list of licenses. Contact your Juniper Networks representative for license information.

For information about how to purchase a software license, contact your Juniper Networks sales representative at <http://www.juniper.net/in/en/contact-us/>.

Table 34: Junos OS Feature License Model Number for MX Series Routers

Licensed Software Feature	Supported Devices	Model Number
Upgrade license—from MX80-10G-ADV to MX80-40G-ADV	MX80	MX80-10G40G-UPG-ADV-B
Upgrade license—from MX80-10G to MX80-40G	MX80	MX80-10G40G-UPG-B
Upgrade license—from MX80-40G-ADV to full MX80	MX80	MX80-40G-UPG-ADV-B
Upgrade license—from MX80-40G to full MX80	MX80	MX80-40G-UPG-B
Upgrade license—from MX80-5G-ADV to MX80-10G-ADV	MX80	MX80-5G10G-UPG-ADV-B
Upgrade license—from MX80-5G to MX80-10G	MX80	MX80-5G10G-UPG-B
Upgrade license to activate 2x10GE P2&3	MX104	S-MX104-ADD-2X10GE

Table 34: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Upgrade license to activate 2X10GE PO&I	MX104	S-MX104-UPG-2X10GE
Upgrade license to activate 4X10GE fixed ports on MX104	MX104	S-MX104-UPG-4X10GE
License to support per VLAN queuing on MX80	MX80	S-MX80-Q
License to support per VLAN queuing on MX104	MX104	S-MX104-Q
Chassis-based software license for inline J-Flow monitoring on MX5, MX10, M40, MX80, and MX104 Series routers	MX5, MX10, M40, MX80, and MX104	S-JFLOW-CH-MX5-104
Chassis-based software license for inline J-Flow monitoring on MX240 routers	MX240	S-JFLOW-CH-MX240
Chassis-based software license for inline J-Flow monitoring on MX480 routers	MX480	S-JFLOW-CH-MX480
Chassis-based software license for inline J-Flow monitoring on MX960 routers	MX960	S-JFLOW-CH-MX960
Chassis-based software license for inline J-Flow monitoring on MX2008 routers	MX2008	S-JFLOW-CH-MX2008
Chassis-based software license for inline J-Flow monitoring on MX2010 routers	MX2010	S-JFLOW-CH-MX2010
Chassis-based software license for inline J-Flow monitoring on MX2020 routers	MX2020	S-JFLOW-CH-MX2020
Flow monitoring and accounting features using J-Flow service on any Modular Port Concentrator (MPC) or MS-DPC	MX240, MX480, and MX960	S-ACCT-JFLOW-CHASSIS
Software License for in-line J-Flow service on Trio MPCs	MX240, MX480, and MX960	S-ACCT-JFLOW-IN

Table 34: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Flow monitoring and accounting features using J-Flow service on any MPC limited to 10G of total JFLOW traffic	MX80	S-ACCT-JFLOW-IN-10G
Flow monitoring and accounting features using J-Flow service on any MPC limited to 10G of total JFLOW traffic	MX80	S-ACCT-JFLOW-IN-10G-UPG
Flow monitoring and accounting features using J-Flow service on any MPC limited to 5G of total JFLOW traffic	MX80	S-ACCT-JFLOW-IN-5G
Security services (IPsec, VPN and group VPN) license based on a single NPU for MS-MIC, MS-DPC or MS-MPC	MX Series router	S-ES-NPU
2000 IKE sessions on MS-DPC; Chassis based, limited to 6000 per Chassis	MX240, MX480, and MX960	S-ES-2K
4000 IKE sessions on MS-DPC; Chassis based, limited to 6000 per Chassis	MX240, MX480, and MX960	S-ES-4K
Upgrade from 2000 IKE sessions to 4000 IKE sessions on MS-DPC; Chassis based, limited to 6000 per Chassis	MX240, MX480, and MX960	S-ES-4K-UPG
6000 IKE sessions on MS-DPC; Chassis based, limited to 6000 per Chassis	MX240, MX480, and MX960	S-ES-6K
Upgrade from 4000 IKE sessions to 6000 IKE Sessions on MS-DPC; Chassis based, limited to 6000 per Chassis	MX240, MX480, and MX960	S-ES-6K-UPG
License to run stateful firewall on one NPU per MS-MIC, MS-DPC or MS-MPC	MX Series routers	S-FW-NPU
License to support DS3 Channelization (down to DS0) on each Modular Interface Card (MIC) for MIC-3D-8DS3-E3; also requires license S-MX80-Q when used on the MX80 platform	MX80, MX104, MX240, MX480, and MX960	S-MIC-3D-8CHDS3
License to support full-scale Layer 3 routes and Layer 3 VPN	MX80	S-MX80-ADV-R
License to support 256K routes	MX104	S-MX104-ADV-R1

Table 34: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
License to support scaling Layer 3 and VPN routes to 1 million or more entries on MX104 platforms	MX104	S-MX104-ADV-R2
License to support full-scale Layer 3 routes and Layer 3 VPN on each slot for MPC-3D-16XGE-SFPP	MX240, MX480, and MX960	S-MPC-3D-16XGE-ADV-R
License to support full-scale Layer 3 routes and Layer 3 VPN on each slot for port queuing MPCs	MX240, MX480, and MX960	S-MPC-3D-PQ-ADV-R
License to support full-scale Layer 3 routes and Layer 3 VPN on each slot for hierarchical quality of service (HQoS) MPCs	MX240, MX480, and MX960	S-MPC-3D-VQ-ADV-R
Subscriber Management Feature Pack License for MX80	MX80	S-MX80-SA-FP
Subscriber Management Feature Pack for MX104 series	MX104	S-MX104-SA-FP
Subscriber Service Management Feature Packet License—RADIUS and SRC-based service activation and deactivation per-service accounting features	MX80	S-MX80-SSM-FP
Subscriber Service Management Feature Packet License	MX104	S-MX104-SSM-FP
Upgrade to Traffic Direct Advanced (per MS-DPC)	MX960	S-MX-TD-UPG
License to run one instance of the NAT software on one NPU per MS-DPC	MX240, MX480, and MX960	S-NAT
License to support inline NAT software on MX5, MX10, MX40, MX80, MX104	MX5, MX10, MX40, MX80, and MX104	S-NAT-IN-MX5-104 (Replaces S-NAT-IN-MX40-MX80 and S-NAT-IN-MX5-MX10)
License to run one instance of the NAT software on one NPU per MS-MIC, MS-DPC, or MS-MPC	MX80, MX104, MX240, MX480, MX960, MX2010, and MX2020	S-NAT-NPU (Replaces S-NAT-IN-MX40-MX80-UPG)
License to run NAT using any MPC in an MX Chassis	MX240, MX480, and MX960	S-NAT-IN-MX-CHASSIS
Subscriber Access Feature Pack License Scaling (4000)	MX240, MX480, MX960, M120, M320, and MX80	S-SA-4K

Table 34: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Subscriber Access Feature Pack License Scaling (8000)	MX240, MX480, MX960, M120, M320, and MX80	S-SA-8K
Subscriber Access Feature Pack License Scaling (16,000)	MX240, MX480, MX960, and MX80	S-SA-16K
Subscriber Access Feature Pack License Scaling (32,000)	MX240, MX480, MX960, M120, and M320	S-SA-32K
Subscriber Access Feature Pack License Scaling (64,000)	MX240, MX480, MX960, M120, and M320	S-SA-64K
Subscriber Access Feature Pack License Scaling (96,000)	MX240, MX480, MX960, M120, and M320	S-SA-96K
Subscriber Access Feature Pack License Scaling (128,000)	MX240, MX480, MX960, M120, and M320	S-SA-128K
Subscriber Access Feature Pack License Scaling (256,000)	MX240, MX480, and MX960	S-SA-256K
Subscriber Access Feature Pack License	MX240, MX480, MX960, M120, and M320	S-SA-FP
Software License for Secure Flow Mirroring Service (FlowTap) (does not require MS-DPC)	MX80, MX104, MX240, MX480, and MX960	S-SFM-FLOWTAP-IN
License to run one instance of the SFW and software on a MS-DPC	MX960, MX480, and MX240	S-SFW
Subscriber Service Management Feature Packet License—RADIUS and SRC-based service activation and deactivation per-service accounting features	MX240, MX480, MX960, M120, and M320	S-SSM-FP
Software license for one member of an MX Virtual Chassis	MX960, MX480, and MX240	S-VCR
Upgrade license—from MX10 to equivalent of MX40; allows additional 2x10G fixed ports to be used on the MX10 router	MX10-T	MX10-40-UPG
Upgrade license—from MX10 to equivalent of MX80; allows additional 4x10G fixed ports to be used on the MX10 router	MX10-T	MX10-80-UPG

Table 34: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Upgrade license—from MX40 to equivalent of MX80; allows additional 2x10G fixed ports to be used on the MX40 router	MX40-T	MX40-80-UPG
Upgrade license—from MX5 to equivalent of MX10; allows second MIC slot to be used on the MX5 router	MX5-T	MX5-10-UPG
Upgrade license—from MX5 to equivalent of MX40; allows second MIC slot and 2x10G fixed ports to be used on the MX5 router	MX5-T	MX5-40-UPG
Upgrade license—from MX5 to equivalent of MX80. Allows second MIC slot and 4x10G fixed ports to be used on the MX5 router	MX5-T	MX5-80-UPG
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 4000 through 8000 subscribers	MX80, MX960, MX480, and MX240	S-SA-UP-8K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 8000 through 16,000 subscribers	MX80, MX960, MX480, and MX240	S-SA-UP-16K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 16,000 through 32,000 subscribers	MX240, MX480, and MX960	S-SA-UP-32K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 32,000 through 64,000 subscribers	MX240, MX480, and MX960	S-SA-UP-64K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 64,000 through 96,000 subscribers	MX240, MX480, and MX960	S-SA-UP-96K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 96,000 through 128,000 subscribers	MX240, MX480, and MX960	S-SA-UP-128K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 128,000 through 256,000 subscribers	MX240, MX480, and MX960	S-SA-UP-256K

Table 34: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
License to use MX as Controller or Aggregation device for Junos Fusion. One license per MX is needed.	MX Series router	S-MX-AD-FUSION-LIC
License to run any supported EX4300 model as a satellite device in Junos Fusion mode. One license per EX4300 is needed	EX4300	S-MX-SAT-EX4300
License to run any supported QFX5100 model as a satellite device in Junos Fusion mode. One license per QFX5100 is needed	QFX5100	S-MX-SAT-QFX5100

Software Feature Licenses for SRX Series Devices

For information about how to purchase a software license, contact your Juniper Networks sales representative at <http://www.juniper.net/in/en/contact-us/>. Platform support depends on the Junos OS release in your installation.

Each feature license is tied to exactly one software feature, and that license is valid for exactly one device.



NOTE: For the most up-to-date license models available, contact your Juniper account team.

Software Features That Require Licenses on EX Series Switches

The following Junos OS features require an Enhanced Feature License (EFL) or Advanced Feature License (AFL) on EX Series devices:

- (EX2200 only) Bidirectional forwarding detection (BFD)
- (EX2200 only) Connectivity fault management (IEEE 802.lag)
- (EX2200 only) Internet Group Management Protocol version 1 (IGMPv1), IGMPv2, and IGMPv3
- (EX2200 and EX3300) OSPFv1/v2 (with 4 active interfaces)
- (EX2200 only) Protocol Independent Multicast (PIM) dense mode, PIM source-specific mode, PIM sparse mode
- (EX2200 and EX3300) Q-in-Q tunneling (IEEE 802.lad)
- (EX2200 only) Real-time performance monitoring (RPM)
- (EX3200, EX4200, EX4500, EX6200, and EX8200) Border Gateway Protocol (BGP) and multiprotocol BGP (MBGP)

- (EX3200, EX4200, EX4500, EX6200, and EX8200) Intermediate System-to-Intermediate System (IS-IS)
- (EX3200, EX4200, EX4500, EX6200, and EX8200) IPv6 protocols: OSPFv3, PIPng, IS-IS for IPv6, IPv6 BGP
- (EX3200, EX4200, EX4500, EX6200, and EX8200) MPLS with RSVP-based label-switched paths (LSPs) and MPLS-based circuit cross-connects (CCCs)

For more details regarding EX Series feature licenses, see “[Understanding Software Licenses for EX Series Switches](#)” on page 307.

For information about how to purchase a software license, contact your Juniper Networks sales representative at <http://www.juniper.net/in/en/contact-us/>.

Software Features That Require Licenses on the QFX Series



NOTE: If you try to configure a feature that is not licensed, you will receive syslog messages saying that you are using a feature that is licensable and that you do not possess a license for the feature. If you try to commit configuration changes for a feature that is not licensed, you will receive a commit warning saying that you have exceeded the allowed license limit for the feature.



NOTE: Virtual Extensible Local Area Network (VXLAN) is not supported on QFX3500 and QFX3600 devices. When you issue the `show licenses` command, you will see VXLAN in the CLI output, but the feature is not enabled.



NOTE: There is no separate license for Virtual Chassis like there is for Virtual Chassis Fabric.

Table 35 on page 299 lists the standard Junos OS features licenses and supported QFX Series devices. For information on disaggregated Junos OS feature licenses on the QFX5200-32C switch, see *Disaggregated Software Features That Require Licenses on the QFX Series*.

For information about how to purchase a software license, contact your Juniper Networks sales representative.

Table 35: Standard Junos OS Feature Licenses and Model Numbers for QFX Series Devices

Licensed Software Feature	Supported Devices	Number of Licenses Required	Model Number
QFX Series premium feature license for Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (IS-IS), and Virtual Extensible Local Area Network (VXLAN), and Open vSwitch Database (OVSDB)	QFX10002-36Q switch	One per switch	QFX10002-36Q-PFL
	QFX10002-72Q switch		QFX10002-72Q-PFL
	QFX10008 switch		QFX10008-PFL
	QFX10016 switch		QFX10016-PFL
	QFX5110-48S switch		QFX5K-C1-PFL
	QFX5110-32Q switch		QFX5K-C2-PFL
QFX Series advanced feature license for Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (IS-IS), Multi-protocol Label Switching (MPLS), and Virtual Extensible Local Area Network (VXLAN), and Open vSwitch Database (OVSDB)	QFX10002-36Q switch	One per switch	QFX10002-36Q-AFL
	QFX10002-72Q switch		QFX10002-72Q-AFL
	QFX10008 switch		QFX10008-AFL
	QFX10016 switch		QFX10016-AFL
	QFX5110-48S switch		QFX5K-C1-AFL
	QFX5110-32Q switch		QFX5K-C2-AFL
QFX Series advanced feature license for Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (IS-IS), Multi-protocol Label Switching (MPLS), and Virtual Extensible Local Area Network (VXLAN), and Open vSwitch Database (OVSDB)	QFX3500, QFX3600, QFX5100-48S, and QFX5100-48T switch	One per switch, two per Virtual Chassis, and two per Virtual Chassis Fabric	QFX-JSL-EDGE-ADV1
QFX Series advanced feature license for Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (IS-IS), Multi-protocol Label Switching (MPLS), and Virtual Extensible Local Area Network (VXLAN) and Open vSwitch Database (OVSDB)	QFX5100-24Q and QFX5100-96S switch	One per switch, two per Virtual Chassis, and two per Virtual Chassis Fabric	QFX5100-HDNSE-LIC

Table 35: Standard Junos OS Feature Licenses and Model Numbers for QFX Series Devices (*continued*)

Licensed Software Feature	Supported Devices	Number of Licenses Required	Model Number
QFX Series advanced feature license for Border Gateway Protocol (BGP)	QFX3100 Director device	One per Node device in a network Node group	QFX-JSL-DRCTR-ADV1
QFX Series advanced feature license for Fibre Channel	QFX3500 switch	One per switch on which fibre channel ports are configured	QFX-JSL-EDGE-FC
QFX Series advanced feature license for Fibre Channel	QFX3100 Director device	One per QFX3500 Node device on which fibre channel ports are configured	QFX-JSL-DRCTR-FC
QFX Series advanced feature license for Fibre Channel - Capacity 16	QFX3100 Director device	One for up to 16 QFX3500 Node devices on which fibre channel ports are configured	QFX-JSL-DRCTR-FC-C16
QFX Series feature license for enabling fabric mode	QFX3500 and QFX3600 device	One per device	QFX3000-JSL-EDGE-FAB
QFX Series feature license for base software for QFX3000-G QFabric system	QFX3100 Director device	One per QFX3000-G QFabric system	QFX3008-JSL-DRCTR-FAB
QFX Series feature license for base software for QFX3000-M QFabric system	QFX3100 Director device	One per QFX3000-M QFabric system	QFX3000M-JSL-DRCTR-FAB
QFX and EX Series feature license for enabling Media Access Control security (MACsec)	QFX switches that support MACsec. See <i>Understanding Media Access Control Security (MACsec)</i> .	One per switch, two per Virtual Chassis,	EX-QFX-MACSEC-AGG
Virtual Chassis Fabric (VCF)	All member devices in a Virtual Chassis Fabric (VCF)	Two per Virtual Chassis Fabric (VCF)	QFX-VCF-LIC

Software Features That Require Licenses on MX Series Routers Only

Table 34 on page 291 lists the licenses you can purchase for each MX Series software feature. Each license allows you to run the specified software feature on a single device.



NOTE: The DHCP server functionality for Junos OS is part of the subscriber management feature. You must have the S-SA-FP, S-MX80-SA-FP or S-MX104-SA-FP license in order to enable the DHCP server. For service accounting, you must also have S-SSM-FP.



NOTE: This is not a complete list of licenses. Contact your Juniper Networks representative for license information.

For information about how to purchase a software license, contact your Juniper Networks sales representative at <http://www.juniper.net/in/en/contact-us/>.

Table 36: Junos OS Feature License Model Number for MX Series Routers

Licensed Software Feature	Supported Devices	Model Number
Upgrade license—from MX80-10G-ADV to MX80-40G-ADV	MX80	MX80-10G40G-UPG-ADV-B
Upgrade license—from MX80-10G to MX80-40G	MX80	MX80-10G40G-UPG-B
Upgrade license—from MX80-40G-ADV to full MX80	MX80	MX80-40G-UPG-ADV-B
Upgrade license—from MX80-40G to full MX80	MX80	MX80-40G-UPG-B
Upgrade license—from MX80-5G-ADV to MX80-10G-ADV	MX80	MX80-5G10G-UPG-ADV-B
Upgrade license—from MX80-5G to MX80-10G	MX80	MX80-5G10G-UPG-B
Upgrade license to activate 2x10GE P2&3	MX104	S-MX104-ADD-2X10GE
Upgrade license to activate 2X10GE P0&1	MX104	S-MX104-UPG-2X10GE
Upgrade license to activate 4X10GE fixed ports on MX104	MX104	S-MX104-UPG-4X10GE

Table 36: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
License to support per VLAN queuing on MX80	MX80	S-MX80-Q
License to support per VLAN queuing on MX104	MX104	S-MX104-Q
Chassis-based software license for inline J-Flow monitoring on MX5, MX10, M40, MX80, and MX104 Series routers	MX5, MX10, M40, MX80, and MX104	S-JFLOW-CH-MX5-104
Chassis-based software license for inline J-Flow monitoring on MX240 routers	MX240	S-JFLOW-CH-MX240
Chassis-based software license for inline J-Flow monitoring on MX480 routers	MX480	S-JFLOW-CH-MX480
Chassis-based software license for inline J-Flow monitoring on MX960 routers	MX960	S-JFLOW-CH-MX960
Chassis-based software license for inline J-Flow monitoring on MX2008 routers	MX2008	S-JFLOW-CH-MX2008
Chassis-based software license for inline J-Flow monitoring on MX2010 routers	MX2010	S-JFLOW-CH-MX2010
Chassis-based software license for inline J-Flow monitoring on MX2020 routers	MX2020	S-JFLOW-CH-MX2020
Flow monitoring and accounting features using J-Flow service on any Modular Port Concentrator (MPC) or MS-DPC	MX240, MX480, and MX960	S-ACCT-JFLOW-CHASSIS
Software License for in-line J-Flow service on Trio MPCs	MX240, MX480, and MX960	S-ACCT-JFLOW-IN
Flow monitoring and accounting features using J-Flow service on any MPC limited to 10G of total JFLOW traffic	MX80	S-ACCT-JFLOW-IN-10G
Flow monitoring and accounting features using J-Flow service on any MPC limited to 10G of total JFLOW traffic	MX80	S-ACCT-JFLOW-IN-10G-UPG

Table 36: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Flow monitoring and accounting features using J-Flow service on any MPC limited to 5G of total JFLOW traffic	MX80	S-ACCT-JFLOW-IN-5G
Security services (IPsec, VPN and group VPN) license based on a single NPU for MS-MIC, MS-DPC or MS-MPC	MX Series router	S-ES-NPU
2000 IKE sessions on MS-DPC; Chassis based, limited to 6000 per Chassis	MX240, MX480, and MX960	S-ES-2K
4000 IKE sessions on MS-DPC; Chassis based, limited to 6000 per Chassis	MX240, MX480, and MX960	S-ES-4K
Upgrade from 2000 IKE sessions to 4000 IKE sessions on MS-DPC; Chassis based, limited to 6000 per Chassis	MX240, MX480, and MX960	S-ES-4K-UPG
6000 IKE sessions on MS-DPC; Chassis based, limited to 6000 per Chassis	MX240, MX480, and MX960	S-ES-6K
Upgrade from 4000 IKE sessions to 6000 IKE Sessions on MS-DPC; Chassis based, limited to 6000 per Chassis	MX240, MX480, and MX960	S-ES-6K-UPG
License to run stateful firewall on one NPU per MS-MIC, MS-DPC or MS-MPC	MX Series routers	S-FW-NPU
License to support DS3 Channelization (down to DS0) on each Modular Interface Card (MIC) for MIC-3D-8DS3-E3; also requires license S-MX80-Q when used on the MX80 platform	MX80, MX104, MX240, MX480, and MX960	S-MIC-3D-8CHDS3
License to support full-scale Layer 3 routes and Layer 3 VPN	MX80	S-MX80-ADV-R
License to support 256K routes	MX104	S-MX104-ADV-R1
License to support scaling Layer 3 and VPN routes to 1 million or more entries on MX104 platforms	MX104	S-MX104-ADV-R2
License to support full-scale Layer 3 routes and Layer 3 VPN on each slot for MPC-3D-16XGE-SFPP	MX240, MX480, and MX960	S-MPC-3D-16XGE-ADV-R
License to support full-scale Layer 3 routes and Layer 3 VPN on each slot for port queuing MPCs	MX240, MX480, and MX960	S-MPC-3D-PQ-ADV-R

Table 36: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
License to support full-scale Layer 3 routes and Layer 3 VPN on each slot for hierarchical quality of service (HQoS) MPCs	MX240, MX480, and MX960	S-MPC-3D-VQ-ADV-R
Subscriber Management Feature Pack License for MX80	MX80	S-MX80-SA-FP
Subscriber Management Feature Pack for MX104 series	MX104	S-MX104-SA-FP
Subscriber Service Management Feature Packet License—RADIUS and SRC-based service activation and deactivation per-service accounting features	MX80	S-MX80-SSM-FP
Subscriber Service Management Feature Packet License	MX104	S-MX104-SSM-FP
Upgrade to Traffic Direct Advanced (per MS-DPC)	MX960	S-MX-TD-UPG
License to run one instance of the NAT software on one NPU per MS-DPC	MX240, MX480, and MX960	S-NAT
License to support inline NAT software on MX5, MX10, MX40, MX80, MX104	MX5, MX10, MX40, MX80, and MX104	S-NAT-IN-MX5-104 (Replaces S-NAT-IN-MX40-MX80 and S-NAT-IN-MX5-MX10)
License to run one instance of the NAT software on one NPU per MS-MIC, MS-DPC, or MS-MPC	MX80, MX104, MX240, MX480, MX960, MX2010, and MX2020	S-NAT-NPU (Replaces S-NAT-IN-MX40-MX80-UPG)
License to run NAT using any MPC in an MX Chassis	MX240, MX480, and MX960	S-NAT-IN-MX-CHASSIS
Subscriber Access Feature Pack License Scaling (4000)	MX240, MX480, MX960, M120, M320, and MX80	S-SA-4K
Subscriber Access Feature Pack License Scaling (8000)	MX240, MX480, MX960, M120, M320, and MX80	S-SA-8K
Subscriber Access Feature Pack License Scaling (16,000)	MX240, MX480, MX960, and MX80	S-SA-16K
Subscriber Access Feature Pack License Scaling (32,000)	MX240, MX480, MX960, M120, and M320	S-SA-32K
Subscriber Access Feature Pack License Scaling (64,000)	MX240, MX480, MX960, M120, and M320	S-SA-64K

Table 36: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Subscriber Access Feature Pack License Scaling (96,000)	MX240, MX480, MX960, M120, and M320	S-SA-96K
Subscriber Access Feature Pack License Scaling (128,000)	MX240, MX480, MX960, M120, and M320	S-SA-128K
Subscriber Access Feature Pack License Scaling (256,000)	MX240, MX480, and MX960	S-SA-256K
Subscriber Access Feature Pack License	MX240, MX480, MX960, M120, and M320	S-SA-FP
Software License for Secure Flow Mirroring Service (FlowTap) (does not require MS-DPC)	MX80, MX104, MX240, MX480, and MX960	S-SFM-FLOWTAP-IN
License to run one instance of the SFW and software on a MS-DPC	MX960, MX480, and MX240	S-SFW
Subscriber Service Management Feature Packet License—RADIUS and SRC-based service activation and deactivation per-service accounting features	MX240, MX480, MX960, M120, and M320	S-SSM-FP
Software license for one member of an MX Virtual Chassis	MX960, MX480, and MX240	S-VCR
Upgrade license—from MX10 to equivalent of MX40; allows additional 2x10G fixed ports to be used on the MX10 router	MX10-T	MX10-40-UPG
Upgrade license—from MX10 to equivalent of MX80; allows additional 4x10G fixed ports to be used on the MX10 router	MX10-T	MX10-80-UPG
Upgrade license—from MX40 to equivalent of MX80; allows additional 2x10G fixed ports to be used on the MX40 router	MX40-T	MX40-80-UPG
Upgrade license—from MX5 to equivalent of MX10; allows second MIC slot to be used on the MX5 router	MX5-T	MX5-10-UPG
Upgrade license—from MX5 to equivalent of MX40; allows second MIC slot and 2x10G fixed ports to be used on the MX5 router	MX5-T	MX5-40-UPG

Table 36: Junos OS Feature License Model Number for MX Series Routers (*continued*)

Licensed Software Feature	Supported Devices	Model Number
Upgrade license—from MX5 to equivalent of MX80. Allows second MIC slot and 4x10G fixed ports to be used on the MX5 router	MX5-T	MX5-80-UPG
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 4000 through 8000 subscribers	MX80, MX960, MX480, and MX240	S-SA-UP-8K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 8000 through 16,000 subscribers	MX80, MX960, MX480, and MX240	S-SA-UP-16K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 16,000 through 32,000 subscribers	MX240, MX480, and MX960	S-SA-UP-32K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 32,000 through 64,000 subscribers	MX240, MX480, and MX960	S-SA-UP-64K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 64,000 through 96,000 subscribers	MX240, MX480, and MX960	S-SA-UP-96K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 96,000 through 128,000 subscribers	MX240, MX480, and MX960	S-SA-UP-128K
Upgrade license—Subscriber Access Feature Pack scaling license upgrade from 128,000 through 256,000 subscribers	MX240, MX480, and MX960	S-SA-UP-256K
License to use MX as Controller or Aggregation device for Junos Fusion. One license per MX is needed.	MX Series router	S-MX-AD-FUSION-LIC
License to run any supported EX4300 model as a satellite device in Junos Fusion mode. One license per EX4300 is needed	EX4300	S-MX-SAT-EX4300
License to run any supported QFX5100 model as a satellite device in Junos Fusion mode. One license per QFX5100 is needed	QFX5100	S-MX-SAT-QFX5100

- Related Documentation**
- [Junos OS Feature Licenses on page 281](#)
 - [License Enforcement on page 282](#)
 - [Junos OS Feature License Keys on page 283](#)
 - [Verifying Junos OS License Installation \(CLI\) on page 324](#)

Understanding Software Licenses for EX Series Switches

To enable and use some of the Juniper Networks operating system (Junos OS) features, you must purchase, install, and manage separate software licenses. If the switch has the appropriate software license, you can configure and use these features.

The Junos OS feature license (that is, the purchased authorization code) is universal. However, to conform to Junos OS feature licensing requirements, you must install a unique license key (a combination of the authorization code and the switch's serial number) on each switch.

For a Virtual Chassis deployment, two license keys are recommended for redundancy—one for the device in the master role and the other for the device in the backup role:

- In an EX8200 Virtual Chassis, the devices in the master and backup roles are always XRE200 External Routing Engines.
- In all other Virtual Chassis, the devices in the master and backup roles are switches.

You do not need additional license keys for Virtual Chassis member switches that are in the linecard role or for the redundant Routing Engine (RE) modules or the redundant Switch Fabric and Routing Engine (SRE) modules in an EX8200 member switch.

This topic describes:

- [Purchasing a Software Feature License on page 307](#)
- [Features Requiring a License on EX2200 Switches on page 308](#)
- [Features Requiring a License on EX2300 Switches on page 309](#)
- [Features Requiring a License on EX3300 Switches on page 309](#)
- [Features Requiring a License on EX3400 Switches on page 311](#)
- [Features Requiring a License on EX4300 Switches on page 311](#)
- [Features Requiring a License on EX4600 Switches on page 313](#)
- [Features Requiring a License on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches on page 314](#)
- [License Warning Messages on page 315](#)

Purchasing a Software Feature License

The following sections list features that require separate licenses. To purchase a software license, contact your Juniper Networks sales representative (<http://www.juniper.net/us/en/contact-us/sales-offices>). You will be asked to supply the

chassis serial number of your switch; you can obtain the serial number by running the **show chassis hardware** command.



NOTE: You are required to provide the 12-digit serial number when purchasing a license for an XRE200 External Routing Engine in an EX8200 Virtual Chassis.

The serial number listed on the XRE200 External Routing Engine serial ID label is 16 digits long. Use the last 12 digits of the 16-digit serial number to purchase the license.

You can use the **show chassis hardware** command output to display the 12-digit serial number of the XRE200 External Routing Engine.

Features Requiring a License on EX2200 Switches

For EX2200 switches, the following features can be added to basic Junos OS by installing an enhanced feature license (EFL):

- Bidirectional Forwarding Detection (BFD)
- Connectivity fault management (IEEE 802.1ag)
- IGMP (Internet Group Management Protocol) version 1 (IGMPv1), IGMPv2, and IGMPv3
- OSPFv1/v2 (with four active interfaces)
- Protocol Independent Multicast (PIM) dense mode, PIM source-specific mode, PIM sparse mode
- Q-in-Q tunneling (IEEE 802.1ad)
- Real-time performance monitoring (RPM)
- Virtual Router
- Virtual Router Redundancy Protocol (VRRP)

Table 37 on page 308 lists the EFLs that you can purchase for EX2200 switch models. If you have the license, you can run all of the enhanced software features mentioned above on your EX2200 switch.

Table 37: Junos OS EFL Part Number on EX2200 Switches

Switch Model	EFL Part Number
EX2200-C-12P-2G EX2200-C-12T-2G	EX-12-EFL
EX2200-24T-4G EX2200-24P-4G EX2200-24T-DC-4G	EX-24-EFL
EX2200-48T-4G EX2200-48P-4G	EX-48-EFL

Features Requiring a License on EX2300 Switches

EX2300 switches have enhanced feature licenses (EFLs).

To use the following features on the EX2300 switches, you must install an EFL:

- Bidirectional Forwarding Detection (BFD)
- IGMP (Internet Group Management Protocol) version 1 (IGMPv1), IGMPv2, and IGMPv3
- IPv6 routing protocols: Multicast Listener Discovery version 1 and 2 (MLD v1/v2), OSPFv3, PIM multicast, VRRPv6
- Multicast Source Discovery protocol (MSDP)
- OSPF v2/v3
- Protocol Independent Multicast (PIM) dense mode, PIM source-specific mode, PIM sparse mode
- Real-time performance monitoring (RPM)
- RIPng (RIPng is for RIP IPv6)
- Virtual Router Redundancy Protocol (VRRP)



NOTE: You require a paper license to create an EX2300 Virtual Chassis. If you do not have a paper license, contact Customer Support.

Table 38 on page 309 lists the EFLs that you can purchase for EX2300 switch models. If you have the license, you can run all of the enhanced software features mentioned above on your EX2300 switch.

Table 38: Junos OS EFL Part Number on EX2300 Switches

Switch Model	EFL Part Number
EX2300-24T EX2300-24P EX2300-C-12P EX2300-C-12T	EX-24-EFL

Features Requiring a License on EX3300 Switches

Two types of licenses are available on EX3300 switches: enhanced feature licenses (EFLs) and advanced feature licenses (AFLs).

To use the following features on the EX3300 switches, you must install an EFL:

- Bidirectional Forwarding Detection (BFD)
- IGMP (Internet Group Management Protocol) version 1 (IGMPv1), IGMPv2, and IGMPv3

- IPv6 routing protocols: Multicast Listener Discovery version 1 and 2 (MLD v1/v2), OSPFv3, PIM multicast, VRRPv6, virtual router support for unicast and filter-based forwarding (FBF)
- OSPFv1/v2
- Protocol Independent Multicast (PIM) dense mode, PIM source-specific mode, PIM sparse mode
- Q-in-Q tunneling (IEEE 802.1ad)
- Real-time performance monitoring (RPM)
- Virtual Router
- Virtual Router Redundancy Protocol (VRRP)

Table 39 on page 310 lists the EFLs that you can purchase for EX3300 switch models. If you have the license, you can run all of the enhanced software features mentioned above on your EX3300 switch.

Table 39: Junos OS EFL Part Number on EX3300 Switches

Switch Model	EFL Part Number
EX3300-24T EX3300-24P EX3300-24T-DC	EX-24-EFL
EX3300-48T EX3300-48T-BF EX3300-48P	EX-48-EFL

To use the following feature on EX3300 switches, you must install an AFL:

- Border Gateway Protocol (BGP) and multiprotocol BGP (MBGP)
- IPv6 routing protocols: IPv6 BGP and IPv6 for MBGP
- Virtual routing and forwarding (VRF) BGP

Table 40 on page 310 lists the AFLs that you can purchase for EX3300 switch models. For EX3300 switches, you must purchase and install a corresponding EFL along with the AFL to enable the advanced license features. If you have both these licenses, you can run all of the advanced software features mentioned above on your EX3300 switch.

Table 40: Junos OS AFL Part Number on EX3300 Switches

Switch Model	AFL Part Number
EX3300-24T EX3300-24P EX3300-24T-DC	EX-24-AFL
EX3300-48T EX3300-48T-BF EX3300-48P	EX-48-AFL

Features Requiring a License on EX3400 Switches

EX3400 switches has an enhanced feature licenses (EFLs) and MACSec license.

To use the following features on the EX3400 switches, you must install an EFL:

- Bidirectional Forwarding Detection (BFD)
- IGMP (Internet Group Management Protocol) version 1 (IGMPv1), IGMPv2, and IGMPv3
- IPv6 routing protocols: : Multicast Listener Discovery version 1 and 2 (MLD v1/v2), OSPFv3, PIM multicast, VRRPv6, virtual router support for unicast and filter-based forwarding (FBF)
- Multicast Source Discovery Protocol (MSDP)
- OSPF v2/v3
- Protocol Independent Multicast (PIM) dense mode, PIM source-specific mode, PIM sparse mode
- Real-time performance monitoring (RPM)
- RIPng (RIPng is for RIP IPv6)
- Unicast reverse-path forwarding (RPF)
- Virtual Router
- Virtual Router Redundancy Protocol (VRRP)

Table 41 on page 311 lists the EFLs that you can purchase for EX3400 switch models. If you have the license, you can run all of the enhanced software features mentioned above on your EX3400 switch.

Table 41: Junos OS EFL Part Number on EX3400 Switches

Switch Model	EFL Part Number
EX3400-24T EX3400-24P	EX-24-EFL
EX3400-48T EX3400-48P EX3400-48T-AFI	EX-48-EFL

Features Requiring a License on EX4300 Switches

Two types of licenses are available on EX4300 switches: enhanced feature licenses (EFLs) and advanced feature licenses (AFLs).

To use the following features on the EX4300 switches, you must install an EFL:

- Bidirectional Forwarding Detection (BFD)
- Connectivity fault management (IEEE 802.1ag)

- IGMP (Internet Group Management Protocol) version 1 (IGMPv1), IGMPv2, and IGMPv3
- Multicast Source Discovery Protocol (MSDP)
- OSPFv2/v3
- Protocol Independent Multicast (PIM) dense mode, PIM source-specific mode, PIM sparse mode
- Real-time performance monitoring (RPM)
- RIPv6 (RIPv6 is for RIPv6)
- Unicast reverse-path forwarding (RPF)
- Virtual Router
- Virtual Router Redundancy Protocol (VRRP)

[Table 42 on page 312](#) lists the EFLs that you can purchase for EX4300 switch models. If you have the license, you can run all of the enhanced software features mentioned above on your EX4300 switch.

Table 42: Junos OS EFL Part Number on EX4300 Switches

Switch Model	EFL Part Number
EX4300-24T EX4300-24P	EX4300-24-EFL
EX4300-48P EX4300-48T EX4300-48T-AFI EX4300-48T-DC EX4300-48T-DC-AFI	EX4300-48-EFL
EX4300-32F EX4300-32F-DC	EX4300-32F-EFL

To use the following features on EX4300 switches, you must install an AFL:

- Border Gateway Protocol (BGP) and multiprotocol BGP (MBGP)
- Intermediate System-to-Intermediate System (IS-IS)

[Table 43 on page 312](#) lists the AFLs that you can purchase for EX4300 switch models. For EX4300 switches, you must purchase and install a corresponding EFL along with the AFL to enable the advanced license features. If you have both these licenses, you can run all of the advanced software features mentioned above on your EX4300 switch.

Table 43: Junos OS AFL Part Number on EX4300 Switches

Switch Model	AFL Part Number
EX4300-24T EX4300-24P	EX4300-24-AFL

Table 43: Junos OS AFL Part Number on EX4300 Switches (*continued*)

Switch Model	AFL Part Number
EX4300-48P EX4300-48T EX4300-48T-AFI EX4300-48T-DC EX4300-48T-DC-AFI	EX4300-48-AFL
EX4300-32F EX4300-32F-DC	EX4300-32F-AFL

You must download a MACsec feature license to enable MACsec. The MACsec feature license is an independent feature license; the enhanced feature licenses (EFLs) or advanced feature licenses (AFLs) that must be purchased to enable some features on EX Series switches cannot be purchased to enable MACsec.

To purchase a feature license for MACsec, contact your Juniper Networks sales representative (<http://www.juniper.net/us/en/contact-us/sales-offices>). The Juniper sales representative will provide you with a feature license file and a license key.

MACsec is supported on EX4300 switches.

Features Requiring a License on EX4600 Switches

To use the following features on EX4600 switches, you must install an advanced feature license:

- Border Gateway Protocol (BGP) and multiprotocol BGP (MBGP)
- Intermediate System-to-Intermediate System (IS-IS)
- Multiprotocol Label Switching (MPLS)

Table 44 on page 313 lists the AFLs that you can purchase for EX4600 switch models.

Table 44: Junos OS AFL Part Number on EX4600 Switches

Switch Model	AFL Part Number
EX4600-40F	EX4600-AFL

You must download a MACsec feature license to enable MACsec. The MACsec feature license is an independent feature license; the enhanced feature licenses (EFLs) or advanced feature licenses (AFLs) that must be purchased to enable some features on EX Series switches cannot be purchased to enable MACsec.

To purchase a feature license for MACsec, contact your Juniper Networks sales representative (<http://www.juniper.net/us/en/contact-us/sales-offices>). The Juniper sales representative will provide you with a feature license file and a license key.

MACsec is supported on EX4600 switches.

Features Requiring a License on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches

To use the following features on EX3200, EX4200, EX4500, EX4550, EX8200, and EX9200 switches, you must install an advanced feature license (AFL):

- Border Gateway Protocol (BGP) and multiprotocol BGP (MBGP)
- Ethernet VPN (available only on EX9200 switches)
- Intermediate System-to-Intermediate System (IS-IS)
- IPv6 routing protocols: IS-IS for IPv6, IPv6 BGP, IPv6 for MBGP
- Logical systems (available only on EX9200 switches)
- MPLS with RSVP-based label-switched paths (LSPs) and MPLS-based circuit cross-connects (CCCs) (Not supported on EX9200 switches)
- Open vSwitch Database (OVSDb) (available only on EX9200 switches)
- Virtual Extensible LAN (VXLAN) (available only on EX9200 switches)

To use the following features on Juniper Networks EX6200 Ethernet Switches, you must install an advanced feature license (AFL):

- Border Gateway Protocol (BGP)
- Intermediate System-to-Intermediate System (IS-IS)
- IPv6 routing protocols: IS-IS for IPv6, IPv6 BGP

[Table 45 on page 314](#) lists the AFLs that you can purchase for EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 switches. If you have the license, you can run all of the advanced software features mentioned above on your EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, or EX9200 switch. An EFL is not applicable to this range of switches.

Table 45: Junos OS AFL Part Number on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches

Switch Model	AFL Part Number
EX3200-24P EX3200-24T EX4200-24F EX4200-24P EX4200-24PX EX4200-24T	EX-24-AFL
EX3200-48P EX3200-48T EX4200-48F EX4200-48P EX4200-48PX EX4200-48T	EX-48-AFL

Table 45: Junos OS AFL Part Number on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches (*continued*)

Switch Model	AFL Part Number
EX4500-40F-BF EX4500-40F-BF-C EX4500-40F-FB EX4500-40F-FB-C	EX-48-AFL
EX4550	EX4550-AFL
EX6210	EX6210-AFL
EX8208	EX8208-AFL
EX8216	EX8216-AFL
EX-XRE200	EX-XRE200-AFL
EX9204	EX9204-AFL
EX9208	EX9208-AFL
EX9214	EX9214-AFL

You must download a MACsec feature license to enable MACsec. The MACsec feature license is an independent feature license; the enhanced feature licenses (EFLs) or advanced feature licenses (AFLs) that must be purchased to enable some features on EX Series switches cannot be purchased to enable MACsec.

To purchase a feature license for MACsec, contact your Juniper Networks sales representative (<http://www.juniper.net/us/en/contact-us/sales-offices>). The Juniper sales representative will provide you with a feature license file and a license key.

MACsec is supported on EX4200 and EX4550 switches.

License Warning Messages

For using features that require a license, you must install and configure a license key. To obtain a license key, use the contact information provided in your certificate.

If you have not purchased the AFL or EFL and installed the license key, you receive warnings when you try to commit the configuration:

```
[edit protocols]
  'bgp'
    warning: requires 'bgp' license
error: commit failed: (statements constraint check failed)
```

The system generates system log (**syslog**) alarm messages notifying you that the feature requires a license—for example:

```
Sep 3 05:59:11 craftd[806]: Minor alarm set, BGP Routing Protocol usage
requires a license
Sep 3 05:59:11 alarmd[805]: Alarm set: License color=YELLOW, class=CHASSIS,
reason=BGP Routing Protocol usage requires a license
Sep 3 05:59:11 alarmd[805]: LICENSE_EXPIRED: License for feature bgp(47) expired
```

Output of the **show system alarms** command displays the active alarms:

```
user@switch> show system alarms
1 alarm currently active
Alarm time          Class  Description
2009-09-03 06:00:11 UTC  Minor  BGP Routing Protocol usage requires a license
```

**Related
Documentation**

- *Managing Licenses for the EX Series Switch (CLI Procedure)*
- *Managing Licenses for the EX Series Switch (J-Web Procedure)*
- *Monitoring Licenses for the EX Series Switch*
- *License Key Components for the EX Series Switch*

CHAPTER 16

Installing and Managing Licenses

- [Adding New Licenses \(CLI Procedure\) on page 317](#)
- [Deleting License Keys \(CLI\) on page 322](#)
- [Saving License Keys \(CLI\) on page 324](#)
- [Verifying Junos OS License Installation \(CLI\) on page 324](#)

Adding New Licenses (CLI Procedure)

Before adding new licenses, complete the following tasks:

- Purchase the required licenses.
- Establish basic network connectivity with the router or switch. For instructions on establishing basic connectivity, see the *Getting Started Guide* or *Quick Start Guide* for your device.

There are two ways to add licenses using the Junos OS CLI:

- The **system license keys key** configuration statement enables you to configure and delete license keys in a Junos OS CLI configuration file.
- The **request system license add** operational command installs a license immediately.



NOTE: On QFabric systems, install your licenses in the default partition of the QFabric system and not on the individual components (Node devices and Interconnect devices).

To add licenses, complete one of the following procedures:

- [Installing a License Using a Configuration Statement on page 317](#)
- [Installing a License Using an Operational Command on page 321](#)

Installing a License Using a Configuration Statement

Starting with Junos OS Release 15.1, you can configure and delete license keys in a Junos OS CLI configuration file. The **system license keys key** statement at the **[edit]** hierarchy level installs a license by using a configuration statement.



NOTE: The `system license keys key` configuration statement is not required to install a license. The operational command `request system license add` installs a license immediately. But because the `set system license keys key` command is a configuration statement, you can use it to install a license as part of a configuration commit, either directly or by configuration file.

The license keys are validated and installed after a successful commit of the configuration file. If a license key is invalid, the commit fails and issues an error message. You can configure individual license keys or multiple license keys by issuing Junos OS CLI commands or by loading the license key configuration contained in a file. All installed license keys are stored in the `/config/license/` directory.

Select a procedure to install a license using configuration:

- [Installing Licenses Using the CLI Directly on page 318](#)
- [Installing Licenses Using a Configuration File on page 319](#)

Installing Licenses Using the CLI Directly

To install an individual license key using the Junos OS CLI:

1. Issue the `set system license keys key name` statement.

The *name* parameter includes the license ID and the license key. For example:

```
[edit]
user@device# set system license keys key "JUNOS_TEST_LIC_FEAT xxxxxx xxxxxx
xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx"
```

To install multiple license keys in the Junos OS CLI, issue the `set system license keys key name` statement for each license key to install. For example:

```
[edit]
user@device# set system license keys key "key_1"
set system license keys key "key_2"
set system license keys key "key_2"
set system license keys key "key_4"
```

2. Issue the `commit` command.

```
[edit]
user@device# commit
commit complete
```

3. Verify that the license key was installed.

For example:

```
user@device# run show system license
```

License usage:

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
sdk-test-feat1	0	1	0	permanent

```

Licenses installed:
  License identifier: JUNOS_TEST_LIC_FEAT
  License version: 2
  Features:
    sdk-test-feat1  - JUNOS SDK Test Feature 1
                    permanent

```

Alternatively, you can issue the **show system license** command from operational mode.

Installing Licenses Using a Configuration File

Before you begin, prepare the configuration file. In this example, use the Unix shell **cat** command to write the **license.conf** file:

1. Go to the shell.

```

[edit]
user@device# exit
user@device> exit
%

```

2. Open the new **license.conf** file.

```
% cat > license.conf
```

3. Type the configuration information for the license key or keys:

- For a single license, for example, type the following content:

```

system {
  license {
    keys {
      key "JUNOS_TEST_LIC_FEAT xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx";
    }
  }
}

```

- For multiple license keys, for example, type something like this:

```

system {
  license {
    keys {
      key "key_1"
      key "key_2"
      key "key_3"
      ...
      key "key_n"
    }
  }
}

```

4. Press Ctrl+d to save the file.

To install a license key configuration in a file:

1. Go to the CLI configuration mode.

```
% cli
user@device> configure
[edit]
user@device#
```

2. Load and merge the license configuration file.

For example:

```
user@device# load merge license.conf
load complete
```

3. Issue the **show | compare** command to see the configuration.

For example:

```
[edit]
user@device# show | compare
[edit system]
+   license {
+       keys {
+           key "JUNOS_TEST_LIC_FEAT xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx";
+       }
+   }
```

4. Issue the **commit** command.

```
[edit]
user@device# commit
```

5. To verify that the license key was installed, issue the **show system license** command.

For example:

```
root@switch> show system license
```

License usage:

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
sdk-test-feat1	0	1	0	permanent

Licenses installed:

```
License identifier: JUNOS_TEST_LIC_FEAT
License version: 2
Features:
  sdk-test-feat1  - JUNOS SDK Test Feature 1
                  permanent
```

Installing a License Using an Operational Command

Complete the procedure that relates to your system:

- [Adding a License to a Device with a Single Routing Engine on page 321](#)
- [Adding a License to a Device with Dual Routing Engines on page 321](#)

Adding a License to a Device with a Single Routing Engine

To add a new license key to the device using an operational command:

1. From the CLI operational mode, enter one of the following CLI commands:

- To add a license key from a file or URL, enter the following command, specifying the filename or the URL where the key is located:

```
user@host> request system license add filename | url
```

- To add a license key from the terminal, enter the following command:

```
user@host> request system license add terminal
```

2. When prompted, enter the license key, separating multiple license keys with a blank line.

If the license key you enter is invalid, an error appears in the CLI output when you press Ctrl+d to exit license entry mode.

3. Go on to [“Verifying Junos OS License Installation \(CLI\)” on page 324](#).

Adding a License to a Device with Dual Routing Engines

On routers that have graceful Routing Engine switchover (GRES) enabled, after successfully adding the new license on the master Routing Engine, the license keys are automatically synchronized on the backup Routing Engine as well. However, in case GRES is not enabled, the new license is added on each Routing Engine separately. This ensures that the license key is enabled on the backup Routing Engine during changeover of mastership between the Routing Engines.

To add a new license key to a router with dual Routing Engines without GRES:

1. After adding the new license key on the master Routing Engine, use the **request chassis routing-engine master switch** command to have the backup Routing Engine become the master Routing Engine.
2. Log in to the active Routing Engine and add the new license key, repeat the same step.



NOTE: Adding a license key to the router or switch might be delayed if a kernel resynchronization operation is in progress at that time. The following message is displayed on the CLI when the license-adding operation is about to be delayed:

A kernel re-sync operation is in progress. License update may take several minutes to complete.

Related Documentation

- [Deleting License Keys \(CLI\) on page 322](#)
- [Junos OS Feature Licenses on page 281](#)
- [Verifying Junos OS License Installation \(CLI\) on page 324](#)
- [request system license add on page 382](#)

Deleting License Keys (CLI)

Before deleting a license, ensure that the features enabled by the license will not be needed.

You can use the **request system license delete** operational command, or the **delete** or **deactivate** configuration command to delete a license:

- [Using the Operational Command to Delete Licenses on page 322](#)
- [Using a Configuration Command to Delete Licenses on page 323](#)

Using the Operational Command to Delete Licenses

To delete licenses using the **request system license delete** command:

1. Display the licenses available to be deleted.

```
user@host> request system license delete license-identifier-list ?
Possible completions:
E00468XXX4      License key identifier
JUNOS10XXX1     License key identifier
JUNOS10XXX2     License key identifier
JUNOS10XXX3     License key identifier
JUNOS10XXX4     License key identifier
[               Open a set of values
```

2. To delete a license key or keys from a device using the CLI operational mode, select one of the following methods:
 - Delete a single license by specifying the license ID. Using this option, you can delete only one license at a time.

```
user@host> request system license delete license-identifier
```
 - Delete all license keys from the device.

```
user@host> request system license delete all
```

- Delete multiple license keys from the device. Specify the license identifier for each key and enclose the list of identifiers in brackets.

```
user@host> request system license delete license-identifier-list [JUNOS10XXX1
JUNOS10XXX3 JUNOS10XXX4 ...]
```

```
Delete license(s) ?
[yes,no] (no) yes
```

3. Verify the license was deleted by entering the **show system license** command.

Using a Configuration Command to Delete Licenses

Starting in Junos OS Release 16.1, to remove licenses from the configuration, you can use either the **delete** or **deactivate** configuration command. The **delete** command deletes a statement or identifier, and all subordinate statements and identifiers contained within the specified statement path are deleted with it. The **deactivate** command adds the **inactive:** tag to a statement, effectively commenting out the statement or identifier from the configuration. Statements or identifiers marked as inactive do not take effect when you issue the **commit** command. To remove the **inactive:** tag from a statement, issue the **activate** command. Statements or identifiers that have been activated take effect when you next issue the **commit** command.

The following procedure uses the **delete** command, but you could use the **deactivate** command as well.

To delete one or all licenses using the **delete** command:



NOTE: You can use the **deactivate** command instead of the **delete** command in this procedure.

1. Display the licenses available to be deleted.

Issue the **run request system license delete license-identifier-list ?** command from the configuration mode of the CLI.

```
[edit]
user@host# run request system license delete license-identifier-list ?
```

A list of licenses on the device is displayed:

```
Possible completions:
E00468XXX4      License key identifier
JUNOS10XXX1      License key identifier
JUNOS10XXX2      License key identifier
JUNOS10XXX3      License key identifier
JUNOS10XXX4      License key identifier
[                Open a set of values
```

2. Delete the license or licenses you want.

- To delete a single license, for example:

```
[edit]
```

```
user@host# delete system license keys key "E00468XXX4"
```

- To delete all licenses, for example:

```
[edit]
```

```
user@host# delete system license keys
```

3. Commit the configuration by entering the **commit** command.
4. Verify the license was deleted by entering the **show system license** command.

Release History Table

Release	Description
16.1	Starting in Junos OS Release 16.1, to remove licenses from the configuration, you can use either the delete or deactivate configuration command.

Related Documentation

- [Verifying Junos OS License Installation \(CLI\) on page 324](#)
- [Adding New Licenses \(CLI Procedure\) on page 317](#)

Saving License Keys (CLI)

To save the licenses installed on a device:

1. From operational mode, do one of the following tasks
 - To save the installed license keys to a file or URL, enter the following command:

```
user@host> request system license save filename | url
```

For example, the following command saves the installed license keys to a file named **license.config**:

```
user@host> request system license save filename | url
```

- To output installed license keys to the terminal, enter the following command:

```
user@host> request system license save terminal
```

Related Documentation

- [Verifying Junos OS License Installation \(CLI\) on page 324](#)
- [Adding New Licenses \(CLI Procedure\) on page 317](#)
- [Deleting License Keys \(CLI\) on page 322](#)

Verifying Junos OS License Installation (CLI)

To verify Junos OS license management, perform the following tasks:

- [Displaying Installed Licenses on page 325](#)
- [Displaying License Usage on page 326](#)

Displaying Installed Licenses

Purpose Verify that the expected licenses are installed and active on the device.

Action From the CLI, enter the **show system license** command.

Sample Output

```
user@host> show system license
License usage:

```

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
subscriber-acct	0	1	0	permanent
subscriber-auth	0	1	0	permanent
subscriber-addr	0	1	0	permanent
subscriber-vlan	0	1	0	permanent
subscriber-ip	0	1	0	permanent
scale-subscriber	0	1000	0	permanent
scale-l2tp	0	1000	0	permanent
scale-mobile-ip	0	1000	0	permanent

```

Licenses installed:
License identifier: E000185416
License version: 2
Features:
  subscriber-acct - Per Subscriber Radius Accounting
                  permanent
  subscriber-auth - Per Subscriber Radius Authentication
                  permanent
  subscriber-addr - Address Pool Assignment
                  permanent
  subscriber-vlan - Dynamic Auto-sensed Vlan
                  permanent
  subscriber-ip   - Dynamic and Static IP
                  permanent

```

Meaning The output shows a list of the license usage and a list of the licenses installed on the device. Verify the following information:

- Each license is present. Licenses are listed in ascending alphanumeric order by license ID.
- The state of each license is **permanent**.



NOTE: A state of invalid indicates that the license key is not a valid license key. Either it was entered incorrectly or it is not valid for the specific device.

- The feature for each license is the expected feature. The features enabled are listed by license. An all-inclusive license has all features listed.
- All configured features have the required licenses installed. The Licenses needed column must show that no licenses are required.

Displaying License Usage

Purpose Verify that the licenses fully cover the feature configuration on the device.

Action From the CLI, enter the **show system license usage** command.

Sample Output

```
user@host> show system license usage
```

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
subscriber-addr	1	0	1	29 days
scale-subscriber	0	1000	0	permanent
scale-l2tp	0	1000	0	permanent
scale-mobile-ip	0	1000	0	permanent

Meaning The output shows any licenses installed on the device and how they are used. Verify the following information:

- Any configured licenses appear in the output. The output lists features in ascending alphabetical order by license name. The number of licenses appears in the third column. Verify that you have installed the appropriate number of licenses.
- The number of licenses used matches the number of configured features. If a licensed feature is configured, the feature is considered used. The sample output shows that the subscriber address pooling feature is configured.
- A license is installed on the device for each configured feature. For every feature configured that does not have a license, one license is needed.

For example, the sample output shows that the subscriber address feature is configured but that the license for the feature has not yet been installed. The license must be installed within the remaining grace period to be in compliance.

PART 4

Troubleshooting Information

- [Troubleshooting Software Installation on page 329](#)

Troubleshooting Software Installation

- [Troubleshooting Software Installation on page 329](#)
- [Troubleshooting a Switch That Has Booted from the Backup Junos OS Image on page 332](#)
- [Disk Space Management for Junos OS Installation on page 333](#)
- [Verifying PIC Combinations on page 334](#)

Troubleshooting Software Installation

This topic describes troubleshooting issues with software installations on EX Series switches.

- [Recovering from a Failed Software Upgrade on an EX Series Switch on page 329](#)
- [Rebooting from the Inactive Partition on page 330](#)
- [Freeing Disk Space for Software Installation on page 331](#)
- [Installation from the Boot Loader Generates 'cannot open package' Error on page 331](#)

Recovering from a Failed Software Upgrade on an EX Series Switch

Problem **Description:** If Junos OS loads but the CLI is not working, or if the switch has no software installed, use this recovery installation procedure to install Junos OS.

Solution If there is already a Junos OS image on the system, you can either install the new Junos OS package in a separate partition and have both Junos OS images remain on the system, or you can wipe the disk clean before the new installation proceeds.

If there is no Junos OS image on the system, follow the instructions in [“Booting an EX Series Switch Using a Software Package Stored on a USB Flash Drive” on page 200](#) to get an image on the system and boot the switch.

To perform a recovery installation:

1. Power on the switch.

The loader script starts.

After the message **Loading /boot/defaults/loader.conf** displays, you are prompted with the following:

Hit [Enter] to boot immediately, or space bar for command prompt.

2. Press the space bar to enter the manual loader.

The **loader>** prompt displays.

3. Enter the following command:

```
loader> install [- --format] [- --external] source
```

where:

- **format**—Use this option to wipe the installation media before installing the software package. If you do not include this option, the system installs the new Junos OS package in a different partition from the partition used by the most recently installed Junos OS package.
- **external**—Use this option to install the software package on an external medium.
- **source**—Represents the name and location of the Junos OS package either on a server on the network or as a file on the USB flash drive:
 - Network address of the server and the path on the server; for example, **tftp://192.168.1.28/junos/jinstall-ex-4200-9.4R1.5-domestic-signed.tgz**
 - The Junos OS package on a USB device is commonly stored in the root drive as the only file; for example, **file:///jinstall-ex-4200-9.4R1.5-domestic-signed.tgz**

The boot process proceeds as normal and ends with a login prompt.

Rebooting from the Inactive Partition

Problem Description: EX Series switches shipped with Junos OS Release 10.4R2 or earlier have Junos OS loaded on the system disk in partition 1. The first time you upgrade, the new software package is installed in partition 2. When you finish the installation and reboot, partition 2 becomes the active partition. Similarly, subsequent software packages are installed in the inactive partition, which becomes the active partition when you reboot at the end of the installation process.

On switches shipped with Release 10.4R3 and later, the same Junos OS image is loaded in each of the two root partitions, and you should copy the new software image to the alternate partition each time you upgrade.

If you performed an upgrade and rebooted, the system resets the active partition. You can use this procedure to manually boot from the inactive partition.



NOTE: If you have completed the installation of the software image but have not yet rebooted, issue the **request system software rollback** command to return to the original software installation package.

Solution Reboot from the inactive partition:

```
user@switch> request system reboot slice alternate
```



NOTE: If you cannot access the CLI, you can reboot from the inactive partition using the following procedure from the loader script prompt:

1. Unload and clear the interrupted boot from the active partition:

```
loader> unload
loader> unset vfs.root.mountfrom
```

2. Select the new (inactive) partition to boot from:

```
loader> set currdev=diskxsy:
```

where *x* is either 0 (internal) or 1 (external), and the *y* indicates the number of the inactive partition, either 1 or 2.

You must include the colon (:) at the end of this command.

3. Boot Junos OS from the inactive partition:

```
loader> boot
```

Freeing Disk Space for Software Installation

Problem **Description:** The software installation process requires a certain amount of unused disk space. If there is not enough space, you might receive an error message such as:

```
fetch: /var/tmp/incoming-package.tgz: No space left on device
```

Solution Identify and delete unnecessary files by using the `request system storage cleanup` command.

Installation from the Boot Loader Generates 'cannot open package' Error

Problem **Description:** When installing a Junos OS software image from the loader prompt, a "cannot open package error" is generated:

```
loader> install - -format
tftp://10.204.33.248/images/Flash_corr/official/jinstall-ex-4200-10.4I2011012-domestic-signed.tgz
Speed: 1000, full duplex
bootp: no reply
No response for RARP request
net_open: RARP failed
cannot open package (error 5)
```

Solution This might be due to the IP address, gateway IP address, netmask address, or server IP address not being properly set. You can set these values either from the shell or from the u-boot prompt.

To set these values from the shell:

```
% nvram setenv ipaddr 10.204.35.235
% nvram setenv netmask 255.255.240.0
% nvram setenv gatewayip 10.204.47.254
% nvram setenv serverip 10.204.33.248
```

To set these values from the u-boot prompt, log in to a console connection, reboot, and stop at the u-boot prompt (Cntrl+c):

```
=> setenv ipaddr 10.204.35.235
=> setenv gatewayip 10.204.47.254
=> setenv serverip 10.204.33.248
=> setenv netmask 255.255.240.0
=> saveenv
=> printenv Verify whether variables are set properly or not
=> boot
```

Related Documentation

- *Installing Software on an EX Series Switch with a Single Routing Engine (CLI Procedure)*
- *Upgrading Software on an EX6200 or EX8200 Standalone Switch Using Nonstop Software Upgrade (CLI Procedure)*
- [Installing Software on EX Series Switches \(J-Web Procedure\) on page 66](#)
- [Understanding Software Installation on EX Series Switches on page 44](#)
- [show system storage partitions \(EX Series Switches Only\) on page 513](#)

Troubleshooting a Switch That Has Booted from the Backup Junos OS Image

Problem Description: The switch boots from the backup root file partition. It is possible that the primary copy of Junos OS failed to boot properly, which could indicate that it is corrupted. This event is flagged in two ways:

- Upon login through the console or management port, the following warning message is displayed:

```
WARNING: THIS DEVICE HAS BOOTED FROM THE BACKUP JUNOS IMAGE
```

It is possible that the primary copy of JUNOS failed to boot up properly, and so this device has booted from the backup copy.

Please re-install JUNOS to recover the primary copy in case it has been corrupted.

- The following alarm message is generated:

```
user@switch> show chassis alarms
```



```

1 alarms currently active
Alarm time           Class  Description
2011-02-17 05:48:49 PST  Minor  Host 0 Boot from backup root

```

If the switch is in a Virtual Chassis, the switch member number appears in the **Description** field, where the switch is called a host.

Solution Install a new Junos OS image on the partition that had the corruption, or take a snapshot (use [request system snapshot](#)) of the currently active partition and use it to replace the image in the alternate partition:

If the switch is a standalone switch or a Virtual Chassis master switch, enter this command:

```
user@switch> request system snapshot slice alternate
```

If the switch is a Virtual Chassis member switch (not the master), enter this command on the Virtual Chassis:

```
user@switch> request system snapshot slice alternate member member-id
```

where *member-id* is the Virtual Chassis member ID number.

- Related Documentation**
- [Verifying Junos OS and Boot Loader Software Versions on an EX Series Switch on page 146](#)
 - [Troubleshooting Software Installation on page 329](#)
 - [show system storage partitions \(EX Series Switches Only\) on page 513](#)

Disk Space Management for Junos OS Installation

A Junos OS installation or upgrade may fail if your router has a shortage of disk space. If a disk space error occurs, use one or more of the following options to complete the installation:

- Use the **request system storage cleanup** command to delete unnecessary files and increase storage space on the router.
- Specify the **unlink** option when you use the **request system software add** command to install the Junos OS:
 - On the M Series, MX Series, and T Series routers, the **unlink** option removes the software package after a successful upgrade.
- Download the software packages you need from the Juniper Networks Support Web site at <http://www.juniper.net/support/>. The download program provides intelligent disk space management to enable installation.

Related Documentation • *Junos OS Configuration Using the CLI*

Verifying PIC Combinations

On Juniper Networks routing platforms, you can typically install any combination of Physical Interface Cards (PICs) on a single Enhanced Flexible PIC Concentrator (FPC) or in two PIC slots served by a single Layer 2/Layer 3 Packet Processing application-specific integrated circuit (ASIC).

Newer Junos OS services for some PICs can require significant Internet Processor ASIC memory, and some configuration rules limit certain combinations of PICs if they are installed on some platforms.

During software installation, the configuration checker in the installation program checks the router's PICs. If any configuration rules affect your PIC combinations, the installation process stops and displays a message similar to the following:

```
The combination of PICS in FPC slot 3 is not supported with this release
PIC slot 0 -
PIC slot 1 - 1x OC-12 ATM-II IQ
PIC slot 2 - 1x G/E IQ, 1000 BASE
PIC slot 3 - 1x Link Service (4)
If you continue the installation, one or more PICs on
FPC slot 3 might appear to be online but
cannot be enabled and cannot pass traffic with this release of JUNOS.
See the Release Notes for more information.
WARNING: This installation attempt will be aborted. If you
WARNING: wish to force the installation despite these warnings
WARNING: you may use the 'force' option on the command line.
pkg_add: package /var/tmp/jbundle-7.6R1.x-domestic-signed.tgz fails requirements
- not installed
```

The configuration checker has the following limitations:

- If a PIC is offline when you upgrade the router with new software, the configuration checker cannot detect PIC combinations affected by configuration rules and cannot warn about them.
- If you specify the **force** option when you upgrade the Junos OS, the configuration checker warns about the affected PIC combination and the software installation continues. However, after rebooting, one or more PICs might fail to initialize.
- The configuration checker looks for combinations of three affected PICs. If an Enhanced FPC contains four affected PICs, the script generates multiple warnings.

If you install a PIC into a router already running Junos OS, you can identify the presence of affected PIC combinations from messages in the system logging (**syslog**) file:

```
Feb 6 17:57:40 CE1 feb BCHIP 0: uCode overflow - needs 129 inst space to load
b3_atm2_LSI_decode for stream 12
Feb 6 17:57:41 CE1 chassisd[2314]: CHASSISD_IFDEV_DETACH_PIC:
ifdev_detach_pic(0/3)
Feb 6 17:57:41 CE1 feb BCHIP 0: binding b3_atm2_LSI_decode to stream 12 failed
```

```
Feb 6 17:57:41 CE1 feb PFE: can not bind B3 ucode prog b3_atm2_LSI_decode to FPC
0: stream 12
```

For more information about checking for unsupported PIC combinations, see the corresponding PIC guide for your router, the [Junos OS Release Notes](#), and *Technical Support Bulletin PSN-2004-12-002, PIC Combination Notes Summary* on the Juniper Networks Support Web site at <http://www.juniper.net/support/>.

For SRX Series Services Gateways

SRX5600 and SRX5800 devices support IOC or SPC on any given card slot, and there is no complexity in equipping the services gateways with the perfect balance of processing and I/O capacity. You can install up to 11 (on SRX5800) and 5 (SRX5600) SPCs and IOCs on the device. However, you must install at least one SPC on device. For more details, see [SRX5600 and SRX5800 Services Gateway Card Guide](#).

SRX3600 supports a maximum of up to seven SPCs, three NPCs, six IOCs, and 11 NP-IOCs per chassis. However you must install at least one SPCs and NPC on the chassis. SRX3400 supports a maximum of up to four SPCs, two NPCs, four IOCs, and six NP-IOCs per chassis. However you must install at least one SPCs and NPC on the chassis. On SRX3400 and SRX3600 devices you must install PICs on the front slots of the chassis. For more details, see [SR X1400](#) , [SRX3400](#) , and [SRX3600 Services Gateway Module Guide](#).

Related Documentation

- [Hardware Overview of SRX Series Services Gateways on page 31](#)
- [Storage Media Names for SRX Series Devices on page 36](#)

PART 5

Configuration Statements and Operational Commands

- [Configuration Statements on page 339](#)
- [Operational Commands on page 367](#)
- [VM Host Software Administrative Commands on page 519](#)
- [VM Host Software Monitoring Commands on page 545](#)

CHAPTER 18

Configuration Statements

- [auto-configuration on page 340](#)
- [auto-configuration \(System\) on page 341](#)
- [auto-image-upgrade on page 343](#)
- [auto-snapshot on page 344](#)
- [autoinstallation on page 345](#)
- [autoinstallation \(JNU Satellite Devices\) on page 346](#)
- [bootp on page 347](#)
- [commit on page 348](#)
- [configuration-servers on page 349](#)
- [delete-after-commit \(JNU Satellites\) on page 350](#)
- [file \(App Engine Virtual Machine Management Service\) on page 351](#)
- [flag \(App Engine Virtual Machine Management Service\) on page 353](#)
- [interfaces \(Autoinstallation\) on page 354](#)
- [level \(App Engine Virtual Machine Management Service\) on page 355](#)
- [license on page 356](#)
- [traceoptions \(App Engine Virtual Machine Management Service\) on page 358](#)
- [traceoptions \(System License\) on page 359](#)
- [vmhost on page 361](#)
- [vmhost management-if disable on page 362](#)
- [vmhost management-if link-mode on page 363](#)
- [vmhost management-if speed on page 364](#)
- [usb on page 365](#)
- [vmhost on page 366](#)

auto-configuration

Syntax	<pre>auto-configuration { command <i>binary-file-path</i>; disable; }</pre>
Hierarchy Level	[edit system processes]
Release Information	Statement introduced in Junos OS Release 8.5.
Description	Configure the autoconfiguration process.
Options	<ul style="list-style-type: none">• command <i>binary-file-path</i>—Path to the binary process.• disable—Disable the autoconfiguration process.
Required Privilege Level	system—To view this statement in the configuration. system-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Autoinstallation Overview on page 95• Example: Configuring Autoinstallation on SRX Series Devices on page 100

auto-configuration (System)

Syntax	<pre> auto-configuration { traceoptions { file { filename; files number; match regular-expression; size maximum-file-size; (world-readable no-world-readable); } flag flag; level (all error info notice verbose warning); no-remote-trace; } } </pre>
Hierarchy Level	[edit system]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure the autoconfiguration process.
Options	<p>traceoptions—Set the trace options.</p> <ul style="list-style-type: none"> file—Configure the trace file information. <ul style="list-style-type: none"> filename—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory <code>/var/log</code>. By default, the name of the file is the name of the process being traced. files number—Maximum number of trace files. When a trace file named <code>trace-file</code> reaches its maximum size, it is renamed to trace-file.0, then trace-file.1, and so on, until the maximum number of trace files is reached. The oldest archived file is overwritten. <p>If you specify a maximum number of files, you also must specify a maximum file size with the size option and a filename.</p> <p>Range: 2 through 1000 files</p> <p>Default: 10 files</p> match regular-expression—Refine the output to include lines that contain the regular expression. size maximum-file-size—Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named trace-file reaches this size, it is renamed trace-file.0. When trace-file again reaches its maximum size, trace-file.0 is renamed trace-file.1 and trace-file is renamed trace-file.0. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

If you specify a maximum number of files, you also must specify a maximum file size with the **size** option and a filename.

Syntax: x K to specify KB, x m to specify MB, or x g to specify GB

Range: 10 KB through 1 GB

Default: 128 KB

- **world-readable | no-world-readable**—By default, log files can be accessed only by the user who configures the tracing operation. The **world-readable** option enables any user to read the file. To explicitly set the default behavior, use the **no-world-readable** option.
- **flag**—Specify the tracing operation to perform. To specify more than one tracing operation, include multiple flag statements. You can include the following flags.
 - **all**—Trace all events.
 - **auth**—Trace VLAN authentication.
 - **configuration**—Trace configurations.
 - **interfaces**—Trace interface operations.
 - **io**—Trace I/O operations.
 - **rtsock**—Trace routing socket operations.
 - **ui**—Trace user interface operations.


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

Related Documentation	<ul style="list-style-type: none">• Autoinstallation Overview on page 95• Example: Configuring Autoinstallation on SRX Series Devices on page 100
------------------------------	--

auto-image-upgrade

Syntax	auto-image-upgrade;
Hierarchy Level	[edit chassis]
Release Information	Statement introduced in Junos OS Release 9.6 for EX Series switches.
Description	<p>Enable automatic software download on an EX Series switch acting as a DHCP client.</p> <p>The DHCP client EX Series switch compares the software package name in the DHCP server message to the name of the software package that booted the switch. If the software packages are different, the DHCP client EX Series switch downloads and installs the software package specified in the DHCP server message.</p> <p>Before you upgrade software using automatic software download, ensure that you have configured DHCP services for the switch, including configuring a path to a boot server and a boot file. See the Junos OS System Basics Configuration Guide for information about using the CLI to configure DHCP services and settings. See <i>Configuring DHCP Services (J-Web Procedure)</i> for information about using the J-Web interface to configure DHCP services and settings.</p>
Default	Automatic software download is disabled.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Upgrading Software by Using Automatic Software Download on page 144 • Understanding Software Installation on EX Series Switches on page 44 • <i>Understanding DHCP Services for Switches</i>

auto-snapshot

Syntax	auto-snapshot;
Hierarchy Level	[edit system]
Release Information	Statement introduced in Junos OS Release 12.3 for EX Series switches.
Description	<p>Enable the automatic snapshot feature, which allows the switch to automatically fix a corrupt Junos OS file in the primary root partition. If the automatic snapshot feature is enabled, the switch automatically takes a snapshot of the Junos OS root file system in the alternate root partition and copies it onto the primary root partition, thereby repairing the corrupt file in the primary root partition. The automatic snapshot procedure takes place whenever the system reboots from the alternate root partition, regardless of whether the reboot is due to a command or due to corruption of the primary root partition.</p>
	<div> NOTE: EX9200 switches do not support the automatic snapshot feature.</div>
Default	<ul style="list-style-type: none">• The automatic snapshot feature is enabled by default on the following EX Series switches:<ul style="list-style-type: none">• EX4550 switches• EX Series switches that ship with Junos OS Release 12.3R1 or later• The automatic snapshot feature is disabled by default on EX Series switches (except the EX4550 switches) running Junos OS Release 12.2 or earlier.• If the automatic snapshot feature was disabled by default before the switch was upgraded to Junos OS Release 12.3R1 or later, the feature remains disabled (for backward compatibility) by default after the upgrade.
Required Privilege Level	system—To view this statement in the configuration. system-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Understanding Resilient Dual-Root Partitions on Switches on page 111• show system auto-snapshot on page 488

autoinstallation

Syntax

```

autoinstallation {
  configuration-servers {
    url {
      password password;
    }
  }
  interfaces {
    interface-name {
      bootp;
      rarp;
    }
  }
  usb {
    disable;
  }
}

```

Hierarchy Level [edit system]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the configuration for autoinstallation.

Options The remaining statements are explained separately. See [CLI Explorer](#).

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

Related Documentation

- [Example: Configuring Autoinstallation on SRX Series Devices on page 100](#)

autoinstallation (JNU Satellite Devices)

Syntax	<pre>autoinstallation { delete-after-commit; configuration-servers { url; } interfaces { interface-name { bootp; rarp; } } }</pre>
Hierarchy Level	[edit system]
Release Information	Statement introduced in Junos OS Release 13.3 for satellite devices in a Junos Node Unifier (JNU) group.
Description	(Satellite devices in a JNU group) Download a configuration file automatically from an FTP or HTTP server. When you power on a router or switch configured for autoinstallation, it requests an IP address from a Dynamic Host Configuration Protocol (DHCP) server. When the router or switch has an address, it sends a request to a configuration server and downloads and installs a configuration.
Options	The remaining statements are explained separately. See CLI Explorer .
Required Privilege Level	system—To view this statement in the configuration. system-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Autoinstallation of Satellite Devices in a Junos Node Unifier Group on page 107• Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 105• Configuring Autoinstallation on JNU Satellite Devices on page 103• Verifying Autoinstallation on JNU Satellite Devices on page 108• delete-after-commit (JNU Satellites) on page 350• configuration-servers

bootp

Syntax	<pre>bootp { command <i>binary-file-path</i>; disable; failover (alternate-media other-routing-engine); }</pre>
Hierarchy Level	[edit system processes]
Release Information	Statement introduced in Junos OS Release 8.5.
Description	Specify the booting process.
Options	<ul style="list-style-type: none"> • command <i>binary-file-path</i>—Path to the binary process. • disable —Disable the booting process. • failover—Configure the device to reboot if the software process fails four times within 30 seconds, and specify the software to use during the reboot. <ul style="list-style-type: none"> • alternate-media—Configure the device to switch to backup media that contains a version of the system if a software process fails repeatedly. • other-routing-engine—Instruct the secondary Routing Engine to take mastership if a software process fails. If this statement is configured for a process, and that process fails four times within 30 seconds, then the device reboots from the secondary Routing Engine.
Required Privilege Level	system—To view this statement in the configuration. system-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> •

commit

Syntax

```
commit {
  delta-export;
  peers {
    authentication string;
    user user-name;
  }
  persist-groups-inheritance;
  server {
    commit-interval seconds;
    days-to-keep-error-logs days;
    maximum-aggregate-pool number;
    maximum-entries number;
    traceoptions {
      file {
        filename;
        files number;
        microsecond-stamp;
        size maximum-file-size;
        (world-readable | no-world-readable);
      }
      flag flag;
      no-remote-trace;
    }
  }
  peers-synchronize;
}
```

Hierarchy Level [edit system]

Release Information Statement introduced in Junos OS Release 12.1.

Description Configure the commit operation.

Options **delta-export**—(Optional) Configure system commit to export only the changes made in the candidate configuration instead of exporting the entire candidate configuration to the configuration database.

peers—(Optional) Configure options for the peers participating in commit synchronization.

persist-groups-inheritance—(Optional) Configure commit to quickly apply configuration groups that use wildcards.

server—(Optional) Configure a default batch commit.

peers-synchronize—(Optional) Synchronize commit action between the peers. The local peer (or requesting peer) on which you enable the peers-synchronize statement copies and loads its configuration to the remote (or responding) peer.

The remaining statements are explained separately. See [CLI Explorer](#).

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

Related Documentation • *Controlling Execution of Commit Scripts During Commit Operations*

configuration-servers

Syntax

```
configuration-servers {
  url {
    password password;
  }
}
```

Hierarchy Level [edit system autoinstallation]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure the URL address of a server from which the configuration files must be obtained.

You can download a configuration file automatically from an FTP, Hypertext Transfer Protocol (HTTP), or Trivial FTP (TFTP) servers. Examples of URLs:

- tftp://hostname/path/filename
- ftp://username:password@ftp.hostname.net
- http://hostname/path/filename
- http://username:password@httpconfig.sp.com

Options

- **url**—Specify the URL address of the server containing the configuration files.
- **password**—Specify the password for authentication with the configuration server. Specifying a password in URLs and in the *password* option might result in commit failure. We recommend you to use the *password* option for specifying the password.

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

Related Documentation • [Example: Configuring Autoinstallation on SRX Series Devices on page 100](#)

delete-after-commit (JNU Satellites)

Syntax	delete-after-commit;
Hierarchy Level	[edit system autoinstallation]
Release Information	Statement introduced in Junos OS Release 13.3 for satellite devices in a Junos Node Unifier (JNU) group.
Description	<p>Specify that during the subsequent commit operation of configuration settings (after the autoinstallation process successfully retrieves, installs, and commits the configuration), the autoinstallation configuration parameters be removed from the router. Removal of the autoinstallation parameters and statements from the committed configuration on the router ensures that the router does not attempt to perform an autoinstallation process when it is powered on the next time. Although you can optionally specify the interfaces to perform autoinstallation or configuration servers from which the files are to be downloaded, you must include the delete-after-commit statement to prevent the router from entering a recursive loop and repeatedly performing an autoinstallation every time it is powered on.</p>
Required Privilege Level	<p>system—To view this statement in the configuration.</p> <p>system-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• Autoinstallation of Satellite Devices in a Junos Node Unifier Group on page 107• Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 105• Configuring Autoinstallation on JNU Satellite Devices on page 103• Verifying Autoinstallation on JNU Satellite Devices on page 108• autoinstallation on page 346• configuration-servers

file (App Engine Virtual Machine Management Service)

Syntax	file { <i>filename</i> < <i>files number</i> > match no-world-readable size <i>size</i> world-readable }
Hierarchy Level	[edit system processes app-engine-virtual-machine-management-service traceoptions]
Release Information	Statement introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers. Statement introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers. Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Description	Trace file information for the Virtual Machine Management Daemon (vmmd), which communicates with the host OS.
Options	<p>filename—Name of the file in which the trace information is stored. By default, the file is created in the <code>/var/log</code> directory.</p> <p>files number—(Optional) Maximum number of trace files. When a trace file reaches the size specified by the size option, the filename is appended with 0 and compressed. For example, when a trace file named trace-file-log reaches size <i>size</i>, it is compressed and renamed as trace-file-log.0.gz. When trace-file-log reaches size <i>size</i> for the second time, trace-file-log.0.gz is renamed as trace-file-log.1.gz and trace-file-log is compressed and renamed as trace-file-log.0.gz. This renaming scheme ensures that the older logs have a greater index number. When number of trace files reaches <i>number</i>, the oldest file is deleted.</p> <p>If you specify a maximum number of files, you also must specify a maximum file size with the size option and a filename.</p> <p>Range: 2 through 1000</p> <p>Default: 10</p> <p>match—Refine the output to include only those lines that match the given regular expression.</p> <p>no-world-readable—Restrict file access to the user who created the trace files.</p> <p>sizesize—Maximum size of each trace file . By default, the number entered is treated as bytes. Alternatively, you can include a suffix to the number to indicate kilobytes (KB), megabytes (MB), or gigabytes (GB). If you specify a maximum file size, you also must specify a maximum number of trace files with the files option.</p> <p>Range: 10 KB through 1 GB</p> <p>Default: 128 KB</p>

world-readable—Enable unrestricted file access.

Required Privilege Level system-control

Related Documentation

- [traceoptions \(App Engine Virtual Machine Management Service\) on page 358](#)
- [level \(App Engine Virtual Machine Management Service\) on page 355](#)
- [flag \(App Engine Virtual Machine Management Service\) on page 353](#)

flag (App Engine Virtual Machine Management Service)

Syntax	flag (all ccif configuration heartbeat init miscellaneous platform pxe routing-instances snmp)
Hierarchy Level	[edit system processes app-engine-virtual-machine-management-service traceoptions]
Release Information	Statement introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers. Statement introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers. Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Description	Perform different tracing operations. To specify more than one tracing operation, include multiple flag statements.
Default	Tracing operations are not performed.
Options	<p>all—Trace all events.</p> <p>ccif—Trace compute node interface events. This is the default option.</p> <p>configuration—Trace configuration events.</p> <p>heartbeat—Trace compute node heartbeat-related events.</p> <p>init—Trace initialization events.</p> <p>miscellaneous—Trace miscellaneous events.</p> <p>platform—Trace platform-related events.</p> <p>pxe—Trace events related to Preboot Execution Environment (PXE).</p> <p>routing-instances—Trace events related to routing instances.</p> <p>snmp—Trace SNMP events.</p>
Required Privilege Level	system-control
Related Documentation	<ul style="list-style-type: none"> • traceoptions (App Engine Virtual Machine Management Service) on page 358 • file (App Engine Virtual Machine Management Service) on page 351

- [level \(App Engine Virtual Machine Management Service\) on page 355](#)

interfaces (Autoinstallation)

Syntax

```
interfaces {  
  interface-name {  
    bootp;  
    rarp;  
  }  
}
```

Hierarchy Level [edit system autoinstallation]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure the interface on which to perform autoinstallation. A request for an IP address is sent from the interface. Specify the IP address procurement protocol.



NOTE: When you run the `system autoinstallation` command, the command will configure unit 0 logical interface for all the active state physical interfaces. However, few commands like `fabric-options` do not allow its physical interface to be configured with a logical interface. If the `system autoinstallation` and the `fabric-options` commands are configured together the following message is displayed incompatible with 'system autoinstallation'.

- Options**
- `bootp`—Enables BOOTP or DHCP during autoinstallation.
 - `rarp`—Enables RARP during autoinstallation.

Required Privilege Level

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

- Related Documentation**
- [Autoinstallation Overview on page 95](#)
 - [Example: Configuring Autoinstallation on SRX Series Devices on page 100](#)

level (App Engine Virtual Machine Management Service)

Syntax	level (all error info notice verbose warning)
Hierarchy Level	[edit system processes app-engine-virtual-machine-management-service traceoptions]
Release Information	Statement introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers. Statement introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers. Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Description	Set level of debugging output.
Default	info
Options	<p>all—Match all levels.</p> <p>error—Match error conditions.</p> <p>info—Match informational messages.</p> <p>notice—Match conditions that must be handled specially.</p> <p>verbose—Match verbose messages.</p> <p>warning—Match warning messages.</p>
Required Privilege Level	system-control
Related Documentation	<ul style="list-style-type: none"> • traceoptions (App Engine Virtual Machine Management Service) on page 358 • flag (App Engine Virtual Machine Management Service) on page 353 • file (App Engine Virtual Machine Management Service) on page 351

license

Syntax

```
license {
  autoupdate {
    url url <password password>;
  }
  keys {
    key key
  }
  renew {
    before-expiration number;
    interval interval-hours;
  }
  traceoptions {
    file {
      filename;
      files number;
      match regular-expression;
      size maximum-file-size;
      (world-readable | no-world-readable);
    }
    flag flag;
    no-remote-trace;
  }
}
```

Hierarchy Level [edit system]

Release Information Statement introduced in Junos OS Release 8.5 for SRX Series and vSRX.
Options **keys** introduced in Junos OS Release 14.1X53-D10.
Statement introduced in Junos OS Release 14.1X53-D10 for EX Series and QFX Series, with option **keys** included.
Statement introduced in Junos OS Release 15.1 for M Series, MX Series, PTX Series, and T Series, with option **keys** included.

Description Specify license information for the device.

Options **autoupdate**—Autoupdate license keys from license servers.

before-expiration *number*—License renewal lead time before expiration, in days.

Range: 0 through 60 days

interval *interval-hours*—License checking interval, in hours.

Range: 1 through 336 hours

keys *key key*—Configure one or more license keys. For example,

```
[edit]
user@device# set system license keys key "key_1"
user@device# set system license keys key "key_2"
user@device# set system license keys key "key_3"
```



```
user@device# set system license keys key "key_4"
user@device# commit
commit complete
```

renew—License renewal lead time and checking interval.

url—URL of a license server.

The remaining statements are explained separately. See [CLI Explorer](#).

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

Related Documentation

traceoptions (App Engine Virtual Machine Management Service)

Syntax `traceoptions {
 file (App Engine Virtual Machine Management Service) filename <files number> | match
 | no-world-readable | <size size> <world-readable >;
 flag (App Engine Virtual Machine Management Service) (all | ccif | configuration | heartbeat
 | init | miscellaneous | platform | pxe | routing-instances | snmp);
 level (App Engine Virtual Machine Management Service) (all | error | info | notice | verbose
 | warning)
 }`

Hierarchy Level [edit system processes app-engine-virtual-machine-management-service]

Release Information Statement introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Statement introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Description Enable traceoptions for the app-engine virtual machine management service system process.

Default Traceoptions are not enabled.

Options **file**—Trace file information.
flag—Perform defined tracing operation.
level—Set traceoptions level.
no-remote-trace—Disable remote tracing.
 The remaining statements are explained separately. See [CLI Explorer](#).

Required Privilege Level system-control

Related Documentation

- *system*
- *processes*
- [level \(App Engine Virtual Machine Management Service\) on page 355](#)
- [flag \(App Engine Virtual Machine Management Service\) on page 353](#)
- [file \(App Engine Virtual Machine Management Service\) on page 351](#)

traceoptions (System License)

Syntax	<pre> traceoptions { file { filename; files number; match regular-expression; size maximum-file-size; (world-readable no-world-readable); } flag flag; no-remote-trace; } </pre>
Hierarchy Level	[edit system license]
Release Information	<p>Statement introduced in Junos OS Release 8.5 for SRX Series and vSRX.</p> <p>Statement introduced in Junos OS Release 14.1X53-D10 for EX Series and QFX Series.</p> <p>Statement introduced in Junos OS Release 15.1 for M Series, MX Series, and T Series.</p>
Description	Set trace options for licenses.
Options	<p>file—Configure the trace file information.</p> <p>filename—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory <code>/var/log</code>. By default, the name of the file is the name of the process being traced.</p> <p>files number—Maximum number of trace files. When a trace file named trace-file reaches its maximum size, it is renamed trace-file.0, then trace-file.1, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.</p> <p>If you specify a maximum number of files, you also must specify a maximum file size with the size maximum file-size option.</p> <p>Range: 2 through 1000 files</p> <p>Default: 10 files</p> <p>match regular-expression—Refine the output to include lines that contain the regular expression.</p> <p>size size—Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). If you specify a maximum file size, you also must specify a maximum number of trace files with the files number option.</p> <p>Range: 10 KB through 1 GB</p> <p>Default: 128 KB</p> <p>world-readable no-world-readable—By default, log files can be accessed only by the user who configures the tracing operation. The world-readable option enables</p>

any user to read the file. To explicitly set the default behavior, use the **no-world-readable** option.

flag flag—Specify which tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements. You can include the following flags.


- **all**—Trace all operations.
- **config**—Trace license configuration processing.
- **events**—Trace licensing events and their processing.

no-remote-trace—Disable the remote tracing.

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

Related Documentation	• license on page 356
------------------------------	---------------------------------------

vmhost

Syntax	<pre>vmhost { management-if no-auto-recovery }</pre>
Hierarchy Level	[edit]
Release Information	<p>Statement introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.</p> <p>Statement introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.</p> <p>Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers</p>
	<div>  NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX. </div>
Description	Configure VM host management properties. Set values in the edit vmhost hierarchy of the configuration.
Options	<p>management-if—Configure management interface properties of the host.</p> <p>no-auto-recovery—Disable the automatic guest recovery by the host.</p>
Required Privilege Level	<p>system—To view this statement in the configuration.</p> <p>system-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX with VM Host Support on page 261 • Disabling Autorecovery on RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines on page 277

vmhost management-if disable

Syntax vmhost management-if disable

Hierarchy Level [edit vmhost]

Release Information Statement introduced in Junos OS Release 15.1F6.



NOTE: The command is supported on the routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines only.

Description	Disable the host interface eth0, which serves as the management port. You can the disable the interface if there are any issues associated with security or any hardware failures either at the local end or the remote end of the interface. if you disable the interface, the transmitter is turned off and the link partner experiences a link-down condition.
--------------------	---


Default The host interface eth0 which serves as the management port is enabled.

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

Related Documentation

- [vmhost on page 361](#)
- [vmhost management-if speed on page 364](#)
- [vmhost management-if link-mode on page 363](#)
- [show vmhost management-if on page 557](#)

vmhost management-if link-mode

Syntax	vmhost management-if link-mode (automatic half-duplex full-duplex)
Hierarchy Level	[edit vmhost]
Release Information	Statement introduced in Junos OS Release 15.1F6.
	<div>  <p>NOTE: The command is supported on the routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines only.</p> </div>
Description	Configure the link mode of the host interface eth0, which serves as the management port as half-duplex or full-duplex. You can also manually select the link mode option as either half-duplex or full-duplex.
Default	The link partners auto-negotiate the speed and duplex link mode and select the highest common capability.
Options	<p>automatic—Autonegotiate the link mode of the management interface. if you set the link mode to automatic, you must also set the link speed to automatic.</p> <p>half-duplex—Set the link mode of the management interface to half-duplex.</p> <p>full-duplex—Set the link mode of the management interface to full-duplex.</p>
Required Privilege Level	<p>system—To view this statement in the configuration.</p> <p>system-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • vmhost on page 361 • vmhost management-if disable on page 362 • vmhost management-if speed on page 364 • show vmhost management-if on page 557

vmhost management-if speed

Syntax `vmhost management-if speed { automatic | 10m | 100m | 1g }`

Hierarchy Level [edit vmhost]

Release Information Statement introduced in Junos OS Release 15.1F6.



NOTE: The command is supported on the routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines only.

Description	Configure the link speed of the host interface eth0, which serves as the management port. If you set the link speed as 10m or 100m , autonegotiation is turned off and the link speed is the speed that you specify.
--------------------	--

Default The link partners auto-negotiate the speed and duplex link mode and select the highest common capability.

Options **automatic**—Autonegotiate the link speed of the management interface. If you set the link speed as **automatic**, speed and link mode are auto-negotiated with the link partner.

10m—Set the link speed of the management interface to 10Mbps.

100m—Set the link speed of the management interface to 100Mbps

1g—Set the link speed of the management interface to 1Gbps. If you set link speed to **1Gbps**, autonegotiation is enabled. However, the interface advertises only 1Gbps speed and full-duplex mode.

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


Related Documentation

- [vmhost on page 361](#)
- [vmhost management-if disable on page 362](#)
- [vmhost management-if link-mode on page 363](#)
- [show vmhost management-if on page 557](#)

usb

Syntax	<pre>usb { disable; }</pre>
Hierarchy Level	[edit system autoinstallation]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Disable the USB autoinstallation process.
Options	disable —Disable the process.
Required Privilege Level	system—To view this statement in the configuration. system-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Example: Configuring Autoinstallation on SRX Series Devices on page 100

vmhost

Syntax	<pre>vmhost { management-if no-auto-recovery }</pre>
Hierarchy Level	[edit]
Release Information	<p>Statement introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.</p> <p>Statement introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.</p> <p>Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers</p>
	<div> NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.</div>
Description	Configure VM host management properties. Set values in the edit vmhost hierarchy of the configuration.
Options	<p>management-if—Configure management interface properties of the host.</p> <p>no-auto-recovery—Disable the automatic guest recovery by the host.</p>
Required Privilege Level	<p>system—To view this statement in the configuration.</p> <p>system-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• RE-MX-X6, RE-MX-X8, RE-PTX-X8 and RCBPTX with VM Host Support on page 261• Disabling Autorecovery on RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines on page 277

CHAPTER 19

Operational Commands

- clear system login lockout
- request system autorecovery state
- request system download abort
- request system download clear
- request system download pause
- request system download resume
- request system download start
- request system firmware upgrade
- request system halt
- request system license add
- request system license delete
- request system license save
- request system license update
- request system partition compact-flash
- request system power-off
- request system reboot
- request system reboot (EX Series)
- request system reboot (Junos OS with Upgraded FreeBSD)
- request system recover
- request system scripts add
- request system scripts delete
- request system scripts rollback
- request system snapshot
- request system snapshot (Junos OS with Upgraded FreeBSD)
- request system snapshot (Maintenance)
- request system software abort in-service-upgrade (ICU)
- request system software add
- request system software add (Maintenance)

- request system software configuration-backup
- request system software configuration-restore
- request system software delete
- request system software rollback
- request system software rollback (SRX Series)
- request system software validate
- request system software validate on (Junos OS with Upgraded FreeBSD)
- request system storage cleanup
- request system storage cleanup (SRX Series)
- request system zeroize
- request vmhost halt
- request vmhost reboot
- show chassis usb storage
- show system autoinstallation status
- show system autorecovery state
- show system boot-messages
- show system auto-snapshot
- show system download
- show system license
- show system license (View)
- show system login lockout
- show system snapshot
- show system snapshot (Junos OS with Upgraded FreeBSD)
- show system snapshot (Upgraded FreeBSD)
- show system snapshot media
- show system storage partitions (EX Series Switches Only)
- show system storage partitions (View SRX Series)
- show version

clear system login logout

Syntax	clear system login logout <all> <user <i>username</i> >
Release Information	Command introduced in Junos OS Release 11.2.
Description	Unlock the user account locked as a result of invalid login attempts.
Options	all —Clear all locked user accounts. user <i>username</i> —Clear the specified locked user account.
Required Privilege Level	clear
Related Documentation	<ul style="list-style-type: none">• <i>lockout-period</i>• show system login logout on page 504
Output Fields	This command produces no output.

request system autorecovery state

Syntax request system autorecovery state (save | recover | clear)

Release Information Command introduced in Junos OS Release 11.2 for SRX300, SRX320, SRX345, and SRX550M devices.

Description Prepare the system for autorecovery of configuration, licenses, and disk information.

Options **save**—Save the current state of the disk partitioning, configuration, and licenses for autorecovery.

The active Junos OS configuration is saved as the Junos rescue configuration, after which the rescue configuration, licenses, and disk partitioning information is saved for autorecovery. Autorecovery information must be initially saved using this command for the autorecovery feature to verify integrity of data on every bootup.



NOTE:

- Any recovery performed at a later stage will restore the data to the same state as it was when the save command was executed.
 - A fresh rescue configuration is generated when the command is executed. Any existing rescue configuration will be overwritten.
-

recover—Recover the disk partitioning, configuration, and licenses.

After autorecovery data has been saved, the integrity of saved items is always checked automatically on every bootup. The recovery command allows you to forcibly re-run the tests at any time if required.

clear—Clear all saved autorecovery information.

Only the autorecovery information is deleted; the original copies of the data used by the router are not affected. Clearing the autorecovery information also disables all autorecovery integrity checks performed during bootup.

Required Privilege Level maintenance

Related Documentation • [show system autorecovery state on page 479](#)

List of Sample Output [request system autorecovery state save on page 371](#)
 [request system autorecovery state recover on page 371](#)
 [request system autorecovery state clear on page 371](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system autorecovery state save

```
user@host> request system autorecovery state save
Saving config recovery information
Saving license recovery information
Saving bsdlablel recovery information
```

Sample Output

request system autorecovery state recover

```
user@host> request system autorecovery state recover

Configuration:
File           Recovery Information  Integrity Check  Action / Status
rescue.conf.gz Saved                 Passed           None
Licenses:
File           Recovery Information  Integrity Check  Action / Status
JUNOS282736.lic Saved                 Passed           None
JUNOS282737.lic Saved                 Failed           Recovered
BSD Labels:
Slice          Recovery Information  Integrity Check  Action / Status
s1             Saved                 Passed           None
s2             Saved                 Passed           None
s3             Saved                 Passed           None
s4             Saved                 Passed           None
```

Sample Output

request system autorecovery state clear

```
user@host> request system autorecovery state clear
Clearing config recovery information
Clearing license recovery information
Clearing bsdlablel recovery information
```

request system download abort

Syntax request system download abort <download-id>

Release Information Command introduced in Junos OS Release 11.2 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 13.2X50-D15 for EX Series switches.

Description Abort a download. The download instance is stopped and cannot be resumed. Any partially downloaded file is automatically deleted to free disk space. Information regarding the download is retained and can be displayed with the **show system download** command until a **request system download clear** operation is performed.



NOTE: Only downloads in the active, paused, and error states can be aborted.

Options download-id—(Required) The ID number of the download to be aborted.

Required Privilege Level maintenance

Related Documentation

- [request system download start on page 376](#)
- [request system download pause on page 374](#)
- [request system download resume on page 375](#)
- [request system download clear on page 373](#)

List of Sample Output [request system download abort on page 372](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system download abort

```
user@host> request system download abort 1
Aborted download #1
```


request system download clear

Syntax	request system download clear
Release Information	Command introduced in Junos OS Release 11.2 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices. Command introduced in Junos OS Release 13.2X50-D15 for EX Series switches.
Description	Delete the history of completed and aborted downloads.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• request system download start on page 376• request system download pause on page 374• request system download resume on page 375• request system download abort on page 372
List of Sample Output	request system download clear on page 373
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system download clear

```
user@host> request system download clear
Cleared information on completed and aborted downloads
```

request system download pause

Syntax request system download pause <download-id>

Release Information Command introduced in Junos OS Release 11.2 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 13.2X50-D15 for EX Series switches.

Description Suspend a particular download instance.



NOTE: Only downloads in the active state can be paused.

Options **download-id**—(Required) The ID number of the download to be paused.

Required Privilege Level maintenance

Related Documentation

- [request system download start on page 376](#)
- [request system download resume on page 375](#)
- [request system download abort on page 372](#)
- [request system download clear on page 373](#)

List of Sample Output [request system download pause on page 374](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system download pause

```
user@host> request system download pause 1
Paused download #1
```

request system download resume

Syntax `request system download resume download-id <max-rate>`

Release Information Command introduced in Junos OS Release 11.2 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 13.2X50-D15 for EX Series switches.

Description Resume a download that has been paused. Download instances that are not in progress because of an error or that have been explicitly paused by the user can be resumed by the user. The file will continue downloading from the point where it paused. By default, the download resumes with the same bandwidth specified with the **request system download start** command. The user can optionally specify a new (maximum) bandwidth with the **request system download resume** command.



NOTE: Only downloads in the paused and error states can be resumed.

Options **download-id**—(Required) The ID number of the download to be resumed.

max-rate—(Optional) The maximum bandwidth for the download.

Required Privilege Level maintenance

Related Documentation

- [request system download start on page 376](#)
- [request system download pause on page 374](#)
- [request system download abort on page 372](#)
- [request system download clear on page 373](#)

List of Sample Output [request system download resume on page 375](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system download resume

```
user@host> request system download resume 1
Resumed download #1
```

request system download start


Syntax	<code>request system download start (<i>sftp-url</i> <i>delay</i> <i>identity-file</i> <i>login</i> <i>max-rate</i> <i>passphrase</i> <i>save as</i>)</code>
Release Information	<p>Command introduced in Junos OS Release 11.2 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.</p> <p>Command introduced in Junos OS Release 13.2X50-D15 for EX Series switches.</p>
Description	Create a download instance and identify it with a unique integer called the download ID.
Options	<p>sftp-url—(Required) The FTP or HTTP URL location of the file to be downloaded securely.</p> <p>delay—(Optional) The number of hours after which the download should start (range from 1 through 48 hours).</p> <p>identity-file—(Required) The name of the file requesting a Secure FTP (SFTP) download. The SFTP in smart download leverages public key authentication to authenticate a download request. Users need to generate a private or public key pair before starting a download, and then upload a public key to an SFTP server.</p> <p>login—(Optional) The username and password for the server in the format username:password.</p> <p>max-rate—(Optional) The maximum average bandwidth for the download. Numbers with the suffix k or K, m or M, and g or G are interpreted as Kbps, Mbps, or Gbps, respectively.</p> <p>passphrase—(Required) The passphrase to protect the private key file stored on the file system. This option does not allow the user to enter a weak passphrase, which ensures stronger security.</p> <p>save-as—(Optional) The filename to be used for saving the file in the <code>/var/tmp</code> location.</p>
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• request system download pause on page 374• request system download resume on page 375• request system download abort on page 372• request system download clear on page 373
List of Sample Output	request system download start on page 377
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system download start

```
user@host> request system download start identity-file mytestkey  
sftp://mysftpserver/homes/kelly/test.tgz max-rate 200 save as newfile.tgz  
Starting download #8
```

request system firmware upgrade

Syntax	<code>request system firmware upgrade</code>
Release Information	Command introduced in Junos OS Release 10.2.
Description	Upgrade firmware on a system.
Options	<p>fpc—Upgrade FPC ROM monitor.</p> <p>pic—Upgrade PIC firmware.</p> <p>re—Upgrade baseboard BIOS/FPGA. There is an active BIOS image and a backup BIOS image.</p> <ul style="list-style-type: none">• bios—(Optional) Upgrade BIOS.• fpga—(Optional) Upgrade baseboard FPGA.• ssd—(Optional) Upgrade Routing Engine solid-state drive (SSD) firmware. <p>disk1—Upgrade SSD disk1 firmware.</p> <p>disk2—Upgrade SSD disk2 firmware.</p> <div><div></div><div><p>NOTE: Starting in Junos OS Release 17.2R1, you can upgrade the SSD firmware on MX Series routers with the RE-S-X6-64G and RE-MX2K-X8-64G Routing Engines.</p></div></div>
	<p>vcpu—Upgrade VCPU ROM monitor.</p>
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• request system license update on page 385
List of Sample Output	request system firmware upgrade on page 378
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system firmware upgrade

```
user@host> request system firmware upgrade re bios
```

Part	Type	Tag	Current version	Available version	Status
Routing Engine 0	RE BIOS	0	1.5	1.9	OK
Routing Engine 0	RE BIOS Backup	1	1.7	1.9	OK

Perform indicated firmware upgrade ? [yes,no] (no) yes

user@host> request system firmware upgrade re bios backup

Part	Type	Tag	Current version	Available version	Status
Routing Engine 0	RE BIOS	0	1.5	1.9	OK
Routing Engine 0	RE BIOS Backup	1	1.7	1.9	OK

Perform indicated firmware upgrade ? [yes,no] (no) yes

user@host> request system firmware upgrade re ssd disk1

Part	Type	Tag	Current version	Available version	Status
Routing Engine 0	RE SSD1	4	12028	12029	OK

Perform indicated firmware upgrade ? [yes,no] (no) yes

Firmware upgrade initiated, use "show system firmware" to monitor status.

request system halt

Syntax	<code>request system halt</code> <code>at <time></code> <code>in <minutes></code> <code>media (compact-flash disk usb)</code> <code>messages <message></code>
Release Information	Command introduced in Junos OS Release 11.4.
Description	Stop the system.
Options	<p>at <i>time</i>— Time at which to stop the system.</p> <p>in <i>minutes</i>— Number of minutes to delay before halting the system.</p> <p>media —Boot media for the next boot.</p> <ul style="list-style-type: none">• compact-flash— Standard boot from a flash device.• disk— Boot from a hard disk.• usb— Boot from a USB device. <p>message <i>message</i>— Message that is displayed to all system users before stopping the system.</p>
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• request system power-off on page 387
List of Sample Output	request system halt on page 380
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system halt

```
user@host> request system halt
Halt the system ? [yes,no] (no) yes

*** FINAL System shutdown message from root@quickland ***

System going down IMMEDIATELY

Shutdown NOW!
```



```
[pid 7560]

root@quickland> Dec  8 08:57:37 Waiting (max 60 seconds) for system process `vnlru'
to stop...done
Waiting (max 60 seconds) for system process `vnlru_mem' 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...2 2 2 2 2 2 2 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 done

syncing disks... All buffers synced.
Uptime: 2d16h25m9s
recorded reboot as normal shutdown

The operating system has halted.
Please press any key to reboot.
```

request system license add

Syntax	<code>request system license add (<i>filename</i> terminal)</code>
Release Information	Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 9.0 for EX Series switches. Command introduced in Junos OS Release 9.5 for SRX Series devices. Command introduced in Junos OS Release 11.1 for the QFX Series.
Description	Add a license key.
Options	<i>filename</i> —License key from a file or URL. Specify the filename or the URL where the key is located. <i>terminal</i> —License key from the terminal.
Required Privilege Level	maintenance
Related Documentation	
List of Sample Output	request system license add on page 382
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system license add

```
user@host> request system license add terminal
E408408918 aeaqib qcsbj a okbuqe rcmxnq vjocwf uxfsta
          z5ufjb kdrmt6 57bimv 2f3ddp qttcdn 627q4a
          jx4s5x hiri
E408408918: successfully added
add license complete (no errors)
```

request system license delete

Syntax	<code>request system license delete (<i>license-identifier</i> license-identifier-list [<i>licenseid001</i> <i>licenseid002</i> <i>licenseid003</i>] all)</code>
Release Information	<p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>Command introduced in Junos OS Release 11.1 for the QFX Series.</p> <p>Option license-identifier-list introduced in Junos OS Release 13.1.</p>
Description	Delete a license key. You can choose to delete one license at a time, all licenses at once, or a list of license identifiers enclosed in brackets.
Options	<p><i>license-identifier</i>—Text string that uniquely identifies a license key.</p> <p>license-identifier-list [<i>licenseid001</i> <i>licenseid002</i> <i>licenseid003</i>....]—Delete multiple license identifiers as a list enclosed in brackets.</p> <p>all—Delete all licenses on the device.</p>
Required Privilege Level	maintenance
Related Documentation	

request system license save


Syntax	<code>request system license save (<i>filename</i> terminal)</code>
Release Information	Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 9.0 for EX Series switches. Command introduced in Junos OS Release 11.1 for the QFX Series. Command introduced in Junos OS Release 9.5 for SRX Series devices.
Description	Save installed license keys to a file or URL.
Options	<i>filename</i> —License key from a file or URL. Specify the filename or the URL where the key is located. terminal —License key from the terminal.
Required Privilege Level	maintenance
Related Documentation	
List of Sample Output	request system license save on page 384
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system license save

```
user@host> request system license save ftp://user@host/license.conf
```

request system license update

Syntax	<code>request system license update</code>
Release Information	Command introduced in Junos OS Release 9.5.
Description	If your device supports initial install from the EMS server in Products Supporting Juniper Agile Licensing, you can use this command to install all licenses from the EMS server. You can also autoupdate license keys from the LMS or EMS server.
	<div>  NOTE: The <code>request system license update</code> command always uses the default Juniper license server: <ul style="list-style-type: none"> For Juniper Agile Licensing (JAL) keys: https://license.juniper.net/ For non-JAL keys: https://ae1.juniper.net </div>
Options	<code>trial</code> —(For non-Juniper Agile Licensing keys only) Immediately updates trial license keys from the LMS server.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none"> <i>Example: Configuring a Proxy Server for License Updates</i>
List of Sample Output	request system license update on page 385 request system license update trial on page 385
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system license update

```
user@host> request system license update
```


```
Trying to update license keys from https://ae1.juniper.net has been sent, use
show system license to check status.
```

request system license update trial

```
user@host> request system license update trial
```

```
Request to automatically update trial license keys from https://ae1.juniper.net
has been sent, use show system license to check status.
```

request system partition compact-flash

Syntax	request system partition compact-flash
Release Information	Command introduced in Junos OS Release 9.2. Command deprecated for Junos OS with Upgraded FreeBSD in Junos OS Release 15.1.
	<div> NOTE: To determine which platforms run Junos OS with Upgraded FreeBSD, see the table listing the platforms currently running Junos OS with upgraded FreeBSD in “Understanding Junos OS with Upgraded FreeBSD” on page 17.</div>
Description	Reboots the device and repartitions the compact flash. The CompactFlash card is repartitioned only if it is possible to restore all the data on the CompactFlash card. Otherwise, the operation is aborted, and a message is displayed indicating that the current disk usage needs to be reduced.
Required Privilege Level	maintenance
List of Sample Output	request system partition compact-flash (If Yes) on page 386 request system partition compact-flash (If No) on page 386
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system partition compact-flash (If Yes)

```
user@host> request system partition compact-flash
Are you sure you want to reboot
and partition the compact-flash ? [yes,no] yes
Initiating repartition operation.
The operation may take several minutes to complete.
System will reboot now...
<System reboots>
<Repartition operation is performed>
<System reboots and starts up normally>
```

Sample Output

request system partition compact-flash (If No)

```
user@host> request system partition compact-flash
Are you sure you want to reboot
and partition the compact-flash ? [yes,no] no
```

request system power-off

Syntax	request system power-off at <i><time></i> in <i><minutes></i> media (compact-flash disk usb) messages <i><message></i>
Release Information	Command introduced in Junos OS Release 11.4.
Description	Power off the system.
Options	<p>at <i>time</i>— Time at which to power off the system.</p> <p>in <i>minutes</i>— Number of minutes to delay before powering off the system.</p> <p>media —Boot media for the next boot.</p> <ul style="list-style-type: none"> • compact-flash— Standard boot from a flash device. • disk— Boot from a hard disk. • usb— Boot from a USB device. <p>message <i>message</i>— Message that is displayed to all system users before powering off the system.</p>
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none"> • request system halt on page 380
List of Sample Output	request system power-off on page 387
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system power-off

```

user@host> request system power-off
Power Off the system ? [yes,no] (no) yes

Shutdown NOW!
[pid 3300]

*** FINAL System shutdown message from root@quickland ***

System going down IMMEDIATELY

```

```
root@quickland> Dec  8 09:37:45 Waiting (max 60 seconds) for system process `vnlru'
to stop...done
Waiting (max 60 seconds) for system process `vnlru_mem' 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...2 2 2 2 2 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 done

syncing disks... All buffers synced.
Uptime: 38m33s
recorded reboot as normal shutdown

The operating system has halted.
Turning the system power off.
```


request system reboot

List of Syntax	Syntax on page 389 Syntax (EX Series Switches) on page 389 Syntax (TX Matrix Router) on page 389 Syntax (TX Matrix Plus Router) on page 389 Syntax (MX Series Router) on page 389
Syntax	<pre>request system reboot <at <i>time</i>> <both-routing-engines> <in <i>minutes</i>> <media (compact-flash disk removable-compact-flash usb)> <message "<i>text</i>"> <other-routing-engine></pre>
Syntax (EX Series Switches)	<pre>request system reboot <all-members> <at <i>time</i>> <both-routing-engines> <in <i>minutes</i>> <local> <media (external internal)> <member <i>member-id</i>> <message "<i>text</i>"> <other-routing-engine> <slice <i>slice</i>></pre>
Syntax (TX Matrix Router)	<pre>request system reboot <all-chassis all-lcc lcc <i>number</i> scc> <at <i>time</i>> <both-routing-engines> <in <i>minutes</i>> <media (compact-flash disk)> <message "<i>text</i>"> <other-routing-engine></pre>
Syntax (TX Matrix Plus Router)	<pre>request system reboot <all-chassis all-lcc lcc <i>number</i> sfc <i>number</i>> <at <i>time</i>> <both-routing-engines> <in <i>minutes</i>> <media (compact-flash disk)> <message "<i>text</i>"> <other-routing-engine> <partition (1 2 alternate)></pre>
Syntax (MX Series Router)	<pre>request system reboot <all-members> <at <i>time</i>> <both-routing-engines></pre>

<in *minutes*>
<local>
<media (external | internal)>
<member *member-id*>
<message "*text*">
<other-routing-engine>

Release Information Command introduced before Junos OS Release 7.4.
Option **other-routing-engine** introduced in Junos OS Release 8.0.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Option **sfc** introduced for the TX Matrix Plus router in Junos OS Release 9.6.
Option **both-routing-engines** introduced in Junos OS Release 12.1.

Description Reboot the software.



NOTE: Starting with Junos OS Release 15.1F3, the statement **request system reboot** reboots only the guest operating system on the PTX5000 with RE-PTX-X8-64G and, MX240, MX480, and MX960 with RE-S-X6-64G.

Starting with Junos OS Release 15.1F5, the statement **request system reboot** reboots only the guest operating system on the MX2010, and MX2020 with REMX2K-X8-64G.

Options **none**—Reboot the software immediately.

all-chassis—(TX Matrix routers and TX Matrix Plus routers only) (Optional) On a TX Matrix router or TX Matrix Plus router, reboot all routers connected to the TX Matrix or TX Matrix Plus router, respectively.

all-lcc—(TX Matrix routers and TX Matrix Plus routers only) (Optional) On a TX Matrix router or TX Matrix Plus router, reboot all line card chassis connected to the TX Matrix or TX Matrix Plus router, respectively.

all-members—(EX4200 switches and MX Series routers only) (Optional) Reboot the software on all members of the Virtual Chassis configuration.

at time—(Optional) Time at which to reboot the software, specified in one of the following ways:

- **now**—Stop or reboot the software immediately. This is the default.
- **+minutes**—Number of minutes from now to reboot the software.
- **yymmddhhmm**—Absolute time at which to reboot the software, specified as year, month, day, hour, and minute.
- **hh:mm**—Absolute time on the current day at which to stop the software, specified in 24-hour time.

both-routing-engines—(Optional) Reboot both Routing Engines at the same time.

in *minutes*—(Optional) Number of minutes from now to reboot the software. This option is an alias for the **at +*minutes*** option.

lcc *number*—(TX Matrix routers and TX Matrix Plus routers only) (Optional) Line-card chassis number.

Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.
- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.
- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

local—(EX4200 switches and MX Series routers only) (Optional) Reboot the software on the local Virtual Chassis member.

media (compact-flash | disk)—(Optional) Boot medium for next boot.

media (external | internal)—(EX Series switches and MX Series routers only) (Optional) Reboot the boot media:

- **external**—Reboot the external mass storage device.
- **internal**—Reboot the internal flash device.

member *member-id*—(EX4200 switches and MX Series routers only) (Optional) Reboot the software on the specified member of the Virtual Chassis configuration. For EX4200 switches, replace *member-id* with a value from 0 through 9. For an MX Series Virtual Chassis, replace *member-id* with a value of 0 or 1.

message "*text*"—(Optional) Message to display to all system users before stopping or rebooting the software.

other-routing-engine—(Optional) Reboot the other Routing Engine from which the command is issued. For example, if you issue the command from the master Routing Engine, the backup Routing Engine is rebooted. Similarly, if you issue the command from the backup Routing Engine, the master Routing Engine is rebooted.

partition—(TX Matrix Plus routers only) (Optional) Reboot using the specified partition on the boot media. This option has the following suboptions:

- **1**—Reboot from partition 1.
- **2**—Reboot from partition 2.

- **alternate**—Reboot from the alternate partition.

scc—(TX Matrix routers only) (Optional) Reboot the Routing Engine on the TX Matrix switch-card chassis. If you issue the command from `re0`, `re0` is rebooted. If you issue the command from `re1`, `re1` is rebooted.

sfc number—(TX Matrix Plus routers only) (Optional) Reboot the Routing Engine on the TX Matrix Plus switch-fabric chassis. If you issue the command from `re0`, `re0` is rebooted. If you issue the command from `re1`, `re1` is rebooted. Replace *number* with 0.

slice slice—(EX Series switches only) (Optional) Reboot a partition on the boot media. This option has the following suboptions:

- 1—Power off partition 1.
- 2—Power off partition 2.
- **alternate**—Reboot from the alternate partition.

Additional Information Reboot requests are recorded in the system log files, which you can view with the **show log** command (see *show log*). Also, the names of any running processes that are scheduled to be shut down are changed. You can view the process names with the **show system processes** command (see *show system processes*).

On a TX Matrix or TX Matrix Plus router, if you issue the **request system reboot** command on the master Routing Engine, all the master Routing Engines connected to the routing matrix are rebooted. If you issue this command on the backup Routing Engine, all the backup Routing Engines connected to the routing matrix are rebooted.



NOTE: Before issuing the **request system reboot** command on a TX Matrix Plus router with no options or the **all-chassis**, **all-lcc**, **lcc number**, or **sfc** options, verify that master Routing Engine for all routers in the routing matrix are in the same slot number. If the master Routing Engine for a line-card chassis is in a different slot number than the master Routing Engine for a TX Matrix Plus router, the line-card chassis might become logically disconnected from the routing matrix after the **request system reboot** command.



NOTE: To reboot a router that has two Routing Engines, reboot the backup Routing Engine (if you have upgraded it) first, and then reboot the master Routing Engine.

Required Privilege Level maintenance

Related Documentation

- *clear system reboot*
- *request system halt*
- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)
- [request vmhost reboot on page 474](#)

List of Sample Output

- [request system reboot on page 393](#)
- [request system reboot \(at 2300\) on page 393](#)
- [request system reboot \(in 2 Hours\) on page 393](#)
- [request system reboot \(Immediately\) on page 393](#)
- [request system reboot \(at 1:20 AM\) on page 393](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system reboot

```
user@host> request system reboot
Reboot the system ? [yes,no] (no)
```

request system reboot (at 2300)

```
user@host> request system reboot at 2300 message ?Maintenance time!?
Reboot the system ? [yes,no] (no) yes

shutdown: [pid 186]
*** System shutdown message from root@test.example.net ***
System going down at 23:00
```

request system reboot (in 2 Hours)

The following example, which assumes that the time is 5 PM (17:00), illustrates three different ways to request the system to reboot in two hours:

```
user@host> request system reboot at +120
user@host> request system reboot in 120
user@host> request system reboot at 19:00
```

request system reboot (Immediately)

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

request system reboot (at 1:20 AM)

To reboot the system at 1:20 AM, enter the following command. Because 1:20 AM is the next day, you must specify the absolute time.

```
user@host> request system reboot at 06060120
request system reboot at 120
Reboot the system at 120? [yes,no] (no) yes
```


request system reboot (EX Series)

Syntax	<pre>request system reboot <all-members local member member-id> <at time> <in minutes> <media (external internal)> <message "text"> <slice (1 2 alternate)></pre>
Release Information	<p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>Option partition changed to slice in Junos OS Release 10.0 for EX Series switches.</p>
Description	<p>Reboot the Junos OS.</p> <p>Reboot requests are recorded in the system log files, which you can view with the show log command. You can view the process names with the show system processes command.</p>
Options	<p>none—Reboots the software immediately.</p> <p>all-members local member member-id—(Optional) Specify which member of the Virtual Chassis to reboot:</p> <ul style="list-style-type: none"> • all-members—Reboots each switch that is a member of the Virtual Chassis. • local—Reboots the local switch, meaning the switch you are logged into, only. • member member-id—Reboots the specified member switch of the Virtual Chassis. <p>at time—(Optional) Time at which to reboot the software, specified in one of the following ways:</p> <ul style="list-style-type: none"> • +minutes—Number of minutes from now to reboot the software. • hh:mm—Absolute time on the current day at which to reboot the software, specified in 24-hour time. • now—Stop or reboot the software immediately. This is the default. • yymmddhhmm—Absolute time at which to reboot the software, specified as year, month, day, hour, and minute. <p>in minutes—(Optional) Number of minutes from now to reboot the software. This option is an alias for the at +minutes option.</p> <p>media (external internal)—(Optional) Boot medium for the next boot. The external option reboots the switch using a software package stored on an external boot source, such as a USB flash drive. The internal option reboots the switch using a software package stored in an internal memory source.</p>

message "text"—(Optional) Message to display to all system users before rebooting the software.

slice (1 | 2 | alternate)—(Optional) Reboot using the specified partition on the boot media. This option has the following suboptions:

- **1**—Reboot from partition 1.
- **2**—Reboot from partition 2.
- **alternate**—Reboot from the alternate partition, which is the partition that did not boot the switch at the last bootup.

Required Privilege Level maintenance

Related Documentation

- *clear system reboot*

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system reboot

```
user@host> request system reboot
Reboot the system ? [yes,no] (no)
```

request system reboot (at 2300)

```
user@host> request system reboot at 2300 message ?Maintenance time!?
Reboot the system ? [yes,no] (no) yes
```

```
shutdown: [pid 186]
*** System shutdown message from root@berry.network.net ***
System going down at 23:00
```

request system reboot (in 2 Hours)

The following example, which assumes that the time is 5 PM (17:00), illustrates three different ways to request the system to reboot in two hours:

```
user@host> request system reboot at +120
user@host> request system reboot in 120
user@host> request system reboot at 19:00
```

request system reboot (Immediately)

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

request system reboot (at 1:20 AM)

To reboot the system at 1:20 AM, enter the following command. Because 1:20 AM is the next day, you must specify the absolute time.


```
user@host> request system reboot at 06060120
request system reboot at 120
Reboot the system at 120? [yes,no] (no) yes
```

request system reboot (Junos OS with Upgraded FreeBSD)

Syntax	<pre>request system reboot <at <i>time</i>> <both-routing-engines> <in <i>minutes</i>> <junos> <message "<i>text</i>"> <network> <oam> <other-routing-engine> <usb></pre>
Release Information	Command introduced in Junos OS Release 15.1 for MX240, MX480, MX960, MX2010, and MX2020 routers and EX9200 switches. Command introduced in Junos OS Release 15.1X53-D30 for QFX5200 switches.
Description	Reboot the software.
Options	<p>none—Reboot the software immediately.</p> <p>at <i>time</i>—(Optional) Time at which to reboot the software, specified in one of the following ways:</p> <ul style="list-style-type: none">• now—Stop or reboot the software immediately. This is the default.• +<i>minutes</i>—Number of minutes from now to reboot the software.• <i>yymmddhhmm</i>—Absolute time at which to reboot the software, specified as year, month, day, hour, and minute. Omitting a value will default to the current date for that value.• <i>hh:mm</i>—Absolute time on the current day at which to stop the software, specified in 24-hour time. <p>both-routing-engines—(Optional) Reboot both Routing Engines at the same time.</p> <p>in <i>minutes</i>—(Optional) Number of minutes from now to reboot the software. This option is an alias for the at +<i>minutes</i> option.</p> <p>junos—(Optional) Reboot from the junos volume.</p> <p>message "<i>text</i>"—(Optional) Message to display to all system users before stopping or rebooting the software.</p> <p>network—(Optional) Reboot from the network.</p> <p>oam—(Optional) Reboot from the oam volume.</p> <p>other-routing-engine—(Optional) Reboot the other Routing Engine from which the command is issued. For example, if you issue the command from the master Routing</p>

Engine, the backup Routing Engine is rebooted. Similarly, if you issue the command from the backup Routing Engine, the master Routing Engine is rebooted.

usb—(Optional) Reboot from the USB device.

Additional Information Reboot requests are recorded in the system log files, which you can view with the **show log** command (see *show log*). Also, the names of any running processes that are scheduled to be shut down are changed. You can view the process names with the **show system processes** command (see *show system processes*).



NOTE: To reboot a router or switch that has two Routing Engines, reboot the backup Routing Engine (if you have upgraded it) first, and then reboot the master Routing Engine.

Required Privilege Level maintenance

Related Documentation

- [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 414](#)
- [show system snapshot \(Junos OS with Upgraded FreeBSD\) on page 508](#)
- *clear system reboot*
- *request system halt*
- [Understanding Junos OS with Upgraded FreeBSD on page 17](#)

List of Sample Output

- [request system reboot on page 399](#)
- [request system reboot \(at 2300\) on page 399](#)
- [request system reboot \(in 2 Hours\) on page 400](#)
- [request system reboot \(Immediately\) on page 400](#)
- [request system reboot \(at 1:20 AM\) on page 400](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system reboot

```
user@host> request system reboot
Reboot the system ? [yes,no] (no)
```

request system reboot (at 2300)

```
user@host> request system reboot at 2300 message "Maintenance time!"
Reboot the system ? [yes,no] (no) yes

shutdown: [pid 186]
```

```
*** System shutdown message from root@berry.network.net ***  
System going down at 23:00
```

request system reboot (in 2 Hours)

The following example, which assumes that the time is 5 PM (17:00), illustrates three different ways to request the system to reboot in two hours:

```
user@host> request system reboot at +120  
user@host> request system reboot in 120  
user@host> request system reboot at 19:00
```

request system reboot (Immediately)

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

request system reboot (at 1:20 AM)

To reboot the system at 1:20 AM, enter the following command. Because 1:20 AM is the next day, you must specify the absolute time.

```
user@host> request system reboot at 06060120  
request system reboot at 120  
Reboot the system at 120? [yes,no] (no) yes
```

request system recover

Syntax	<code>request system recover (junos-volume oam-volume)</code>
Release Information	<p>Command introduced in Junos OS Release 15.1 for MX240, MX480, MX960, MX2010, and MX2020 routers and EX9200 switches only.</p> <p>Command introduced in Junos OS Release 15.1X53-D30 for QFX5200 switches.</p> <p>Command introduced in Junos OS Release 15.1X53-D60 for QFX10000 switches.</p> <p>Command introduced in Junos OS Release 16.1R1 for VM host on MX240, MX480, MX960, MX2010, and MX2020 routers and PTX5000 routers.</p> <p>Command introduced in Junos OS Release 17.3 for SRX5400, SRX5600, and SRX5800 devices.</p>
Description	Recover a specified volume of the system.
Options	<p>junos-volume—Specify the <code>/junos</code> volume to be recovered. The <code>/junos</code> volume is the main drive and contains all the software and files needed for the day-to-day running of the device, including configuration information and logs. The <code>/junos</code> volume also contains non-recovery snapshots, which are new with Junos OS with upgraded FreeBSD. Non-recovery snapshots cannot be used for recovery of a failed system.</p> <p>oam-volume—Specify the <code>/oam</code> volume to be recovered. The compact flash drive is the <code>/oam</code> volume and stores recovery snapshot backup information. In case of failure of the <code>/junos</code> volume, the <code>/oam</code> volume can be used to boot the system. The <code>/oam</code> volume has the recovery snapshot, which is created with the request system snapshot recovery command. A recovery snapshot is always replaced when a new recovery snapshot is taken.</p>
Additional Information	<p>If you try to recover the junos volume while you are booted on the junos volume, you will get an error message.</p> <p>To recover the junos volume, do the following:</p> <ol style="list-style-type: none"> 1. Recover the oam volume. <pre>user@host> request system recover oam</pre> 2. Reboot on the oam volume. <pre>user@host> request system reboot oam</pre>
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • Understanding Junos OS with Upgraded FreeBSD on page 17 • Understanding Junos OS with Upgraded FreeBSD Disk Volumes on page 25 • Understanding Junos OS with Upgraded FreeBSD Snapshots on page 23

List of Sample Output [request system recover junos-volume \(While booted on the junos volume\) on page 402](#)
[request system recover junos-volume \(While booted on the oam volume\) on page 402](#)
[request system recover oam-volume on page 402](#)

Sample Output

[request system recover junos-volume \(While booted on the junos volume\)](#)

```
user@host> request system recover junos-volume
ERROR: You are currently running on the Junos volume
ERROR: A recovery of the Junos volume is not possible
```

[request system recover junos-volume \(While booted on the oam volume\)](#)

```
user@host> request system recover junos-volume
NOTICE: Recovering the Junos volume ...
ada0p3 deleted
ada0 created
ada0p1 added
bootcode written to ada0
ada0p3 added
ada0p2 added
/dev/gpt/junos: 20303.9MB (41582448 sectors) block size 32768, fragment size 4096

        using 33 cylinder groups of 626.22MB, 20039 blks, 80256 inodes.
super-block backups (for fsck_ffs -b #) at:
 192, 1282688, 2565184, 3847680, 5130176, 6412672, 7695168, 8977664, 10260160,
11542656, 12825152, 14107648, 15390144, 16672640, 17955136, 19237632,
20520128, 21802624, 23085120, 24367616, 25650112, 26932608, 28215104,
29497600, 30780096, 32062592, 33345088, 34627584, 35910080, 37192576,
38475072, 39757568, 41040064
NOTICE: Junos volume recovered
```

[request system recover oam-volume](#)

```
user@host> request system recover oam-volume
NOTICE: Recovering the OAM volume ...
ada1p2 deleted
ada1 created
ada1p1 added
bootcode written to ada1
ada1p2 added
/dev/gpt/oam: 3831.6MB (7847136 sectors) block size 32768, fragment size 4096
        using 7 cylinder groups of 626.09MB, 20035 blks, 80256 inodes.
super-block backups (for fsck_ffs -b #) at:
 192, 1282432, 2564672, 3846912, 5129152, 6411392, 7693632
Verified oam signed by PackageProductionEc_2017 method ECDSA256+SHA256
Installing OAM volume contents ...
The OAM volume is now installed
NOTICE: Creating a recovery snapshot on the OAM volume ...
Creating image ...
Compressing image ...
Image size is 1717MB
Recovery snapshot created successfully
NOTICE: OAM volume recovered
```

request system scripts add

Syntax `request system scripts add <package-name>`
`<no-copy>`
`<unlink>`

Release Information Command introduced before Junos OS Release 9.0.

Description CLI command to install AI-Script (jais) packages on Juniper Networks devices.

Options `no-copy`—Don't save a copy of the jais package file.

```
user@host> request system scripts add no-copy <package-name>
```



NOTE: If you use the `no-copy` option during the jais installation, the jais package cannot be rolled back.

`unlink`—Remove the package after successful installation.

```
user@host> request system scripts add unlink <package-name>
```

Required Privilege Level maintenance

Related Documentation

- [request system scripts delete on page 404](#)
- [request system scripts rollback on page 405](#)
- [request system scripts event-scripts](#)

request system scripts delete

Syntax `request system scripts delete <package-name>`

Release Information Command introduced before Junos OS Release 9.0.

Description CLI command to delete AI-Script (jais) packages on Juniper Networks devices.

Options No options are available.

Required Privilege Level maintenance

Related Documentation

- [request system scripts add on page 403](#)
- [request system scripts rollback on page 405](#)
- *request system scripts event-scripts*

request system scripts rollback

Syntax	request system scripts rollback
Release Information	Command introduced before Junos OS Release 9.0.
Description	Attempt to roll back to most recent installation of AI-Scripts (jais) package.
Options	No options are available.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• request system scripts add on page 403• request system scripts delete on page 404• <i>request system scripts event-scripts</i>

request system snapshot

List of Syntax	Syntax on page 406 Syntax (ACX Series Routers) on page 406 Syntax (EX Series Switches) on page 406 Syntax (MX Series Routers) on page 406 Syntax (TX Matrix Routers) on page 406 Syntax (TX Matrix Plus Routers) on page 406
Syntax	request system snapshot <partition>
Syntax (ACX Series Routers)	request system snapshot <media <i>type</i> > <partition>
Syntax (EX Series Switches)	request system snapshot <all-members local member <i>member-id</i> > <media <i>type</i> > <partition> <re0 re1 routing-engine <i>routing-engine-id</i> > <slice alternate>
Syntax (MX Series Routers)	request system snapshot <all-members> <config-partition> <local> <member <i>member-id</i> > <media <i>usb-port-number</i> > <partition> <root-partition>
Syntax (TX Matrix Routers)	request system snapshot <all-chassis all-lcc lcc <i>number</i> scc> <config-partition> <partition> <root-partition>
Syntax (TX Matrix Plus Routers)	request system snapshot <all-chassis all-lcc lcc <i>number</i> sfc <i>number</i> > <config-partition> <partition> <root-partition>
Release Information	Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 10.0 for EX Series switches. Command introduced in Junos OS Release 12.2 for ACX Series routers. Options <config-partition> and <root-partition> introduced in Junos OS Release 13.1 for M Series, MX Series, T Series, and TX Series routers.

Option **media usb-port-number** introduced in Junos OS Release 13.2 for MX104 routers. Options **<config-partition>**, **<root-partition>**, and **<slice>** deprecated for Junos OS with Upgraded FreeBSD in Junos OS Release 15.1



NOTE: To determine which platforms run Junos OS with Upgraded FreeBSD, see the table listing the platforms currently running Junos OS with upgraded FreeBSD in “[Understanding Junos OS with Upgraded FreeBSD](#)” on page 17.

- Description**
- On the router, back up the currently running and active file system partitions 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 the router's flash drive, and the **/altroot** and **/altconfig** file systems are on the router's hard drive.
 - On the switch, take a snapshot of the files currently used to run the switch—the complete contents of the root (**/**), **/altroot**, **/config**, **/var**, and **/var-tmp** directories, which include the running Junos OS, the active configuration, and log files.



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.



NOTE: Starting with Junos OS Release 15.1F3, the statement **request system snapshot** creates a snapshot of the guest OS image only for the PTX5000 with RE-DUO-C2600-16G, and the MX240, MX480, and MX960 routers with RE-S-1800X4-32G-S.

Starting with Junos OS Release 15.1F5, the statement **request system snapshot** creates a snapshot of the guest OS image only for the MX2010 and MX2020 routers with REMX2K-1800-32G-S.

On these routers, in order to create snapshot of the host OS image along with Junos OS image, use the **request vmhost snapshot** command.

Options The specific options available depend upon the router or switch:

none—Back up the currently running software as follows:

- On the router, back up the currently running and active file system partitions 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 the router's flash drive, and the `/altroot` and `/altconfig` file systems are on the router's hard drive.

- On the switch, take a snapshot of the files currently used to run the switch and copy them to the media that the switch did not boot from. If the switch is booted from internal media, the snapshot is copied to external (USB) media. If the switch is booted from external (USB) media, the snapshot is copied to internal media.
- If the snapshot destination is external media but a USB flash drive is not connected, an error message is displayed.
- If the automatic snapshot procedure is already in progress, the command returns the following error: **Snapshot already in progress. Cannot start manual snapshot.** For additional information about the automatic snapshot feature, see [“Understanding Resilient Dual-Root Partitions on Switches” on page 111.](#)

all-chassis | all-lcc | lcc *number* —(TX Matrix and TX Matrix Plus router only) (Optional)

- **all-chassis**—On a TX Matrix router, archive data and executable areas for all Routing Engines in the chassis. On a TX Matrix Plus router, archive data and executable areas for all Routing Engines in the chassis.
- **all-lcc**—On a TX Matrix router, archive data and executable areas for all T640 routers (or line-card chassis) connected to a TX Matrix router. On a TX Matrix Plus router, archive data and executable areas for all routers (or line-card chassis) connected to a TX Matrix Plus router.
- **lcc *number***—On a TX Matrix router, archive data and executable areas for a specific T640 router (or line-card chassis) that is connected to a TX Matrix router. On a TX Matrix Plus router, archive data and executable areas for a specific router (line-card chassis) that is connected to a TX Matrix Plus router.

Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.
- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.
- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

all-members | local | member *member-id*—(EX Series switch Virtual Chassis and MX Series routers only) (Optional) Specify where to place the snapshot (archive data and executable areas) in a Virtual Chassis:

- **all-members**—Create a snapshot (archive data and executable areas) for all members of the Virtual Chassis.

- **local**—Create a snapshot (archive data and executable areas) on the member of the Virtual Chassis that you are currently logged into.
- **member *member-id***—Create a snapshot (archive data and executable areas) for the specified member of the Virtual Chassis.

config-partition—(M, MX, T, TX Series routers only) Create a snapshot of the configuration partition only and store it onto the default **/altconfig** on the hard disk device or an **/altconfig** on a USB device. Option deprecated for Junos OS with Upgraded FreeBSD in Junos OS Release 15.1.



NOTE: To determine which platforms run Junos OS with Upgraded FreeBSD, see the table listing the platforms currently running Junos OS with upgraded FreeBSD in “[Understanding Junos OS with Upgraded FreeBSD](#)” on page 17.

media type—(ACX Series, M320, T640, MX960 routers, and EX Series switches only)(Optional) Specify the boot device the software is copied to:

- **compact-flash**—Copy software to the primary compact flash drive.
- **external**—(Switches only) Copy software to an external mass storage device, such as a USB flash drive. If a USB drive is not connected, the switch displays an error message.
- **internal**—Copy software to an internal flash drive.
- **removable-compact-flash**—Copy software to the removable compact flash drive.
- **usb**—(ACX Series, M320, T640, MX960 routers only) Copy software to the device connected to the USB port.
- **usb0**—(MX104 routers only) Copy software to the device connected to the USB0 port.
- **usb1**—(MX104 routers only) Copy software to the device connected to the USB1 port.

partition—(Optional) Repartition the flash drive before a snapshot occurs. If the partition table on the flash drive is corrupted, the **request system snapshot** command fails and reports errors. The partition option is only supported for restoring the software image from the hard drive to the flash drive.

(Routers only) You cannot issue the **request system snapshot** command when you enable flash disk mirroring. We recommend that you disable flash disk mirroring when you upgrade or downgrade the software. For more information, see the *Junos OS Administration Library*.

(EX Series switches only) If the snapshot destination is the media that the switch did not boot from, you must use the **partition** option.

re0 | re1 | routing-engine *routing-engine-id*—(EX6200 and EX8200 switches only) Specify where to place the snapshot in a redundant Routing Engine configuration.

- **re0**—Create a snapshot on Routing Engine 0.
- **re1**—Create a snapshot on Routing Engine 1.
- **routing-engine *routing-engine-id***—Create a snapshot on the specified Routing Engine.

root-partition—(M, MX, T, TX Series routers only) Create a snapshot of the root partition only and store it onto the default **/altroot** on the hard disk device or an **/altroot** on a USB device. Option deprecated for Junos OS with Upgraded FreeBSD in Junos OS Release 15.1.



NOTE: To determine which platforms run Junos OS with Upgraded FreeBSD, see the table listing the platforms currently running Junos OS with upgraded FreeBSD in “[Understanding Junos OS with Upgraded FreeBSD](#)” on page 17.

slice alternate—(EX Series switches only) (Optional) Take a snapshot of the active root partition and copy it to the alternate slice on the boot media.

Option deprecated for Junos OS with Upgraded FreeBSD in Junos OS Release 15.1.



NOTE: To determine which platforms run Junos OS with Upgraded FreeBSD, see the table listing the platforms currently running Junos OS with upgraded FreeBSD in “[Understanding Junos OS with Upgraded FreeBSD](#)” on page 17.

scc—(TX Matrix router only) (Optional) Archive data and executable areas for a TX Matrix router (or switch-card chassis).

sfc *number*—(TX Matrix Plus router only) (Optional) Archive data and executable areas for a TX Matrix Plus router (or switch-fabric chassis). Replace *number* with 0.

- Additional Information**
- (Routers only) Before upgrading the software on the router, when you have a known stable system, issue the **request system snapshot** command to back up the software, including the configuration, to the **/altroot** and **/altconfig** file systems. After you have upgraded the software on the router and are satisfied that the new packages are

successfully installed and running, issue the **request system snapshot** command again to back up the new software to the **/altroot** and **/altconfig** file systems.

- (Routers only) You cannot issue the **request system snapshot** command when you enable flash disk mirroring. We recommend that you disable flash disk mirroring when you upgrade or downgrade the software. For more information, see the *Junos OS Administration Library*
- (TX Matrix and TX Matrix Plus router only) On a routing matrix, if you issue the **request system snapshot** command on the master Routing Engine, all the master Routing Engines connected to the routing matrix are backed up. If you issue this command on the backup Routing Engine, all the backup Routing Engines connected to the routing matrix are backed up.

Required Privilege Level maintenance

Related Documentation

- [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 414](#)
- [show system snapshot on page 505](#)
- [show system auto-snapshot on page 488](#)

List of Sample Output

- [request system snapshot \(Routers\) on page 411](#)
- [request system snapshot \(EX Series Switches\) on page 411](#)
- [request system snapshot \(When the Partition Flag Is On\) on page 412](#)
- [request system snapshot \(MX104 Routers When Media Device is Missing\) on page 412](#)
- [request system snapshot \(When Mirroring Is Enabled\) on page 412](#)
- [request system snapshot all-lcc \(Routing Matrix\) on page 412](#)
- [request system snapshot all-members \(Virtual Chassis\) on page 412](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system snapshot (Routers)

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

The following filesystems were archived: / /config
```

request system snapshot (EX Series Switches)

```
user@switch> request system snapshot partition
Clearing current label...
Partitioning external media (/dev/da1) ...
Partitions on snapshot:

Partition Mountpoint Size Snapshot argument
```

```
s1a    /altroot    179M    none
s2a    /           180M    none
s3d    /var/tmp    361M    none
s3e    /var        121M    none
s4d    /config     60M     none
Copying '/dev/da0s1a' to '/dev/da1s1a' .. (this may take a few minutes)
Copying '/dev/da0s2a' to '/dev/da1s2a' .. (this may take a few minutes)
Copying '/dev/da0s3d' to '/dev/da1s3d' .. (this may take a few minutes)
Copying '/dev/da0s3e' to '/dev/da1s3e' .. (this may take a few minutes)
Copying '/dev/da0s4d' to '/dev/da1s4d' .. (this may take a few minutes)
The following filesystems were archived: /altroot / /var/tmp /var /config
```

request system snapshot (When the Partition Flag Is On)

```
user@host> request system snapshot partition
Performing preliminary partition checks ...
Partitioning ad0 ...
umount: /altroot: not currently mounted
Copying / to /altroot.. (this may take a few minutes)

The following filesystems were archived: / /config
```

request system snapshot (MX104 Routers When Media Device is Missing)

```
user@host > request system snapshot media usb0
error: usb0 media missing or invalid
```

request system snapshot (When Mirroring Is Enabled)

```
user@host> request system snapshot
Snapshot is not possible since mirror-flash-on-disk is configured.
```

request system snapshot all-lcc (Routing Matrix)

```
user@host> request system snapshot all-lcc
lcc0-re0:
-----
Copying '/' to '/altroot' .. (this may take a few minutes)
Copying '/config' to '/altconfig' .. (this may take a few minutes)
The following filesystems were archived: / /config

lcc2-re0:
-----
Copying '/' to '/altroot' .. (this may take a few minutes)
Copying '/config' to '/altconfig' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

request system snapshot all-members (Virtual Chassis)

```
user@switch> request system snapshot all-members media internal
fpc0:
-----
Copying '/dev/da0s2a' to '/dev/da0s1a' .. (this may take a few minutes)
The following filesystems were archived: /

fpc1:
-----
Copying '/dev/da0s2a' to '/dev/da0s1a' .. (this may take a few minutes)
The following filesystems were archived: /
```


fpc2:

Copying '/dev/da0s2a' to '/dev/da0s1a' .. (this may take a few minutes)
The following filesystems were archived: /

fpc3:

Copying '/dev/da0s2a' to '/dev/da0s1a' .. (this may take a few minutes)
The following filesystems were archived: /

fpc4:

Copying '/dev/da0s2a' to '/dev/da0s1a' .. (this may take a few minutes)
The following filesystems were archived: /

fpc5:

Copying '/dev/da0s2a' to '/dev/da0s1a' .. (this may take a few minutes)
The following filesystems were archived: /

request system snapshot (Junos OS with Upgraded FreeBSD)

Syntax `request system snapshot`
 `<delete snapshot-name>`
 `<load snapshot-name>`
 `<media type>`
 `<recovery>`

Release Information Command introduced in Junos OS Release 15.1 for MX240, MX480, MX960, MX2010, and MX2020 routers and EX9200 switches.
 Command introduced in Junos OS Release 17.3 for SRX5400, SRX5600, and SRX5800 devices.

Description On the router or switch, back up the currently running and active file system partitions to standby partitions that are not running. Non-recovery snapshots are named **snap.date.time** and stored in the **/packages/sets** directory.



.....
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.
.....

Options **none**—On the router or switch, back up the currently running and active file system partitions to standby partitions that are not running. Specifically, this creates a non-recovery snapshot named **snap.<date>.<time>** which is stored in **/packages/sets**.

delete snapshot-name—(Optional) Delete a specific non-recovery snapshot from **/packages/sets**. Wildcards are supported, so **request system snapshot delete snap*** deletes all snapshots.

load snapshot-name—(Optional) Load a specific snapshot from **/packages/sets**.

media type—(Optional) Specify the boot device the software is copied to:

- **usb**—(MX960 routers only) Copy software to the device connected to the USB port.

recovery—Create a recovery snapshot and store it in the **/oam** volume.

Additional Information Before upgrading the software on the router or switch, when you have a known stable system, issue the **request system snapshot** command to back up the software, including the configuration, to the **/packages/sets** file systems. After you have upgraded the software on the router or switch and are satisfied that the new packages are successfully installed and running, issue the **request system snapshot** command again to back up the new software to the **/packages/sets** file systems.

Required Privilege Level maintenance

Related Documentation

- [request system reboot \(Junos OS with Upgraded FreeBSD\) on page 398](#)
- [show system snapshot \(Junos OS with Upgraded FreeBSD\) on page 508](#)
- [Understanding Junos OS with Upgraded FreeBSD on page 17](#)

List of Sample Output

- [request system snapshot recovery on page 415](#)
- [request system snapshot delete on page 415](#)
- [request system snapshot on page 415](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

[request system snapshot recovery](#)

```
user@host> request system snapshot recovery
Creating image ...
Compressing image . . .
Image size is 777MB

Recovery snapshot created successfully
```

[request system snapshot delete](#)

```
user@host> request system snapshot delete snap.20150112.122106
NOTICE: Snapshot 'snap.20150112.122106' deleted successfully
```

[request system snapshot](#)

```
user@host> request system snapshot
NOTICE: Snapshot snap.20150119.122106 created successfully
```

request system snapshot (Maintenance)

Syntax request system snapshot
 <factory>
 <media (compact-flash | hard-disk | internal | usb)>
 <node (all | local | node-id | primary)>
 <partition>
 <slice (alternate) >

Release Information Command introduced in Junos OS Release 10.2.

Description Back up the currently running and active file system partitions on the device.

- Options**
- **factory**— (Optional) Specifies that only the files shipped from the factory are included in the snapshot.
 - **media**—(Optional) Specify the media to be included in the snapshot:
 - **compact-flash**—Copy the snapshot to the CompactFlash card.
 - **hard-disk**— Copy the snapshot to the hard disk.
 - **usb**— Copy the snapshot to the USB storage device.
 - **internal**— Copies the snapshot to internal media. This is the default.



NOTE: USB option is available on all SRX series devices; hard disk and compact-flash options are available only on SRX5800, SRX5600, and SRX5400 devices; media internal option is available only on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

- **external**— Copies the snapshot to an external storage device. This option is available for the compact flash on the SRX650 Services Gateway.
- **node**—(Optional) Specify the archive data and executable areas of a specific node.
 - **node-id**—Specify for node (0, 1).
 - **all**—Specify for all nodes.
 - **local**—Specify for local nodes.
 - **primary**— pecify for primary nodes.
- **partition**—(Default) Specify that the target media should be repartitioned before the backup is saved to it.



NOTE: The target media is partitioned whether or not it is specified in the command, because this is a mandatory option.

Example: `request system snapshot media usb partition`

Example: `request system snapshot media usb partition factory`

- **slice**—(Optional) Take a snapshot of the root partition the system has currently booted from to another slice in the same media.
- **alternate**—(Optional) Store the snapshot on the other root partition in the system.



NOTE: The slice option cannot be used along with the other `request system snapshot` options, because the options are mutually exclusive. If you use the factory, media, or partition option, you cannot use the slice option; if you use the slice option, you cannot use any of the other options.

Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none"> • Backing Up the Current Installation (SRX Series Devices) on page 57
List of Sample Output	request system snapshot media hard-disk on page 417 request system snapshot media usb (when usb device is missing on page 417 request system snapshot media compact-flash on page 418 request system snapshot partition on page 418
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

`request system snapshot media hard-disk`

```
user@host> request system snapshot media hard-disk
Verifying compatibility of destination media partitions...
Running newfs (880MB) on hard-disk media / partition (ad2s1a)...
Running newfs (98MB) on hard-disk media /config partition (ad2s1e)...
Copying '/dev/ad0s1a' to '/dev/ad2s1a' .. (this may take a few minutes)
...
```

`request system snapshot media usb (when usb device is missing`

```
user@host> request system snapshot media usb
Verifying compatibility of destination media partitions...
Running newfs (254MB) on usb media / partition (dals1a)...
Running newfs (47MB) on usb media /config partition (dals1e)...
```

```
Copying '/dev/da0s2a' to '/dev/da1s1a' .. (this may take a few minutes)
Copying '/dev/da0s2e' to '/dev/da1s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

request system snapshot media compact-flash

```
user@host> request system snapshot media compact-flash
error: cannot snapshot to current boot device
```

request system snapshot partition

```
user@host> request system snapshot partition
Verifying compatibility of destination media partitions...
Running newfs (439MB) on internal media / partition (da0s1a)...
Running newfs (46MB) on internal media /config partition (da0s1e)...
Copying '/dev/da1s1a' to '/dev/da0s1a' .. (this may take a few minutes)
Copying '/dev/da1s1e' to '/dev/da0s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

request system software abort in-service-upgrade (ICU)

Syntax request system software abort in-service-upgrade

Release Information Command introduced in Junos OS Release 11.2 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

Description Abort an in-band cluster upgrade (ICU). This command must be issued from a router session other than the one on which you issued the **request system in-service-upgrade** command that launched the ICU. If an ICU is in progress, this command aborts it. If the node is being upgraded, this command will cancel the upgrade. The command is also helpful in recovering the node in case of a failed ICU.



NOTE: We recommend that you use the command only when there is an issue with the ongoing session of ISSU. You may need to manually intervene to bring the system to sane state if after issuing the command the system does not recover from the abort.

Options This command has no options.

Required Privilege Level view

Related Documentation

- *request system software in-service-upgrade (Maintenance)*

List of Sample Output [request system software abort in-service-upgrade on page 419](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system software abort in-service-upgrade

```
user@host> request system software abort in-service-upgrade
In-Service-Upgrade aborted
```

request system software add

List of Syntax [Syntax on page 420](#)
 [Syntax \(EX Series Switches\) on page 420](#)
 [Syntax \(TX Matrix Router\) on page 420](#)
 [Syntax \(TX Matrix Plus Router\) on page 421](#)
 [Syntax \(MX Series Router\) on page 421](#)
 [Syntax \(QFX Series\) on page 421](#)
 [Syntax \(OCX Series\) on page 422](#)

Syntax `request system software add package-name`
 `<best-effort-load>`
 `<delay-restart>`
 `<device-alias alias-name>`
 `<force>`
 `<no-copy>`
 `<no-validate>`
 `<re0 | re1>`
 `<reboot>`
 `<satellite slot-id>`
 `<set [package-name1 package-name2]>`
 `<unlink>`
 `<upgrade-group [all | upgrade-group-name]>`
 `<upgrade-with-config>`
 `<satellite slot-id>`
 `<validate>`
 `<version version-string>`

Syntax (EX Series Switches) `request system software add package-name`
 `<best-effort-load>`
 `<delay-restart>`
 `<force>`
 `<no-copy>`
 `<no-validate>`
 `<re0 | re1>`
 `<reboot>`
 `<set [package-name1 package-name2]>`
 `<upgrade-with-config>`
 `<validate>`
 `<validate-on-host hostname>`
 `<validate-on-routing-engine routing-engine>`

Syntax (TX Matrix Router) `request system software add package-name`
 `<best-effort-load>`
 `<delay-restart>`
 `<force>`
 `<lcc number | scc>`
 `<no-copy>`
 `<no-validate>`
 `<re0 | re1>`
 `<reboot>`
 `<set [package-name1 package-name2]>`


```

<unlink>
<upgrade-with-config>
<validate>
<validate-on-host hostname>
<validate-on-routing-engine routing-engine>

```

Syntax (TX Matrix Plus Router) request system software add *package-name*

```

<best-effort-load>
<delay-restart>
<force>
<lcc number | sfc number>
<no-copy>
<no-validate>
<re0 | re1>
<reboot>
<set [package-name1 package-name2]>
<unlink>
<upgrade-with-config>
<validate>
<validate-on-host hostname>
<validate-on-routing-engine routing-engine>

```

Syntax (MX Series Router) request system software add *package-name*

```

<best-effort-load>
<delay-restart>
<device-alias alias-name>
<force>
<member member-id>
<no-copy>
<no-validate>
<re0 | re1>
<reboot>
<satellite slot-id>
<set [package-name1 package-name2]>
<upgrade-group [all | upgrade-group-name]>
<unlink>
<upgrade-with-config>
<validate>
<version version-string>
<validate-on-host hostname>
<validate-on-routing-engine routing-engine>

```

Syntax (QFX Series) request system software add *package-name*

```

<best-effort-load>
<component all>
<delay-restart>
<force>
<force-host>
<no-copy>
<no-validate>
<partition>
<reboot>
<unlink>
<upgrade-with-config>

```

```
<validate>  
<validate-on-host hostname>  
<validate-on-routing-engine routing-engine>
```

Syntax (OCX Series) request system software add *package-name*
<best-effort-load>
<delay-restart>
<force>
<force-host>
<no-copy>
<no-validate>
<reboot>
<unlink>
<upgrade-with-config>
<validate>

Release Information Command introduced before Junos OS Release 7.4.
best-effort-load and **unlink** options added in Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
sfc option introduced in Junos OS Release 9.6 for the TX Matrix Plus router.
Command introduced in Junos OS Release 11.1 for the QFX Series.
set [package-name1 package-name2] option added in Junos OS Release 11.1 for EX Series switches. Added in Junos OS Release 12.2 for M Series, MX Series, and T Series routers.



NOTE: On EX Series switches, the set [*package-name1 package-name2*] option allows you to install only two software packages on a mixed EX4200 and EX4500 Virtual Chassis, whereas, on M Series, MX Series, and T Series routers, the set [*package-name1 package-name2 package-name3*] option allows you to install multiple software packages and software add-on packages at the same time.

upgrade-with-config and **upgrade-with-config-format** *format* options added in Junos OS Release 12.3 for M Series routers, MX Series routers, and T Series routers, EX Series Ethernet switches, and QFX Series devices.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
device-alias, **satellite**, **upgrade-group**, and **version** options introduced in Junos OS Release 14.2R3 for Junos Fusion.
validate-on-host and **validate-on-routing-engine** options added in Junos OS Release 15.1F3 for PTX5000 routers and MX240, MX480, and MX960 routers.
upgrade-with-config-format *format* option deleted in Junos OS Release 16.1 for M Series routers, MX Series routers, and T Series routers, EX Series Ethernet switches, and QFX Series devices.

Description



NOTE: We recommend that you always download the software image to `/var/tmp` only. On EX Series and QFX Series switches, you must use the `/var/tmp` directory. Other directories are not supported.

Install a software package or bundle on the router or switch.

For information on valid filename and URL formats, see *Format for Specifying Filenames and URLs in Junos OS CLI Commands*.



CAUTION: Any configuration changes performed after inputting the `request system software add` command will be lost when the system reboots with an upgraded version of Junos OS.



NOTE: When graceful Routing Engine switchover (GRES) is enabled on a device, you must perform a unified ISSU operation to update the software running on the device. With GRES enabled, if you attempt to perform a software upgrade by entering the `request system software add package-name` command, an error message is displayed stating that only in-service-software-upgrades are supported when GRES is configured. In such a case, you must either remove the GRES configuration before you attempt the upgrade or perform a unified ISSU.



NOTE: Starting with Junos OS Release 15.1F3, the statement `request system software add` installs a software package for the guest OS only for the PTX5000 router with RE-DUO-C2600-16G, and for MX240, MX480, and MX960 routers with RE-S-1800X4-32G-S.

Starting with Junos OS Release 15.1F5, the statement `request system software add` installs a software package for the guest OS only for the MX2010 and MX2020 routers with REMX2K-1800-32G-S.

On these routers, in order to install both Junos software and host software packages, use the `request vmhost software add` command.

Options `package-name`—Location from which the software package or bundle is to be installed.
For example:

- `/var/tmp/package-name`—For a software package or bundle that is being installed from a local directory on the router or switch.
- `protocol://hostname/pathname/package-name`—For a software package or bundle that is to be downloaded and installed from a remote location. Replace *protocol* with one of the following:
 - `ftp`—File Transfer Protocol.

Use **ftp://hostname/pathname/package-name**. To specify authentication credentials, use

ftp://<username>:<password>@hostname/pathname/package-name. To have the system prompt you for the password, specify **prompt** in place of the password. If a password is required, and you do not specify the password or **prompt**, an error message is displayed.

- **http**—Hypertext Transfer Protocol.
Use **http://hostname/pathname/package-name**. To specify authentication credentials, use **http://<username>:<password>@hostname/pathname/package-name**. If a password is required and you omit it, you are prompted for it.
- **scp**—Secure copy (available only for Canada and U.S. version).
Use **scp://hostname/pathname/package-name**. To specify authentication credentials, use **scp://<username>:<password>@hostname/pathname/package-name**.



NOTE:

- The *pathname* in the protocol is the relative path to the user's home directory on the remote system and not the root directory.
 - Do not use the **scp** protocol in the **request system software add** command to download and install a software package or bundle from a remote location. The previous statement does not apply to the QFabric switch. The software upgrade is handled by the management process (**mgd**), which does not support **scp**.
Use the **file copy** command to copy the software package or bundle from the remote location to the **/var/tmp** directory on the hard disk:
file copy scp://source/package-name /var/tmp
Then install the software package or bundle using the **request system software add** command:
request system software add /var/tmp/package-name
-

best-effort-load—(Optional) Activate a partial load and treat parsing errors as warnings instead of errors.

component all—(QFabric systems only) (Optional) Install software package on all of the QFabric components.

delay-restart—(Optional) Install a software package or bundle, but do not restart software processes.

device-alias alias-name—(Junos Fusion only) (Optional) Install the satellite software package onto the specified satellite device using the satellite device's alias name.

force—(Optional) Force the addition of the software package or bundle (ignore warnings).

force-host—(Optional) Force the addition of host software package or bundle (ignore warnings) on the QFX5100 device.

lcc *number*—(TX Matrix routers and TX Matrix Plus routers only) (Optional) In a routing matrix based on the TX Matrix router, install a software package or bundle on a T640 router that is connected to the TX Matrix router. In a routing matrix based on the TX Matrix Plus router, install a software package or bundle on a router that is connected to the TX Matrix Plus router.

Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.
- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.
- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

member *member-id*—(MX Series routers only) (Optional) Install a software package on the specified Virtual Chassis member. Replace *member-id* with a value of 0 or 1.

partition—(QFX3500 switches only) (Optional) Format and repartition the media before installation.

satellite *slot-id*—(Junos Fusion only) (Optional) Install the satellite software package onto the specified satellite device using the satellite devices FPC slot identifier.

scc—(TX Matrix routers only) (Optional) Install a software package or bundle on a Routing Engine on a TX Matrix router (or switch-card chassis).

sfc *number*—(TX Matrix Plus routers only) (Optional) Install a software package or bundle on a Routing Engine on a TX Matrix Plus router. Replace *number* with 0.

no-copy—(Optional) Install a software package or bundle, but do not save copies of the package or bundle files.

no-validate—(Optional) When loading a software package or bundle with a different release, suppress the default behavior of the **validate** option.



NOTE: Software packages from unidentified providers cannot be loaded. To authorize providers, include the **provider-id** statement at the [edit system extensions provider] hierarchy level.

re0 | re1—(Optional) On routers or switches that support dual or redundant Routing Engines, load a software package or bundle on the Routing Engine in slot 0 (re0) or the Routing Engine in slot 1 (re1).

reboot—(Optional) After adding the software package or bundle, reboot the system. On a QFabric switch, the software installation is not complete until you reboot the component for which you have installed the software.

set [package-name1 package-name2]—(Mixed EX4200 and EX4500 Virtual Chassis, M Series, MX Series, and T Series routers only) (Optional) Install multiple packages at same time:

- In the case of mixed EX4200 and EX4500 Virtual Chassis, install two software packages—a package for an EX4200 switch and the same release of the package for an EX4500 switch—to upgrade all member switches in a mixed EX4200 and EX4500 Virtual Chassis.
- In the case of M Series, MX Series, and T Series routers, install multiple (two or more) software packages and software add-on packages at the same time. The variable **package-name** can either be a list of installation packages, each separated by a blank space, or the full URL to the directory or tar file containing the list of installation packages.

In each case, **installation-package** can either be a list of installation packages, each separated by a blank space, or the full URL to the directory or tar file containing the list of installation packages.

Use the **request system software add set** command to retain any SDK configuration by installing the SDK add-on packages along with the core Junos OS installation package.

unlink—(Optional) On M Series, T Series, and MX Series routers, use the unlink option to remove the software package from this directory after a successful upgrade is completed.

upgrade-group [all | upgrade-group-name]—(Junos Fusion only) (Required to configure a Junos Fusion using autoconversion or manual conversion) Associate a satellite software image with a satellite software upgrade group. The satellite software package is associated with the specified satellite software upgrade group using the *upgrade-group-name*, or for all satellite software upgrade groups in a Junos Fusion when the all keyword is specified.

A satellite software upgrade group is a group of satellite devices in a Junos Fusion that are designated to upgrade to the same satellite software version using the same satellite software package. See *Understanding Software in a Junos Fusion Provider Edge*, *Understanding Software in a Junos Fusion Enterprise*, and *Managing Satellite Software Upgrade Groups in a Junos Fusion*.

upgrade-with-config—(Optional) Install one or more configuration files.



NOTE: Configuration files specified with this option must have the extension `.text` or `.xml` and have the extension specified. Using the extension `.txt` will not work.

validate—(Optional) Validate the software package or bundle against the current configuration as a prerequisite to adding the software package or bundle. This is the default behavior when the software package or bundle being added is a different release.



NOTE: The **validate** option only works on systems that do not have graceful-switchover (GRES) enabled. To use the **validate** option on a system with GRES, either disable GRES for the duration of the installation, or install using the command **request system software in-service-upgrade**, which requires nonstop active routing (NSR) to be enabled when using GRES.

validate-on-host *hostname*—(Optional) Validate the software package by comparing it to the running configuration on a remote Junos OS host. Specify a host, replacing ***hostname*** with the remote hostname. You can optionally provide the username that will be used to log in to the remote host by specifying the hostname in the format ***user@hostname***.

validate-on-routing-engine *routing-engine*—(Optional) Validate the software bundle or package by comparing it to the running configuration on a Junos OS Routing Engine on the same chassis. Specify a Routing Engine, replacing ***routing-engine*** with the routing engine name.

Additional Information Before upgrading the software on the router or switch, when you have a known stable system, issue the **request system snapshot** command to back up the software, including the configuration, to the **/altroot** and **/altconfig** file systems. After you have upgraded the software on the router or switch and are satisfied that the new package or bundle is successfully installed and running, issue the **request system snapshot** command again to back up the new software to the **/altroot** and **/altconfig** file systems.



NOTE: The **request system snapshot** command is currently not supported on the QFabric system. Also, you cannot add or install multiple packages on a QFabric system.

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.

If you are upgrading more than one package at the same time, delete the operating system package, jkernel, last. Add the operating system package, jkernel, first and the routing software package, jroute, last. If you are upgrading all packages at once, delete and add them in the following order:

```
user@host> request system software add /var/tmp/jbase
user@host> request system software add /var/tmp/jkernel
user@host> request system software add /var/tmp/jpfe
user@host> request system software add /var/tmp/jdocs
user@host> request system software add /var/tmp/jroute
user@host> request system software add /var/tmp/jcrypto
```

By default, when you issue the **request system software add *package-name*** command on a TX Matrix master Routing Engine, all the T640 master Routing Engines that are connected to it are upgraded to the same version of software. If you issue the same command on the TX Matrix backup Routing Engine, all the T640 backup Routing Engines that are connected to it are upgraded to the same version of software.

Likewise, when you issue the **request system software add *package-name*** command on a TX Matrix Plus master Routing Engine, all the T1600 or T4000 master Routing Engines that are connected to it are upgraded to the same version of software. If you issue the same command on the TX Matrix Plus backup Routing Engine, all the T1600 or T4000 backup Routing Engines that are connected to it are upgraded to the same version of software.

Required Privilege Level

maintenance

Related Documentation

- *Format for Specifying Filenames and URLs in Junos OS CLI Commands*
- [request system software delete on page 435](#)
- [request system software rollback on page 439](#)
- [request system storage cleanup on page 452](#)
- *Upgrading Software*
- *Upgrading Software on a QFabric System*
- *Managing Satellite Software Upgrade Groups in a Junos Fusion*
- [request system software add \(Maintenance\) on page 432](#)
- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)

List of Sample Output

[request system software add validate on page 429](#)
[request system software add validate-on-host on page 429](#)
[request system software add \(Mixed EX4200 and EX4500 Virtual Chassis\) on page 431](#)
[request system software add component all \(QFabric Systems\) on page 431](#)

[request system software add upgrade-group \(Junos Fusion\) on page 431](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

[request system software add validate](#)

```
user@host> request system software add validate /var/tmp/jinstall-7.2R1.7-domestic-signed.tgz
Checking compatibility with configuration
Initializing...
Using jbase-7.1R2.2
Using /var/tmp/jinstall-7.2R1.7-domestic-signed.tgz
Verified jinstall-7.2R1.7-domestic.tgz signed by PackageProduction_7_2_0
Using /var/validate/tmp/jinstall-signed/jinstall-7.2R1.7-domestic.tgz
Using /var/validate/tmp/jinstall/jbundle-7.2R1.7-domestic.tgz
Checking jbundle requirements on /
Using /var/validate/tmp/jbundle/jbase-7.2R1.7.tgz
Using /var/validate/tmp/jbundle/jkernel-7.2R1.7.tgz
Using /var/validate/tmp/jbundle/jcrypto-7.2R1.7.tgz
Using /var/validate/tmp/jbundle/jpfe-7.2R1.7.tgz
Using /var/validate/tmp/jbundle/jdocs-7.2R1.7.tgz
Using /var/validate/tmp/jbundle/jroute-7.2R1.7.tgz
Validating against /config/juniper.conf.gz
mgd: commit complete
Validation succeeded
Validating against /config/rescue.conf.gz
mgd: commit complete
Validation succeeded
Installing package '/var/tmp/jinstall-7.2R1.7-domestic-signed.tgz' ...
Verified jinstall-7.2R1.7-domestic.tgz signed by PackageProduction_7_2_0
Adding jinstall...

WARNING: This package will load JUNOS 7.2R1.7 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 ...
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-7.2R1.7-domestic-signed.tgz ...
Saving state for rollback ...
```

[request system software add validate-on-host](#)

```
user@host> request system software add validate-on-host user@xyz
:/var/tmp/jinstall-15.1-20150516_ib_15_2_psd.0-domestic-signed.tgz
user@host> request system software add validate-on-host user@xyz
:/var/tmp/jinstall-15.1-20150516_ib_15_2_psd.0-domestic-signed.tgz
```

```
Extracting JUNOS version from package...
Connecting to remote host xyz...
Password:
Sending configuration to xyz...
Validating configuration on xyz...
PACKAGE TYPE: not found
Checking compatibility with configuration
Initializing...
Using jbase-15.1-20150516_ib_15_2_psd.0
Verified manifest signed by PackageDevelopmentEc_2015
Using jruntime-15.1-20150516_ib_15_2_psd.0
Verified manifest signed by PackageDevelopmentEc_2015
Using jkernel-15.1-20150516_ib_15_2_psd.0
Verified manifest signed by PackageDevelopmentEc_2015
Using jroute-15.1-20150516_ib_15_2_psd.0
Verified manifest signed by PackageDevelopmentEc_2015
Using jcrypto-15.1-20150516_ib_15_2_psd.0
Verified manifest signed by PackageDevelopmentEc_2015
Using jweb-15.1-20150516_ib_15_2_psd.0
Verified manifest signed by PackageDevelopmentEc_2015
Using /var/packages/jtools-15.1-20150516_ib_15_2_psd.0
Verified manifest signed by PackageDevelopmentEc_2015
Using /var/tmp/config.tgz
Hardware Database regeneration succeeded
Validating against /config/juniper.conf.gz
mgd: warning: schema: init: 'logical-systems-vlans' contains-node 'juniper-config
  vlans': not found
mgd: commit complete
Validation succeeded
Installing package
'/var/tmp/jinstall-15.1-20150516_ib_15_2_psd.0-domestic-signed.tgz' ...
Verified jinstall-15.1-20150516_ib_15_2_psd.0-domestic.tgz signed by
PackageDevelopmentEc_2015
Adding jinstall...

WARNING:    The software that is being installed has limited support.
WARNING:    Run 'file show /etc/notices/unsupported.txt' for details.

WARNING:    This package will load JUNOS 15.1-20150516_ib_15_2_psd.0 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-15.1-20150516_ib_15_2_psd.0-domestic-signed.tgz ...
Saving state for rollback ...
```

Sample Output

request system software add (Mixed EX4200 and EX4500 Virtual Chassis)

```
user@switch> request system software add set
[/var/tmp/jinstall-ex-4200-11.1R1.1-domestic-signed.tgz
/var/tmp/jinstall-ex-4500-11.1R1.1-domestic-signed.tgz]
...
```

request system software add component all (QFabric Systems)

```
user@switch> request system software add /pbdata/packages/jinstall-qfabric-12.2X50-D1.3.rpm
component all
...
```

request system software add upgrade-group (Junos Fusion)

```
user@aggregation-device> request system software add /var/tmp/satellite-3.0R1.1-signed.tgz
upgrade-group group1
```

request system software add (Maintenance)

Syntax	<code>request system software add <i>package-name</i></code>
Release Information	Partition option introduced in the command in Junos OS Release 10.1.
Description	Install the new software package on the device, for example: request system software add junos-srxsme-10.0R2-domestic.tgz no-copy no-validate partition reboot .
Options	<ul style="list-style-type: none">• delay-restart—Install the software package but does not restart the software process.• best-effort-load—Activate a partial load and treat parsing errors as warnings instead of errors.• no-copy—Install the software package but does not saves the copies of package files.• no-validate—Do not check the compatibility with current configuration before installation starts.• partition—Format and re-partition the media before installation.• reboot—Reboot the device after installation is completed.• unlink—Remove the software package after successful installation.• validate—Check the compatibility with current configuration before installation starts.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• <i>request system reboot</i>

request system software configuration-backup

Syntax	<code>request system software configuration-backup (path)</code>
Release Information	Command introduced in Junos OS Release 11.3 for the QFX Series.
Description	Save the currently active configuration and any installation-specific parameters such as a configuration that you have entered outside of the CLI, Director group IP addresses, and the default partition IP address.
Options	path —(QFabric System) Provide the path to the location of the backup configuration files. You can save the backup configuration files to either a URL, local directory, remote server, or removable drive.
Required Privilege Level	<code>configure</code> —To enter configuration mode, but other required privilege levels depend on where the statement is located in the configuration hierarchy.
Related Documentation	<ul style="list-style-type: none"> • request system software configuration-restore on page 434
List of Sample Output	request system software configuration-backup on page 433
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system software configuration-backup

```

user@switch request system software configuration-backup ftp://ftp.test.net/test
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                               Dload  Upload  Total   Spent    Left
Speed
100      4035    0    0   100 4035    0    138k  --:--:--  --:--:--  --:--:--
0

```

request system software configuration-restore


Syntax	<code>request system software configuration-restore (path)</code>
Release Information	Command introduced in Junos OS Release 11.3 for the QFX Series.
Description	Restore a previously saved configuration and any installation-specific parameters, such as a configuration that you have entered outside of the CLI, Director group IP addresses, and the default partition IP address.
Options	path —(QFabric System) Provide the path to the location of the backup configuration files. The path can be to a local file, a file on an external flash drive, or an SCP or FTP destination.
Required Privilege Level	configure —To enter configuration mode, but other required privilege levels depend on where the statement is located in the configuration hierarchy.
Related Documentation	<ul style="list-style-type: none">• request system software configuration-backup on page 433• <i>Performing a QFabric System Recovery Installation on the Director Group</i>
List of Sample Output	request system software configuration-restore on page 434
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system software configuration-restore

```
user@switch request system software configuration-restore ftp://ftp.test.net/test
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left     Speed
100 4035  100 4035    0     0  153k      0  --:--:--  --:--:--  --:--:-- 3803k
```

request system software delete

List of Syntax	Syntax on page 435 Syntax (TX Matrix Router) on page 435 Syntax (TX Matrix Plus Router) on page 435
Syntax	<pre>request system software delete <i>software-package</i> <force> <reboot> <set [<i>package-name package-name</i>]> <upgrade-group [<i>all upgrade-group-name</i>]> <version <i>version-string</i>></pre>
Syntax (TX Matrix Router)	<pre>request system software delete <i>software-package</i> <force> <lcc <i>number</i> scc> <reboot> <set [<i>package-name package-name</i>]></pre>
Syntax (TX Matrix Plus Router)	<pre>request system software delete <i>software-package</i> <force> <lcc <i>number</i> sfc <i>number</i>> <reboot> <set [<i>package-name package-name</i>]></pre>
Release Information	<p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>sfc option introduced in Junos OS Release 9.6 for the TX Matrix Plus router.</p> <p>Command introduced in Junos OS Release 11.1 for the QFX Series.</p> <p>set [<i>package-name package-name</i>] option added in Junos OS Release 12.2 for M Series, MX Series, and T Series routers.</p> <p>reboot option introduced in Junos OS Release 12.3.</p> <p>Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.</p> <p>upgrade-group, and version options introduced in Junos OS Release 14.2R3 for Junos Fusion.</p>
Description	Remove a software package or bundle from the router or switch.
	
<p>CAUTION: Before removing a software package or bundle, make sure that you have already placed the new software package or bundle that you intend to load onto the router or switch.</p>	
Options	<p><i>software-package</i>—Software package or bundle name. You can delete any or all of the following software bundles or packages:</p>

- **jbase**—(Optional) Junos base software suite
- **jcrypto**—(Optional, in domestic version only) Junos security software
- **jdocs**—(Optional) Junos online documentation file
- **jkernel**—(Optional) Junos kernel software suite
- **jpfe**—(Optional) Junos Packet Forwarding Engine support
- **jroute**—(Optional) Junos routing software suite
- **junos**—(Optional) Junos base software



NOTE: On EX Series switches, some of the package names are different than those listed. To see the list of packages that you can delete on an EX Series switch, enter the command **show system software**.

force—(Optional) Ignore warnings and force removal of the software.

lcc *number*—(TX Matrix routers and TX Matrix Plus routers only) (Optional) In a routing matrix, delete a software package or bundle on a T640 router indicated by **lcc *number*** that is connected to the TX Matrix router. In a routing matrix, delete a software package or bundle on a router indicated by **lcc *number*** that is connected to the TX Matrix Plus router.

Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.
- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.
- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

re0 | re1—(Optional) On routers or switches that support dual or redundant Routing Engines, delete a software package or bundle on the Routing Engine in slot 0 (re0) or the Routing Engine in slot 1 (re1).

reboot—As of Junos OS 12.3 and greater, automatically reboot upon completing the **request system software delete** command.

scc—(TX Matrix routers only) (Optional) Remove an extension or upgrade package from the TX Matrix router (or switch-card chassis).

set [*package-name package-name*]
(Optional) Install multiple software packages or software add-on packages at the same time.

sfc number—(TX Matrix Plus routers only) (Optional) Remove an extension or upgrade package from the TX Matrix Plus router. Replace *number* with 0.

upgrade-group [**all** | *upgrade-group-name*]—(Junos Fusion only) Delete the satellite software image association with the specified satellite software upgrade group.

A satellite software upgrade group is a group of satellite devices in the same Junos Fusion that are designated to upgrade to the same satellite software version using the same satellite software package.

version version-string—(Junos Fusion only) (Optional) Delete a satellite software package association with a satellite software upgrade group by selecting the satellite software package's version.

Additional Information Before upgrading the software on the router or switch, when you have a known stable system, issue the **request system snapshot** command to back up the software, including the configuration, to the /altroot and /altconfig file systems (on routers) or the /, /altroot, /config, /var, and /var/tmp file systems (on switches). After you have upgraded the software on the router or switch and are satisfied that the new packages are successfully installed and running, issue the **request system snapshot** command again to back up the new software to the /altroot and /altconfig file systems (on routers) or the /, /altroot, /config, /var, and /var/tmp file systems (on switches). 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.

Required Privilege Level maintenance

Related Documentation

- [request system software add on page 420](#)
- [request system software rollback on page 439](#)
- [request system software validate on page 445](#)
- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)

List of Sample Output [request system software delete jdocs on page 437](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

[request system software delete jdocs](#)

The following example displays the system software packages before and after the **jdocs** package is deleted through the **request system software delete** command:

```
user@host> show system software
Information for jbase:

Comment:
JUNOS Base OS Software Suite [7.2R1.7]

Information for jcrypto:

Comment:
JUNOS Crypto Software Suite [7.2R1.7]

Information for jdocs:

Comment:
JUNOS Online Documentation [7.2R1.7]

Information for jkernel:

Comment:
JUNOS Kernel Software Suite [7.2R1.7]

...

user@host> request system software delete jdocs
Removing package 'jdocs' ...

user@host> show system software
Information for jbase:

Comment:
JUNOS Base OS Software Suite [7.2R1.7]

Information for jcrypto:

Comment:
JUNOS Crypto Software Suite [7.2R1.7]

Information for jkernel:

Comment:
JUNOS Kernel Software Suite [7.2R1.7]

...
```

request system software rollback

List of Syntax	Syntax on page 439 Syntax (EX Series Switches) on page 439 Syntax (TX Matrix Router) on page 439 Syntax (TX Matrix Plus Router) on page 439 Syntax (MX Series Router) on page 439
Syntax	request system software rollback
Syntax (EX Series Switches)	request system software rollback <all-members> <local> <member <i>member-id</i> > <reboot>
Syntax (TX Matrix Router)	request system software rollback <lcc <i>number</i> scc> <reboot>
Syntax (TX Matrix Plus Router)	request system software rollback <lcc <i>number</i> sfc <i>number</i> > <reboot>
Syntax (MX Series Router)	request system software rollback <all-members> <device-alias <i>alias-name</i> > <local> <member <i>member-id</i> > <reboot> <satellite <i>slot-id</i> > <upgrade-group [all <i>upgrade-group-name</i>]>
Release Information	Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 9.0 for EX Series switches. sfc option introduced in Junos OS Release 9.6 for the TX Matrix Plus router. Command introduced in Junos OS Release 11.1 for the QFX Series. Command behavior changed in Junos OS Release 12.1. reboot option introduced in Junos OS Release 12.3. device-alias , satellite , and upgrade-group options introduced in Junos OS Release 14.2R3 for Junos Fusion. force option deprecated in Junos OS Release 15.1 for Junos OS with Upgraded FreeBSD.



NOTE: To determine which platforms run Junos OS with Upgraded FreeBSD, see the table listing the platforms currently running Junos OS with upgraded FreeBSD in “[Understanding Junos OS with Upgraded FreeBSD](#)” on page 17.

Description This command reverts to the last successfully installed package before the **request system software (add | delete)** command. It uses the copy stored in the `/var/sw/pkg` directory.

Additional Information

- On Junos Fusion, the **request system software rollback** command can be used to roll back the version of satellite software associated with a satellite software upgrade group. Rolling back the version of satellite software associated with a satellite software upgrade group triggers a satellite software upgrade.
- On M Series and T Series routers, if **request system software add <jinstall> reboot** was used for the previous installation, then **request system software rollback** has no effect. In this case, use **jinstall** to reinstall the required package.
- On M Series and T Series routers, if **request system software add <sdk1>** was used for the previous installation, then **request system software rollback** removes the last installed SDK package (**sdk1** in this example).
- On SRX Series devices with dual root systems, when **request system software rollback** is run, the system switches to the alternate root. Each root can have a different version of Junos OS. Roll back takes each root back to the previously installed image.
- On QFX3500 and QFX3600 devices in a mixed Virtual Chassis, when the **request system software rollback** command is issued, the system does not rollback to the image stored in the alternate partition.
- On QFX5100 switches, the **reboot** option has been removed. To reboot the switch after a software rollback, issue the **request system reboot** command as a separate, secondary command.

Options **all-members**—(EX4200 switches and MX Series routers only) (Optional) Attempt to roll back to the previous set of packages on all members of the Virtual Chassis configuration.

device-alias *alias-name*—(Junos Fusion only) (Optional) Rollback the satellite software package onto the specified satellite device using the satellite devices FPC slot identifier.

lcc *number*—(TX Matrix routers and TX Matrix Plus routers only) (Optional) On a TX Matrix router, attempt to roll back to the previous set of packages on a T640 router connected to the TX Matrix router. On a TX Matrix Plus router, attempt to roll back to the previous set of packages on a connected router connected to the TX Matrix Plus router.

Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.
- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.

- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

local—(EX4200 switches and MX Series routers only) (Optional) Attempt to roll back to the previous set of packages on the local Virtual Chassis member.

member *member-id*—(EX4200 switches and MX Series routers only) (Optional) Attempt to roll back to the previous set of packages on the specified member of the Virtual Chassis configuration. For EX4200 switches, replace *member-id* with a value from 0 through 9. For an MX Series Virtual Chassis, replace *member-id* with a value of 0 or 1.

none—For all versions of Junos OS up to and including Junos OS 11.4, revert to the set of software as of the last successful **request system software add**. As of Junos OS 12.1 and later, revert to the last known good state before the most recent **request system software (add | delete)** command.

reboot—For Junos OS 12.3 and later, the system reboots automatically to complete the rollback.

satellite *slot-id*—(Junos Fusion only) (Optional) Roll back the satellite software package onto the specified satellite device using the satellite devices FPC slot identifier.

scc—(TX Matrix routers only) (Optional) Attempt to roll back to the previous set of packages on the TX Matrix router (or switch-card chassis).

sfc *number*—(TX Matrix Plus routers only) (Optional) Attempt to roll back to the previous set of packages on the TX Matrix Plus router. Replace *number* with 0.

upgrade-group [all | *upgrade-group-name*]—(Junos Fusion only) Roll back the satellite software image associated with the specified satellite software upgrade group, or for all satellite software upgrade groups in the Junos Fusion when **all** is entered.

Required Privilege Level maintenance

Related Documentation

- *request system software abort*
- [request system software add on page 420](#)
- [request system software delete on page 435](#)
- [request system software validate on page 445](#)
- *request system configuration rescue delete*
- *request system configuration rescue save*
- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)

List of Sample Output [request system software rollback on page 443](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system software rollback

```

user@host> request system software rollback
Verified SHA1 checksum of ./jbase-7.2R1.7.tgz
Verified SHA1 checksum of ./jdocs-7.2R1.7.tgz
Verified SHA1 checksum of ./jroute-7.2R1.7.tgz
Installing package './jbase-7.2R1.7.tgz' ...
Available space: 35495 require: 7335
Installing package './jdocs-7.2R1.7.tgz' ...
Available space: 35339 require: 3497
Installing package './jroute-7.2R1.7.tgz' ...
Available space: 35238 require: 6976
NOTICE: uncommitted changes have been saved in
/var/db/config/juniper.conf.pre-install
Reloading /config/juniper.conf.gz ...
Activating /config/juniper.conf.gz ...
mgd: commit complete
Restarting mgd ...
Restarting aprobed ...
Restarting apsd ...
Restarting cosd ...
Restarting fsad ...
Restarting fud ...
Restarting gcdrd ...
Restarting ilmid ...
Restarting irsd ...
Restarting l2tpd ...
Restarting mib2d ...
Restarting nasd ...
Restarting pppoed ...
Restarting rdd ...
Restarting rmopd ...
Restarting rtspd ...
Restarting sampled ...
Restarting serviced ...
Restarting snmpd ...
Restarting spd ...
Restarting vrrpd ...

WARNING: cli has been replaced by an updated version:
CLI release 7.2R1.7 built by builder on 2005-04-22 02:03:44 UTC
Restart cli using the new version ? [yes,no] (yes) yes

Restarting cli ...
user@host

```

request system software rollback (SRX Series)

Syntax	<code>request system software rollback <node-id></code>
Release Information	Command introduced in Junos OS Release 10.1. Command introduced in Junos OS Release 15.1X49-D50 for SRX1500 devices.
Description	Revert to the software that was loaded at the last successful request system software add command. .
Options	<i>node-id</i> —Identification number of the chassis cluster node. It can be 0 or 1.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• <i>request system reboot</i>

request system software validate

List of Syntax	Syntax on page 445 Syntax (TX Matrix Router) on page 445 Syntax (TX Matrix Plus Router) on page 445 Syntax (MX Series Router) on page 445
Syntax	<pre>request system software validate <i>package-name</i> <on (host <i>host</i> <username <i>username</i>> routing-engine <i>routing-engine</i>)> <set [<i>package-name package-name</i>]> <upgrade-with-config> <upgrade-with-config-format <i>format</i>></pre>
Syntax (TX Matrix Router)	<pre>request system software validate <i>package-name</i> <lcc <i>number</i> scc> <on (host <i>host</i> <username <i>username</i>> routing-engine <i>routing-engine</i>)> <set [<i>package-name package-name</i>]> <upgrade-with-config> <upgrade-with-config-format <i>format</i>></pre>
Syntax (TX Matrix Plus Router)	<pre>request system software validate <i>package-name</i> <lcc <i>number</i> sfc <i>number</i>> <on (host <i>host</i> <username <i>username</i>> routing-engine <i>routing-engine</i>)> <set [<i>package-name package-name</i>]> <upgrade-with-config> <upgrade-with-config-format <i>format</i>></pre>
Syntax (MX Series Router)	<pre>request system software validate <<i>package-name</i>> <member <i>member-id</i>> <on (host <i>host</i> <username <i>username</i>> routing-engine <i>routing-engine</i>)> <set [<i>package-name package-name</i>]> <upgrade-with-config> <upgrade-with-config-format <i>format</i>></pre>
Release Information	<p>Command introduced before Junos OS Release 7.4.</p> <p>sfc option introduced for the TX Matrix Plus router in Junos OS Release 9.6.</p> <p>Command introduced in Junos OS Release 11.1 for the QFX Series.</p> <p>set [<i>package-name package-name</i>] option added in Junos OS Release 12.2 for M Series, MX Series, T Series routers.</p> <p>upgrade-with-config and upgrade-with-config-format <i>format</i> options added in Junos OS Release 12.3 for M Series routers, MX Series routers, and T Series routers.</p> <p>Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.</p> <p>on (host <i>host</i> <username <i>username</i>> routing-engine <i>routing-engine</i>) option introduced in Junos OS Release 13.3, Junos OS Release 14.1, and Junos OS Release 15.1.</p>
Description	<p>Validate candidate software against the current configuration of the router or a remote host.</p>

Options **lcc *number***—(TX Matrix routers and TX Matrix Plus routers only) (Optional) On a TX Matrix router, validate the software bundle or package on a specific T640 router (or line-card chassis) that is connected to the TX Matrix router. On a TX Matrix Plus router, validate the software bundle or package for a specific router that is connected to the TX Matrix Plus router.

Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.
- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.
- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

on (host *host* <username *username*> | routing-engine *routing-engine*)—(Optional) Validate the software bundle or package by comparing it to the running configuration on a remote host or Routing Engine. Specify either a host, replacing *host* with the remote hostname, or a Routing Engine, replacing *routing-engine* with the Routing Engine name. If you specify a remote host, you can optionally provide the username to be used to log in to the remote host.

member *member-id*—(MX Series routers only) (Optional) Validate the software bundle or package on the specified member of the Virtual Chassis configuration. For an MX Series Virtual Chassis, replace *member-id* with a value of 0 or 1.

package-name—Name of the software bundle or package to test.

scc—(TX Matrix routers only) (Optional) Validate the software bundle or package for the TX Matrix router (or switch-card chassis).

set [*package-name package-name*]—(M Series, MX Series, T Series routers) (Optional) Install multiple software packages or software add-on packages at the same time.

sfc *number*—(TX Matrix Plus routers only) (Optional) Validate the software bundle or package for the TX Matrix Plus router.

upgrade-with-config—(Optional) Install one or more configuration files.

upgrade-with-config-format *format*—(Optional) Specify the configuration file format, **text** or **xml**. The default format is **text**.



NOTE: The `upgrade-with-config` and `upgrade-with-config-format` options are only available locally on the router or switch. In a routing matrix, the configuration is applied only to the local router and is not propagated to other routers.

The options are validated during the validation process and applied to the router or switch during the upgrade process. If the upgrade process is successful, the options are removed from the configuration. If the upgrade process fails, the configuration file is renamed with the `.failed` suffix.

Additional Information By default, when you issue the `request system software validate` command on a TX Matrix master Routing Engine, all the T640 master Routing Engines that are connected to it are validated. If you issue the same command on the TX Matrix backup Routing Engine, all the T640 backup Routing Engines that are connected to it are upgraded to the same version of software.

Likewise, if you issue the `request system software validate` command on a TX Matrix Plus master Routing Engine, all the T1600 or T4000 master Routing Engines that are connected to it are validated. If you issue the same command on a TX Matrix Plus backup Routing Engine, all the T1600 or T4000 backup Routing Engines that are connected to it are upgraded to the same version of software.

Required Privilege Level maintenance

Related Documentation

- [request system software abort](#)
- [request system software add on page 420](#)
- [request system software delete on page 435](#)
- [request system software rollback on page 439](#)
- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)

List of Sample Output [request system software validate \(Successful Case\) on page 447](#)
[request system software validate \(Failure Case\) on page 448](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

`request system software validate (Successful Case)`

```
user@host> request system software validate /var/sw/pkg/jbundle-5.3I20020124_0520_sjg.tgz
```

```
Checking compatibility with configuration
Initializing...
Using /packages/jbase-5.3I20020122_1901_sjg
Using /var/sw/pkg/jbundle-5.3I20020124_0520_sjg.tgz
Using /var/chroot/var/tmp/jbundle/jbase-5.3I20020124_0520_sjg.tgz
Using /var/chroot/var/tmp/jbundle/jkernel-5.3I20020124_0520_sjg.tgz
Using /var/chroot/var/tmp/jbundle/jcrypto-5.3I20020124_0520_sjg.tgz
Using /var/chroot/var/tmp/jbundle/jpfe-5.3I20020124_0520_sjg.tgz
Using /var/chroot/var/tmp/jbundle/jdocs-5.3I20020124_0520_sjg.tgz
Using /var/chroot/var/tmp/jbundle/jroute-5.3I20020124_0520_sjg.tgz
Validating against /config/juniper.conf.gz
mgd: commit complete

WARNING: cli has been replaced by an updated version:
CLI release 5.3I0 built by sjg on 2002-01-24 05:23:53 UTC
Restart cli using the new version ? [yes,no] (yes)
```

request system software validate (Failure Case)

```
user@host> request system software validate 6.3/
Pushing bundle to lcc0-re0
error: Failed to transfer package to lcc0-re0

user@host> request system software validate test
Pushing bundle to lcc0-re0
Pushing bundle to lcc2-re0

lcc0-re0:
gzip: stdin: not in gzip format
tar: child returned status 1
ERROR: Not a valid package: /var/tmp/test
```

request system software validate on (Junos OS with Upgraded FreeBSD)

Syntax (MX240, MX480, MX960, MX2010, MX2020 Routers only)	request system software validate on <host <i>host-name</i> [username <i>user-name</i>]> <routing-engine (re0 re1)>
Release Information	Command introduced in Junos OS Release 15.1 for MX240, MX480, MX960, MX2010, and MX2020 routers only.
Description	<p>Direct validation of a running configuration is not possible on a device running Junos OS with upgraded FreeBSD. Nevertheless, validation is an important step in the installation of an upgraded operating system. This command allows validation on a device that is not running Junos OS with upgraded FreeBSD.</p> <p>This command validates the current configuration on a Routing Engine that is not running Junos OS with upgraded FreeBSD or a remote host.</p>
Options	<p>The specific options available are:</p> <p>host <i>host-name</i> [username <i>user-name</i>]—Validate the current configuration on a remote host. The host-name is resolved through DNS. Optionally, you can supply a username to employ on the remote host. If you omit the username option, the currently logged-in user-name is sent to the remote host.</p> <p>routing-engine (re0 re1)—Validate the current configuration on another Routing Engine on the same device. The other Routing Engine cannot be running Junos OS with upgraded FreeBSD or the validation does not succeed.</p>
Additional Information	If the authenticity of the remote host cannot be established, you are prompted to continue the validation or not. If you choose not to continue, the validation process does not take place.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none"> • request system reboot (Junos OS with Upgraded FreeBSD) on page 398 • show system snapshot (Junos OS with Upgraded FreeBSD) on page 508 • Understanding Junos OS with Upgraded FreeBSD on page 17
List of Sample Output	request system software validate on host on page 450 request system software validate on routing-engine on page 450
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system software validate on host

```
user@host> request system software validate on host remote-validator
The authenticity of host 'remote-validator (192.168.164.174)' can't be established.
ECDSA key fingerprint is 73:d0:78:ce:8d:09:aa:92:4c:ce:45:52:1d:76:86:b5.
Are you sure you want to continue connecting (yes/no)? yes
Password: *****

Sending /var/tmp/config.tgz to remote-validator...
Validating /var/tmp/config.tgz on remote-validator...
Checking compatibility with configuration
Initializing...
Using jbase-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using jruntime-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using jkernel-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using jroute-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using jcrypto-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using jweb-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using /var/tmp/config.tgz
Hardware Database regeneration succeeded
Validating against /config/juniper.conf.gz
mgd: warning: schema: init: 'logical-systems-vlans' contains-node 'juniper-config
  vlans': not found
mgd: commit complete
Validation succeeded
```

request system software validate on routing-engine

```
user@host> request system software validate on routing-engine re1

Sending /var/tmp/config.tgz to re1...
Validating /var/tmp/config.tgz on re1...
Checking compatibility with configuration
Initializing...
Using jbase-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using jruntime-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using jkernel-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using jroute-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using jcrypto-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using jweb-15.1-20150416.2
Verified manifest signed by PackageDevelopmentEc_2015
Using /var/tmp/config.tgz
Hardware Database regeneration succeeded
Validating against /config/juniper.conf.gz
mgd: warning: schema: init: 'logical-systems-vlans' contains-node 'juniper-config
```

```
vlan1: not found  
mgd: commit complete  
Validation succeeded
```

request system storage cleanup

List of Syntax	Syntax on page 452
	Syntax (EX Series Switches) on page 452
	Syntax (MX Series Router) on page 452
	Syntax (QFX Series) on page 452
	Syntax (SRX Series) on page 452
Syntax	<code>request system storage cleanup</code> <code><dry-run></code> <code><no-confirm></code> <code><re0 re1 routing-engine (backup both local master other)></code>
Syntax (EX Series Switches)	<code>request system storage cleanup</code> <code><all-members></code> <code><dry-run></code> <code><local></code> <code><member <i>member-id</i>></code> <code><no-confirm></code> <code><re0 re1 routing-engine (backup both local master other)></code> <code><satellite [slot-id <i>slot-id</i> device-alias <i>alias-name</i>]></code>
Syntax (MX Series Router)	<code>request system storage cleanup</code> <code><all-members></code> <code><dry-run></code> <code><local></code> <code><member <i>member-id</i>></code> <code><no-confirm></code> <code><re0 re1 routing-engine (backup both local master other)></code> <code><satellite [slot-id <i>slot-id</i> device-alias <i>alias-name</i>]></code>
Syntax (QFX Series)	<code>request system storage cleanup</code> <code><component (<i>serial number</i> <i>UUID</i> all)></code> <code><director-group <i>name</i>></code> <code><dry-run></code> <code><infrastructure <i>name</i>></code> <code><interconnect-device <i>name</i>></code> <code><name-tag <i>name-tag</i>></code> <code><no-confirm></code> <code><node-group <i>name</i>></code> <code><prune></code> <code><qfabric (<i>component name</i>) dry-run name-tag repository></code> <code><repository (core log)></code> <code><re0 re1 routing-engine (backup both local master other)></code>
Syntax (SRX Series)	<code>request system storage cleanup</code> <code><dry-run></code> <code><no-confirm></code> <code><re0 re1 routing-engine (backup both local master other)></code>

- Release Information** Command introduced in Junos OS Release 7.4.
dry-run option introduced in Junos OS Release 7.6.
 Command introduced in Junos OS Release 9.0 for EX Series switches.
 Command introduced in Junos OS Release 9.2 for SRX Series.
 Command introduced in Junos OS Release 11.1 for the QFX Series.
 Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
satellite option introduced in Junos OS Release 14.2R3.
no-confirm and **(re0 | re1 | routing-engine (backup | both | local | master | other))** options introduced in Junos OS 17.3R1.
- Description** Free storage space on the router or switch by rotating log files and proposing a list of files for deletion. User input is required for file deletion. On a QFabric system, you can delete debug files located on individual devices or on the entire QFabric system.
- Options** **all-members**—(EX4200 switches and MX Series routers only) (Optional) Delete files on the Virtual Chassis master Routing Engine only.



NOTE: To delete files on the other members of the Virtual Chassis configuration, log in to each backup Routing Engine and delete the files using the **request system storage cleanup local** command.

component (UUID | serial number | all)—(QFabric systems only) (Optional) Delete files located on individual QFabric system devices or on the entire QFabric system.

director-group name—(QFabric systems only) (Optional) Delete files on the Director group.

dry-run—(Optional) List files proposed for deletion (without deleting them).

infrastructure name—(QFabric systems only) (Optional) Delete files on the fabric control Routing Engine and fabric manager Routing Engine.

interconnect-device name—(QFabric systems only) (Optional) Delete files on the Interconnect device.

local—(EX4200 switches and MX Series routers only) (Optional) Delete files on the local Virtual Chassis member.

member member-id—(EX4200 switches and MX Series routers only) (Optional) Delete files on the specified member of the Virtual Chassis configuration. For EX4200 switches, replace **member-id** with a value from 0 through 9. For an MX Series Virtual Chassis, replace **member-id** with a value of 0 or 1.

name-tag name-tag—(QFabric systems only) (Optional) Delete debug files that match a specific regular expression.

node-group name—(QFabric systems only) (Optional) Delete files on the Node group.

no-confirm—(Optional) Do not ask for confirmation before doing the cleanup.

prune—(QFabric systems only) (Optional) Delete debug files located in either the core or log debug repositories of a QFabric system device.

qfabric component *name*—(QFabric systems only) (Optional) Delete debug files located in the debug repositories of a QFabric system device.

(re0 | re1 | routing-engine (backup | both | local | master | other))—(Optional) Request operation on system storage on RE0, RE1, or on specified Routing Engine by these classifications: backup, both, local, master, or other.

When Routing Engine is specified, the below message is shown before listing the files and deleting them.

```
Please check the list of files to be deleted using the dry-run option. i.e.  
request system storage cleanup dry-run  
Do you want to proceed ? [yes,no] (no)
```

repository (core | log)—(QFabric systems only) (Optional) Specify the repository on the QFabric system device for which you want to delete debug files.

satellite [slot-id *slot-id* | device-alias *alias-name*]—(Junos Fusion only) (Optional) Specify the satellite device in the Junos Fusion by FPC ID or device alias name for which you want to delete debug files.

Additional Information	If logging is configured and being used, the dry-run option rotates the log files. In that case, the output displays the message "Currently rotating log files, please wait." If no logging is currently under way, the output displays only a list of files to delete.
Required Privilege Level	maintenance
List of Sample Output	request system storage cleanup dry-run on page 455 request system storage cleanup on page 456 request system storage cleanup director-group (QFabric Systems) on page 456 request system storage cleanup infrastructure device-name (QFabric Systems) on page 458 request system storage cleanup interconnect-device device-name (QFabric Systems) on page 459 request system storage cleanup node-group group-name (QFabric Systems) on page 460 request system storage cleanup qfabric component device-name (QFabric Systems) on page 461 request system storage cleanup qfabric component device-name repository core (QFabric Systems) on page 461 request system storage cleanup qfabric component all (QFabric Systems) on page 461
Output Fields	Table 46 on page 455 describes the output fields for the request system storage cleanup command. Output fields are listed in the approximate order in which they appear.

Table 46: request system storage cleanup Output Fields

Field Name	Field Description
List of files to delete:	Shows list of files available for deletion.
Size	Size of the core-dump file.
Date	Last core-dump file modification date and time.
Name	Name of the core-dump file.
Directory to delete:	Shows list of directories available for deletion.
Repository scope:	Repository where core-dump files and log files are stored. The core-dump files are located in the core repository, and the log files are located in the log repository. The default Repository scope is shared since both the core and log repositories are shared by all of the QFabric system devices.
Repository head:	Name of the top-level repository location.
Repository name:	Name of the repository: core or log .
Creating list of debug artifacts to be removed under:	Shows location of files available for deletion.
List of debug artifacts to be removed under:	Shows list of files available for deletion.

Sample Output

request system storage cleanup dry-run

```
user@host> 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
	11.4K	Mar 8 15:00	/var/log/messages.1.gz
	7245B	Feb 5 15:00	/var/log/messages.3.gz
	11.8K	Feb 22 13:00	/var/log/messages.2.gz
	3926B	Mar 16 13:57	/var/log/messages.0.gz
	3962B	Feb 22 12:47	/var/log/sampled.1.gz
	4146B	Mar 8 12:20	/var/log/sampled.0.gz
	4708B	Dec 21 11:39	/var/log/sampled.2.gz
	7068B	Jan 16 18:00	/var/log/messages.4.gz
	13.7K	Dec 27 22:00	/var/log/messages.5.gz
	890B	Feb 22 17:22	/var/tmp/sampled.pkts
	65.8M	Oct 26 09:10	/var/sw/pkg/jinstall-7.4R1.7-export-signed.tgz
	63.1M	Oct 26 09:13	/var/sw/pkg/jbundle-7.4R1.7.tgz

request system storage cleanup

```
user@host> 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
11.4K	Mar 8 15:00	/var/log/messages.1.gz	
7245B	Feb 5 15:00	/var/log/messages.3.gz	
11.8K	Feb 22 13:00	/var/log/messages.2.gz	
3926B	Mar 16 13:57	/var/log/messages.0.gz	
11.6K	Mar 8 15:00	/var/log/messages.5.gz	
7254B	Feb 5 15:00	/var/log/messages.6.gz	
12.9K	Feb 22 13:00	/var/log/messages.8.gz	
3726B	Mar 16 13:57	/var/log/messages.7.gz	
3962B	Feb 22 12:47	/var/log/sampled.1.gz	
4146B	Mar 8 12:20	/var/log/sampled.0.gz	
4708B	Dec 21 11:39	/var/log/sampled.2.gz	
7068B	Jan 16 18:00	/var/log/messages.4.gz	
13.7K	Dec 27 22:00	/var/log/messages.5.gz	
890B	Feb 22 17:22	/var/tmp/sampled.pkts	
65.8M	Oct 26 09:10	/var/sw/pkg/jinstall-7.4R1.7-export-signed.tgz	
63.1M	Oct 26 09:13	/var/sw/pkg/jbundle-7.4R1.7.tgz	

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

request system storage cleanup director-group (QFabric Systems)

```
user@switch> request system storage cleanup director-group
List of files to delete:
```

	Size	Date	Name
4.0K	2011-11-07 05:16:29	/tmp/2064.sfcauth	
4.0K	2011-11-07 05:07:34	/tmp/30804.sfcauth	
4.0K	2011-11-07 04:13:41	/tmp/26792.sfcauth	
4.0K	2011-11-07 04:13:39	/tmp/26432.sfcauth	
0	2011-11-07 07:45:40	/tmp/cluster_cleanup.log	
1.3M	2011-11-07 07:39:11	/tmp/cn_monitor.20111107-052401.log	
4.0K	2011-11-07 07:36:29	/tmp/clustat.28019.log	
4.0K	2011-11-07 07:36:29	/tmp/clustat_x.28019.log	
9.6M	2011-11-07 05:30:24	/tmp/sfc.2.log	
4.0K	2011-11-07 05:28:11	/tmp/mgd-init.1320672491.log	
248K	2011-11-07 05:19:24	/tmp/cn_monitor.20111107-045111.log	
4.0K	2011-11-07 05:17:18	/tmp/clustat.3401.log	
4.0K	2011-11-07 05:17:18	/tmp/clustat_x.3401.log	
8.0K	2011-11-07 04:58:25	/tmp/mgd-init.1320670633.log	
0	2011-11-07 04:54:01	/tmp/mysql_db_install_5.1.37.log	
4.0K	2011-11-07 04:52:08	/tmp/cn_send.log	
0	2011-11-07 04:52:00	/tmp/init_eth0.log	
4.0K	2011-11-07 04:49:35	/tmp/install_interfaces.sh.log	
4.0K	2011-11-07 04:48:15	/tmp/bootstrap.sh.log	
160K	2011-11-07 04:47:43	/tmp/bootstrap_cleanup.log	
38M	2011-11-07 04:42:42	/tmp/cn_monitor.20111104-110308.log	
4.0K	2011-11-07 04:38:47	/tmp/clustat.30913.log	
4.0K	2011-11-07 04:38:47	/tmp/clustat_x.30913.log	
4.0K	2011-11-07 04:38:03	/tmp/dcf_upgrade.sh.remove.log	
4.0K	2011-11-07 04:38:03	/tmp/peer_update.log	
4.0K	2011-11-07 04:38:02	/tmp/dcf_upgrade.log	

```

4.0K 2011-11-07 04:38:02 /tmp/perl_mark_upgrade.log
8.0K 2011-11-07 04:13:42 /tmp/install_dcf_rpm.log
4.0K 2011-11-07 04:13:06 /tmp/00_cleanup.sh.1320667986.log
0 2011-11-07 04:13:06 /tmp/ccif_patch_4410_4450.sh.1320667986.log
4.0K 2011-11-07 04:13:06 /tmp/pcf-tools.sh.1320667986.log
0 2011-11-07 04:13:06 /tmp/initial.sh.1320667986.log
0 2011-11-07 04:13:06 /tmp/inventory.sh.1320667986.log
4.0K 2011-11-07 04:13:06 /tmp/qf-db.sh.1320667986.log
4.0K 2011-11-07 04:13:06 /tmp/sfc.sh.1320667986.log
8.0K 2011-11-07 04:13:05 /tmp/jinstall-qfabric.log
8.0K 2011-11-04 11:10:24 /tmp/mgd-init.1320430192.log
4.0K 2011-11-04 11:07:03 /tmp/mysql_dcf_db_install.log
8.0K 2011-11-04 10:55:07 /tmp/ccif_patch_4410_4450.sh.1320429307.log
8.0K 2011-11-04 10:55:07 /tmp/initial.sh.1320429307.log
4.0K 2011-11-04 10:55:07 /tmp/inventory.sh.1320429307.log
8.0K 2011-11-04 10:55:07 /tmp/sfc.sh.1320429307.log
4.0K 2011-11-04 10:54:09 /tmp/ks-script-Ax0tz5.log
4.0K 2011-11-07 04:13:06 /tmp//sfc.sh.1320667986.log
8.0K 2011-11-04 10:55:07 /tmp//sfc.sh.1320429307.log

```

Directory to delete:

```

45M 2011-11-08 10:57:43 /tmp/sfc-captures

```

List of files to delete:

	Size	Date	Name
4.0K	2011-11-08 05:47:47	/tmp/5713.sfcauth	
4.0K	2011-11-08 05:14:32	/tmp/14494.sfcauth	
4.0K	2011-11-08 05:11:47	/tmp/9978.sfcauth	
4.0K	2011-11-08 05:09:37	/tmp/6128.sfcauth	
4.0K	2011-11-08 05:04:28	/tmp/29703.sfcauth	
4.0K	2011-11-07 11:59:10	/tmp/7811.sfcauth	
4.0K	2011-11-07 11:36:08	/tmp/32415.sfcauth	
4.0K	2011-11-07 11:30:30	/tmp/22406.sfcauth	
4.0K	2011-11-07 11:24:37	/tmp/12131.sfcauth	
4.0K	2011-11-07 10:48:42	/tmp/12687.sfcauth	
4.0K	2011-11-07 09:27:20	/tmp/31082.sfcauth	
4.0K	2011-11-07 07:33:58	/tmp/14633.sfcauth	
4.0K	2011-11-07 05:08:25	/tmp/15447.sfcauth	
4.0K	2011-11-07 04:12:29	/tmp/26874.sfcauth	
4.0K	2011-11-07 04:12:27	/tmp/26713.sfcauth	
4.0K	2011-11-07 03:49:17	/tmp/17691.sfcauth	
4.0K	2011-11-05 01:32:23	/tmp/5716.sfcauth	
4.0K	2011-11-07 08:00:17	/tmp/sfcsnmpd.log	
4.0K	2011-11-07 07:57:50	/tmp/cluster_cleanup.log	
824K	2011-11-07 07:38:37	/tmp/cn_monitor.20111107-053643.log	
4.0K	2011-11-07 07:36:30	/tmp/clustat.18399.log	
4.0K	2011-11-07 07:36:30	/tmp/clustat_x.18399.log	
4.0K	2011-11-07 07:35:47	/tmp/command_lock.log	
4.0K	2011-11-07 05:39:54	/tmp/mgd-init.1320673194.log	
92K	2011-11-07 05:19:25	/tmp/cn_monitor.20111107-050412.log	
4.0K	2011-11-07 05:17:20	/tmp/clustat.30115.log	
4.0K	2011-11-07 05:17:20	/tmp/clustat_x.30115.log	
8.0K	2011-11-07 05:08:07	/tmp/mgd-init.1320671241.log	
4.0K	2011-11-07 05:04:57	/tmp/cn_send.log	
0	2011-11-07 05:04:52	/tmp/init_eth0.log	
4.0K	2011-11-07 05:02:38	/tmp/install_interfaces.sh.log	
4.0K	2011-11-07 05:01:19	/tmp/bootstrap.sh.log	
160K	2011-11-07 05:00:47	/tmp/bootstrap_cleanup.log	
28M	2011-11-07 04:42:27	/tmp/cn_monitor.20111104-112954.log	
4.0K	2011-11-07 04:38:49	/tmp/clustat.6780.log	

```

4.0K  2011-11-07 04:38:49 /tmp/clustat_x.6780.log
4.0K  2011-11-07 04:38:05 /tmp/issue_event.log
4.0K  2011-11-07 04:38:05 /tmp/peer_upgrade_reboot.log
12K   2011-11-07 04:38:05 /tmp/primary_update.log
4.0K  2011-11-07 04:38:04 /tmp/dcf_upgrade.sh.remove.log
4.0K  2011-11-07 04:38:04 /tmp/peer_rexec_upgrade.log
4.0K  2011-11-07 04:13:42 /tmp/peer_install_dcf_rpm.log
4.0K  2011-11-07 04:11:57 /tmp/dcf-tools.sh.1320667917.log
0     2011-11-07 04:11:57 /tmp/initial.sh.1320667917.log
0     2011-11-07 04:11:57 /tmp/inventory.sh.1320667917.log
4.0K  2011-11-07 04:11:57 /tmp/qf-db.sh.1320667917.log
4.0K  2011-11-07 04:11:57 /tmp/sfc.sh.1320667917.log
4.0K  2011-11-07 04:11:56 /tmp/00_cleanup.sh.1320667916.log
0     2011-11-07 04:11:56 /tmp/ccif_patch_4410_4450.sh.1320667916.log
8.0K  2011-11-07 04:11:56 /tmp/jinstall-qfabric.log
4.0K  2011-11-07 04:11:33 /tmp/dcf_upgrade.log
8.0K  2011-11-04 11:53:12 /tmp/mgd-init.1320432782.log
8.0K  2011-11-04 11:06:17 /tmp/ccif_patch_4410_4450.sh.1320429977.log
8.0K  2011-11-04 11:06:17 /tmp/initial.sh.1320429977.log
4.0K  2011-11-04 11:06:17 /tmp/inventory.sh.1320429977.log
8.0K  2011-11-04 11:06:17 /tmp/sfc.sh.1320429977.log
4.0K  2011-11-04 11:05:19 /tmp/ks-script_tnWeb.log
4.0K  2011-11-07 04:11:57 /tmp//sfc.sh.1320667917.log
8.0K  2011-11-04 11:06:17 /tmp//sfc.sh.1320429977.log

```

Directory to delete:

```
49M   2011-11-08 10:45:20 /tmp/sfc-captures
```

request system storage cleanup infrastructure device-name (QFabric Systems)

```
user@switch> request system storage cleanup infrastructure FC
re0:
```

List of files to delete:

	Size	Date	Name
	139B	Nov 8 19:03	/var/log/default-log-messages.0.gz
	5602B	Nov 8 19:03	/var/log/messages.0.gz
	28.4K	Nov 8 10:15	/var/log/messages.1.gz
	35.2K	Nov 7 13:45	/var/log/messages.2.gz
	207B	Nov 7 16:02	/var/log/wtmp.0.gz
	27B	Nov 7 12:14	/var/log/wtmp.1.gz
	184.4M	Nov 7 12:16	/var/sw/pkg/jinstall-dc-re-11.3I20111104_1216_dc-builder-domestic-signed.tgz
	124.0K	Nov 7 15:59	/var/tmp/gres-tp/env.dat
	0B	Nov 7 12:57	/var/tmp/gres-tp/lock
	155B	Nov 7 16:02	/var/tmp/krt_gencfg_filter.txt
	0B	Nov 7 12:35	/var/tmp/last_ccif_update
	1217B	Nov 7 12:15	/var/tmp/loader.conf.preinstall
	184.4M	Nov 6 07:11	/var/tmp/mchassis-install.tgz
	10.8M	Nov 7 12:16	/var/tmp/preinstall/bootstrap-install-11.3I20111104_1216_dc-builder.tar
	57.4K	Nov 7 12:16	/var/tmp/preinstall/configs-11.3I20111104_1216_dc-builder.tgz
	259B	Nov 7 12:16	/var/tmp/preinstall/install.conf
	734.3K	Nov 4 13:46	/var/tmp/preinstall/jboot-dc-re-11.3I20111104_1216_dc-builder.tgz
	177.8M	Nov 7 12:16	/var/tmp/preinstall/jbundle-dc-re-11.3I20111104_1216_dc-builder-domestic.tgz
	124B	Nov 7 12:15	/var/tmp/preinstall/metatags

```
1217B Nov 7 12:16 /var/tmp/preinstall_boot_loader.conf
0B Nov 7 16:02 /var/tmp/rtbdb/if-rtbdb
```

request system storage cleanup interconnect-device device-name (QFabric Systems)

```
user@switch> request system storage cleanup interconnect IC
re1:
```

List of files to delete:

Size	Date	Name
11B	Nov 7 15:55	/var/jail/tmp/alarmd.ts
128B	Nov 8 19:06	/var/log/default-log-messages.0.gz
9965B	Nov 8 19:06	/var/log/messages.0.gz
15.8K	Nov 8 12:30	/var/log/messages.1.gz
15.8K	Nov 8 11:00	/var/log/messages.2.gz
15.7K	Nov 8 07:30	/var/log/messages.3.gz
15.8K	Nov 8 04:00	/var/log/messages.4.gz
15.7K	Nov 8 00:30	/var/log/messages.5.gz
18.7K	Nov 7 21:00	/var/log/messages.6.gz
17.6K	Nov 7 19:00	/var/log/messages.7.gz
58.3K	Nov 7 16:00	/var/log/messages.8.gz
20.3K	Nov 7 15:15	/var/log/messages.9.gz
90B	Nov 7 15:41	/var/log/wtmp.0.gz
57B	Nov 7 12:41	/var/log/wtmp.1.gz
124.0K	Nov 7 15:42	/var/tmp/gres-tp/env.dat
0B	Nov 7 12:40	/var/tmp/gres-tp/lock
0B	Nov 7 12:41	/var/tmp/if-rtbdb/env.lck
12.0K	Nov 7 15:41	/var/tmp/if-rtbdb/env.mem
132.0K	Nov 7 15:55	/var/tmp/if-rtbdb/shm_usr1.mem
2688.0K	Nov 7 15:41	/var/tmp/if-rtbdb/shm_usr2.mem
2048.0K	Nov 7 15:41	/var/tmp/if-rtbdb/trace.mem
730B	Nov 7 19:57	/var/tmp/juniper.conf+.gz
155B	Nov 7 15:53	/var/tmp/krt_gencfg_filter.txt
0B	Nov 7 15:41	/var/tmp/rtbdb/if-rtbdb

```
re0:
```

List of files to delete:

Size	Date	Name
11B	Nov 7 15:55	/var/jail/tmp/alarmd.ts
121B	Nov 8 19:06	/var/log/default-log-messages.0.gz
16.7K	Nov 8 19:06	/var/log/messages.0.gz
22.2K	Nov 8 17:45	/var/log/messages.1.gz
18.4K	Nov 8 17:00	/var/log/messages.2.gz
21.6K	Nov 8 16:00	/var/log/messages.3.gz
17.9K	Nov 8 14:30	/var/log/messages.4.gz
19.4K	Nov 8 13:30	/var/log/messages.5.gz
18.2K	Nov 8 12:30	/var/log/messages.6.gz
20.4K	Nov 8 11:30	/var/log/messages.7.gz
21.4K	Nov 8 10:15	/var/log/messages.8.gz
21.0K	Nov 8 09:00	/var/log/messages.9.gz
19.9K	Nov 8 08:13	/var/log/snmp-traps.0.gz
203B	Nov 8 15:36	/var/log/wtmp.0.gz
57B	Nov 7 12:41	/var/log/wtmp.1.gz
124.0K	Nov 7 15:42	/var/tmp/gres-tp/env.dat
0B	Nov 7 12:40	/var/tmp/gres-tp/lock
0B	Nov 7 12:41	/var/tmp/if-rtbdb/env.lck

```

12.0K Nov 7 15:41 /var/tmp/if-rtbdb/env.mem
132.0K Nov 7 15:55 /var/tmp/if-rtbdb/shm_usr1.mem
2688.0K Nov 7 15:41 /var/tmp/if-rtbdb/shm_usr2.mem
2048.0K Nov 7 15:41 /var/tmp/if-rtbdb/trace.mem
727B Nov 7 15:54 /var/tmp/juniper.conf+.gz
155B Nov 7 15:55 /var/tmp/krt_gencfg_filter.txt
0B Nov 7 15:41 /var/tmp/rtbdb/if-rtbdb

```

request system storage cleanup node-group group-name (QFabric Systems)

```

user@switch> request system storage cleanup node-group NW-NG
BBAK0372:

```

List of files to delete:

	Size	Date	Name
	126B	Nov 8 19:07	/var/log/default-log-messages.0.gz
	179B	Nov 7 13:32	/var/log/install.0.gz
	22.9K	Nov 8 19:07	/var/log/messages.0.gz
	26.5K	Nov 8 17:30	/var/log/messages.1.gz
	20.5K	Nov 8 13:15	/var/log/messages.2.gz
	33.2K	Nov 7 17:45	/var/log/messages.3.gz
	35.5K	Nov 7 15:45	/var/log/messages.4.gz
	339B	Nov 8 17:10	/var/log/wtmp.0.gz
	58B	Nov 7 12:40	/var/log/wtmp.1.gz
	124.0K	Nov 8 17:08	/var/tmp/gres-tp/env.dat
	0B	Nov 7 12:39	/var/tmp/gres-tp/lock
	0B	Nov 7 12:59	/var/tmp/if-rtbdb/env.lck
	12.0K	Nov 8 17:09	/var/tmp/if-rtbdb/env.mem
	2688.0K	Nov 8 17:09	/var/tmp/if-rtbdb/shm_usr1.mem
	132.0K	Nov 8 17:09	/var/tmp/if-rtbdb/shm_usr2.mem
	2048.0K	Nov 8 17:09	/var/tmp/if-rtbdb/trace.mem
	1082B	Nov 8 17:09	/var/tmp/juniper.conf+.gz
	155B	Nov 7 17:39	/var/tmp/krt_gencfg_filter.txt
	0B	Nov 8 17:09	/var/tmp/rtbdb/if-rtbdb

EE3093:

List of files to delete:

	Size	Date	Name
	11B	Nov 8 17:33	/var/jail/tmp/alarmd.ts
	119B	Nov 8 19:08	/var/log/default-log-messages.0.gz
	180B	Nov 7 17:41	/var/log/install.0.gz
	178B	Nov 7 13:32	/var/log/install.1.gz
	2739B	Nov 8 19:08	/var/log/messages.0.gz
	29.8K	Nov 8 18:45	/var/log/messages.1.gz
	31.8K	Nov 8 17:15	/var/log/messages.2.gz
	20.6K	Nov 8 16:00	/var/log/messages.3.gz
	15.4K	Nov 8 10:15	/var/log/messages.4.gz
	15.4K	Nov 8 02:15	/var/log/messages.5.gz
	25.5K	Nov 7 20:45	/var/log/messages.6.gz
	48.0K	Nov 7 17:45	/var/log/messages.7.gz
	32.8K	Nov 7 13:45	/var/log/messages.8.gz
	684B	Nov 8 17:02	/var/log/wtmp.0.gz
	58B	Nov 7 12:40	/var/log/wtmp.1.gz
	124.0K	Nov 7 17:34	/var/tmp/gres-tp/env.dat
	0B	Nov 7 12:40	/var/tmp/gres-tp/lock
	0B	Nov 7 12:59	/var/tmp/if-rtbdb/env.lck


```

12.OK Nov 7 17:39 /var/tmp/if-rtbdb/env.mem
2688.OK Nov 7 17:39 /var/tmp/if-rtbdb/shm_usr1.mem
132.OK Nov 7 17:40 /var/tmp/if-rtbdb/shm_usr2.mem
2048.OK Nov 7 17:39 /var/tmp/if-rtbdb/trace.mem
155B Nov 7 17:40 /var/tmp/krt_gencfg_filter.txt
0B Nov 7 17:39 /var/tmp/rtbdb/if-rtbdb

```

request system storage cleanup qfabric component device-name (QFabric Systems)

```

user@switch> request system storage cleanup qfabric component Test
Repository type: regular
Repository head: /pbstorage
Creating list of debug artifacts to be removed under: /pbstorage/rdumps/Test
Removing debug artifacts ... (press control C to abort)
Removing /pbstorage/rdumps/Test/cosd.core.0.0.05162011123308.gz ... done
Removing /pbstorage/rdumps/Test/cosd.core.1.0.05162011123614.gz ... done
Removing /pbstorage/rdumps/Test/cosd.core.2.0.05162011123920.gz ... done
Removing /pbstorage/rdumps/Test/livekcore.05132011163930.gz ... done
Removing /pbstorage/rdumps/Test/tnetd.core.0.1057.05162011124500.gz ... done
Removing /pbstorage/rdumps/Test/vmcore.05132011120528.gz ... done
Removing /pbstorage/rdumps/Test/vmcore.kz ... done
Creating list of debug artifacts to be removed under: /pbstorage/rlogs/Test
Removing debug artifacts ... (press control C to abort)
Removing /pbstorage/rlogs/Test/kdumpinfo.05132011120528 ... done
Removing /pbstorage/rlogs/Test/kernel.tarball.0.1039.05122011234415.tgz ... done
Removing /pbstorage/rlogs/Test/kernel.tarball.1.1039.05132011175544.tgz ... done
Removing /pbstorage/rlogs/Test/tnetd.tarball.0.1057.05162011175453.tgz ... done

```

request system storage cleanup qfabric component device-name repository core (QFabric Systems)

```

user@switch> request system storage cleanup qfabric component Test repository core
Repository scope: shared
Repository head: /pbdata/export
Repository name: core
Creating list of debug artifacts to be removed under: /pbdata/export/rdumps/Test
NOTE: core repository under /pbdata/export/rdumps/Test empty

```

request system storage cleanup qfabric component all (QFabric Systems)

```

user@switch> request system storage cleanup qfabric component all
Repository scope: shared
Repository head: /pbdata/export
Creating list of debug artifacts to be removed under: /pbdata/export/rdumps
NOTE: core repository under /pbdata/export/rdumps/all empty
Creating list of debug artifacts to be removed under: /pbdata/export/rlogs
List of debug artifacts to clean up ... (press control C to abort)
/pbdata/export/rlogs/73747cd8-0710-11e1-b6a4-00e081c5297e/install-11072011125819.log
/pbdata/export/rlogs/77116f18-0710-11e1-a2a0-00e081c5297e/install-11072011125819.log
/pbdata/export/rlogs/BBAK0372/install-11072011121538.log
/pbdata/export/rlogs/BBAK0394/install-11072011121532.log
/pbdata/export/rlogs/EE3093/install-11072011121536.log
/pbdata/export/rlogs/WS001/YN5999/install-11072011121644.log
/pbdata/export/rlogs/WS001/YW3803/install-11072011122429.log
/pbdata/export/rlogs/cd78871a-0710-11e1-878e-00e081c5297e/install-11072011125932.log
/pbdata/export/rlogs/d0afdale-0710-11e1-a1d0-00e081c5297e/install-11072011125930.log
/pbdata/export/rlogs/d0afdale-0710-11e1-a1d0-00e081c5297e/install-11072011133211.log
/pbdata/export/rlogs/d0afdale-0710-11e1-a1d0-00e081c5297e/install-11072011155302.log
/pbdata/export/rlogs/d31ab7a6-0710-11e1-ad1b-00e081c5297e/install-11072011125931.log
/pbdata/export/rlogs/d4d0f254-0710-11e1-90c3-00e081c5297e/install-11072011125932.log

```


request system storage cleanup (SRX Series)

Syntax	<code>request system storage cleanup <dry-run></code>
Release Information	Command introduced in Junos OS Release 9.2 for SRX Series.
Description	Free storage space on the device by rotating log files and proposing a list of files for deletion. User input is required for file deletion.
Options	dry-run —(Optional) List files proposed for deletion (without deleting them).
Additional Information	If logging is configured and being used, the dry-run option rotates the log files. In that case, the output displays the message “Currently rotating log files, please wait.” If no logging is currently under way, the output displays only a list of files to delete.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none"> Cleaning Up Files with the CLI
List of Sample Output	request system storage cleanup dry-run on page 463 request system storage cleanup on page 465
Output Fields	Table 46 on page 455 describes the output fields for the request system storage cleanup command. Output fields are listed in the approximate order in which they appear.

Table 47: request system storage cleanup Output Fields

Field Name	Field Description
List of files to delete:	Shows list of files available for deletion.
Size	Size of the core-dump file.
Date	Last core-dump file modification date and time.
Name	Name of the core-dump file.

Sample Output

request system storage cleanup dry-run

```

user@host> request system storage cleanup dry-run
List of files to delete:

      Size Date      Name
-----
11B Jul 14 22:51 /var/jail/tmp/alarmd.ts

```

```
84.3K Jul 20 22:09 /var/log/chassisd.0.gz
83.0K Jul 20 04:35 /var/log/chassisd.1.gz
84.0K Jul 19 10:52 /var/log/chassisd.2.gz
90.4K Jul 18 17:16 /var/log/chassisd.3.gz
91.8K Jul 20 04:30 /var/log/hostlogs/auth.log.1.gz
93.1K Jul 17 05:45 /var/log/hostlogs/auth.log.2.gz
97.6K Jun 7 01:30 /var/log/hostlogs/auth.log.3.gz
92.0K Apr 25 15:15 /var/log/hostlogs/auth.log.4.gz
78.0K Jul 21 05:44 /var/log/hostlogs/daemon.log.1.gz
78.6K Jul 21 02:59 /var/log/hostlogs/daemon.log.2.gz
78.5K Jul 21 00:14 /var/log/hostlogs/daemon.log.3.gz
78.8K Jul 20 21:30 /var/log/hostlogs/daemon.log.4.gz
58.7K Jul 21 05:14 /var/log/hostlogs/debug.1.gz
58.5K Jul 21 00:59 /var/log/hostlogs/debug.2.gz
58.7K Jul 20 20:44 /var/log/hostlogs/debug.3.gz
58.7K Jul 20 16:29 /var/log/hostlogs/debug.4.gz
166.9K Jul 13 00:33 /var/log/hostlogs/kern.log.1.gz
166.5K Jun 1 02:32 /var/log/hostlogs/kern.log.2.gz
163.5K May 5 00:03 /var/log/hostlogs/kern.log.3.gz
152.3K Mar 2 23:23 /var/log/hostlogs/kern.log.4.gz
260.0K Apr 13 10:28 /var/log/hostlogs/lcmd.log.1.gz
257.3K Mar 7 00:38 /var/log/hostlogs/lcmd.log.2.gz
240.8K Feb 7 19:45 /var/log/hostlogs/lcmd.log.3.gz
241.1K Feb 7 14:00 /var/log/hostlogs/lcmd.log.4.gz
370.6K Jul 21 00:45 /var/log/hostlogs/syslog.1.gz
370.9K Jul 20 12:30 /var/log/hostlogs/syslog.2.gz
370.4K Jul 20 00:15 /var/log/hostlogs/syslog.3.gz
370.2K Jul 19 12:00 /var/log/hostlogs/syslog.4.gz
55.0K Jul 14 22:50 /var/log/hostlogs/vjunos0.log.1.gz
1467B Oct 28 2015 /var/log/install.0.gz
119.9K Jul 21 07:37 /var/log/messages.0.gz
147.4K May 27 01:30 /var/log/messages.1.gz
71.4K Apr 14 11:19 /var/log/messages.2.gz
90.7K Feb 28 14:15 /var/log/messages.3.gz
10.1K Jan 12 2016 /var/log/messages.4.gz
55.1K Jan 6 2016 /var/log/messages.5.gz
81.5K Dec 1 2015 /var/log/messages.6.gz
43.3K Oct 28 2015 /var/log/messages.7.gz
54.8K Oct 20 2015 /var/log/messages.8.gz
35.8K Oct 19 2015 /var/log/messages.9.gz
12.4K Jul 21 07:37 /var/log/security.0.gz
59.4K Jul 19 01:30 /var/log/security.1.gz
51.8K Apr 25 10:00 /var/log/security.2.gz
43.6K Apr 14 11:19 /var/log/security.3.gz
52.7K Apr 5 02:15 /var/log/security.4.gz
54.4K Mar 25 17:15 /var/log/security.5.gz
51.9K Mar 16 05:15 /var/log/security.6.gz
52.0K Mar 5 02:15 /var/log/security.7.gz
53.4K Feb 22 22:15 /var/log/security.8.gz
55.6K Feb 13 13:00 /var/log/security.9.gz
4063B Jul 14 22:51 /var/tmp/cleanup-pkgs.log
0B Jul 14 22:51 /var/tmp/eedebg_bin_file
50.9K Feb 8 20:33 /var/tmp/event_tags.php
34B Jul 14 22:51 /var/tmp/gksdchk.log
124.0K Apr 26 06:12 /var/tmp/gres-tp/env.dat
0B Oct 9 2015 /var/tmp/gres-tp/lock
4B Jul 14 22:52 /var/tmp/idp_license_info
46B Jul 14 22:51 /var/tmp/kmdchk.log
57B Jul 14 22:51 /var/tmp/krt_rpf_filter.txt
30B Jul 14 22:53 /var/tmp/policy_status
0B Jul 14 22:51 /var/tmp/rtsdb/if-rtsdb
```

```

349B Jul 14 22:51 /var/tmp/sd-upgrade/debug_log
  0B Oct  9 2015 /var/tmp/spu_kmd_init
53B Feb  7 23:11 /var/tmp/vjunos-install.log
  0B Jul 14 22:51 /var/tmp/vpn_tunnel_orig.id

```

request system storage cleanup

```

user@host> request system storage cleanup
List of files to delete:

```

Size	Date	Name
11B	Oct 28 23:40	/var/jail/tmp/alarmd.ts
92.4K	Jan 11 17:12	/var/log/chassisd.0.gz
92.4K	Jan 11 06:06	/var/log/chassisd.1.gz
92.5K	Jan 10 19:00	/var/log/chassisd.2.gz
92.5K	Jan 10 07:53	/var/log/chassisd.3.gz
92.2K	Jan 10 15:00	/var/log/hostlogs/auth.log.1.gz
92.2K	Jan 1 18:45	/var/log/hostlogs/auth.log.2.gz
92.1K	Jan 4 17:30	/var/log/hostlogs/auth.log.3.gz
92.2K	Jan 1 18:45	/var/log/hostlogs/auth.log.4.gz
79.0K	Jan 12 01:59	/var/log/hostlogs/daemon.log.1.gz
78.8K	Jan 11 23:15	/var/log/hostlogs/daemon.log.2.gz
78.7K	Jan 11 20:30	/var/log/hostlogs/daemon.log.3.gz
79.1K	Jan 11 17:44	/var/log/hostlogs/daemon.log.4.gz
59.1K	Jan 11 21:59	/var/log/hostlogs/debug.1.gz
59.2K	Jan 11 17:44	/var/log/hostlogs/debug.2.gz
59.2K	Jan 11 13:29	/var/log/hostlogs/debug.3.gz
59.3K	Jan 11 09:14	/var/log/hostlogs/debug.4.gz
186.6K	Oct 20 16:31	/var/log/hostlogs/kern.log.1.gz
238.3K	Jan 11 23:15	/var/log/hostlogs/lcmd.log.1.gz
238.4K	Jan 11 17:30	/var/log/hostlogs/lcmd.log.2.gz
238.6K	Jan 11 11:45	/var/log/hostlogs/lcmd.log.3.gz
238.5K	Jan 11 06:00	/var/log/hostlogs/lcmd.log.4.gz
372.5K	Jan 11 17:00	/var/log/hostlogs/syslog.1.gz
372.5K	Jan 11 04:45	/var/log/hostlogs/syslog.2.gz
371.9K	Jan 10 16:30	/var/log/hostlogs/syslog.3.gz
372.7K	Jan 10 04:15	/var/log/hostlogs/syslog.4.gz
10.1K	Jan 12 02:03	/var/log/messages.0.gz
55.1K	Jan 6 21:25	/var/log/messages.1.gz
81.5K	Dec 1 21:30	/var/log/messages.2.gz

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

request system zeroize

Syntax request system zeroize
 <media>
 <local>

Release Information Command introduced before Junos OS Release 9.0.
 Command introduced in Junos OS Release 11.2 for EX Series switches.
 Option **media** added in Junos OS Release 11.4 for EX Series switches.
 Command introduced in Junos OS Release 12.2 for MX Series routers.
 Command introduced in Junos OS Release 12.3 for the QFX Series.
 Option **local** added in Junos OS Release 14.1.
 Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Remove all configuration information on the Routing Engines and reset all key values on the device where you run the command.

- If the device has dual Routing Engines, the command is broadcast to all Routing Engines on the device.
- In a Virtual Chassis or Virtual Chassis Fabric (VCF) composed of EX Series switches (except EX8200 Virtual Chassis) or QFX Series switches, this command operates only on the member switch where you run the command, even if that switch is in the master Routing Engine role. The command is not forwarded to the backup Routing Engine member or to member switches in the line-card role. To apply this command to more than one member of an EX Series or QFX Series Virtual Chassis or VCF, we recommend you remove and disconnect each of those members from the Virtual Chassis or VCF, and then run the command on each isolated switch individually.

The command removes all data files, including customized configuration and log files, by unlinking the files from their directories. The command removes all user-created files from the system, including all plain-text passwords, secrets, and private keys for SSH, local encryption, local authentication, IPsec, RADIUS, TACACS+, and SNMP.

This command reboots the device and sets it to the factory default configuration. After the reboot, you cannot access the device through the management Ethernet interface. Log in through the console as **root** and start the Junos OS CLI by typing **cli** at the prompt.



NOTE: If the configuration contains the **commit synchronize** statement at the **[edit system]** hierarchy level, and you issue a **commit** in the master Routing Engine, the master configuration is automatically synchronized with the backup. If the backup Routing Engine is down when you issue the **commit**, the Junos OS displays a warning and commits the candidate configuration in the master Routing Engine. When the backup Routing Engine comes up, its configuration will automatically be synchronized with the master. A newly inserted backup Routing Engine or a Routing Engine that comes up after running the **request system zeroize** command also automatically synchronizes its configuration with the master Routing Engine configuration.



NOTE: Starting with Junos OS Release 15.1F3, the `request system zeroize` command removes all configuration information on the guest OS for the PTX5000 router with RE-DUO-C2600-16G, and MX240, MX480, and MX960 with RE-S-1800X4-32G-S.

Starting with Junos OS Release 15.1F5, the `request system zeroize` command removes all configuration information on the guest OS for the MX2010 and MX2020 with REMX2K-1800-32G-S.

On these routers, in order to remove all configuration information on both guest OS and host OS, use the `request vmhost zeroize` command.

To completely erase user-created data so that it is unrecoverable, use the **media** option.

Options **media**—(Optional) In addition to removing all configuration and log files, causes memory and the media to be scrubbed, removing all traces of any user-created files. Every storage device attached to the system is scrubbed, including disks, flash drives, removable USBs, and so on. The duration of the scrubbing process is dependent on the size of the media being erased. As a result, the `request system zeroize media` operation can take considerably more time than the `request system zeroize` operation. However, the critical security parameters are all removed at the beginning of the process.



NOTE: On QFX Series platforms running Junos OS Release 14.1X53 or earlier, the **media** option is not available. On QFX Series platforms running releases later than Junos OS Release 14.1X53 that do not have the FreeBSD 10 kernel, the **media** option is available, but if you use it, the system will issue a warning that the **media** option is not supported and will continue with the zeroize operation. On platforms that are not QFX Series platforms, the **media** option is not available in Junos OS Release 17.2 or later with the FreeBSD 10 kernel.

local—(Optional) Remove all the configuration information and restore all the key values on the active Routing Engine.



NOTE: Specifying this option has no effect on switches in a Virtual Chassis or VCF composed of EX Series switches (except EX8200 Virtual Chassis) or QFX switches, because in these configurations, the `request system zeroize` command only operates locally by default.

Required Privilege Level maintenance

Related Documentation

- [request system snapshot on page 406](#)
- *Reverting to the Default Factory Configuration for the EX Series Switch*
- *Reverting to the Rescue Configuration for the EX Series Switch*
- *Reverting to the Default Factory Configuration*
- [Reverting to the Rescue Configuration on page 225](#)
- [Reverting to the Default Factory Configuration by Using the request system zeroize Command on page 225](#)

List of Sample Output [request system zeroize on page 468](#)
[request system zeroize media on page 469](#)

Sample Output

request system zeroize

```
user@host> request system zeroize
warning: System will be rebooted and may not boot without configuration
Erase all data, including configuration and log files? [yes,no] (no) yes

0 1 1 0 0 0 done

syncing disks... All buffers synced.
Uptime: 5d19h20m26s
recorded reboot as normal shutdown
Rebooting...

U-Boot 1.1.6 (Mar 11 2011 - 04:39:06)

Board: EX4200-24T 2.11
EPLD: Version 6.0 (0x85)
DRAM: Initializing (1024 MB)
FLASH: 8 MB

Firmware Version: --- 01.00.00 ---
USB: scanning bus for devices... 2 USB Device(s) found
      scanning bus for storage devices... 1 Storage Device(s) found

ELF file is 32 bit
Consoles: U-Boot console

FreeBSD/PowerPC U-Boot bootstrap loader, Revision 2.4
(user@device.example.net, Fri Mar 11 03:03:36 UTC 2011)
Memory: 1024MB
bootsequencing is enabled
bootsuccess is set
new boot device = disk0s1:
Loading /boot/defaults/loader.conf
/kernel data=0x915c84+0xa1260 syms=[0x4+0x7cbd0+0x4+0xb1c19]

Hit [Enter] to boot immediately, or space bar for command prompt.
```



```

Booting [/kernel]...
Kernel entry at 0x800000e0 ...
GDB: no debug ports present
KDB: debugger backends: ddb
KDB: current backend: ddb
Copyright (c) 1996-2011, Juniper Networks, Inc.
All rights reserved.
Copyright (c) 1992-2006 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
    The Regents of the University of California. All rights reserved.
JUNOS 11.1R1.8 #0: 2011-03-09 20:14:25 UTC

user@device.example.net:/volume/build/junos/11.1/release/11.1R1.8/obj-powerpc/bsd/kernels/
JUNIPER-EX/kernel
Timecounter "decrementer" frequency 50000000 Hz quality 0
cpu0: Freescale e500v2 core revision 2.2
cpu0: HID0 80004080
...

```

request system zeroize media

```

user@host> request system zeroize media
warning: System will be rebooted and may not boot without configuration
Erase all data, including configuration and log files? [yes,no] (no) yes

warning: ipsec-key-management subsystem not running - not needed by configuration.
warning: zeroizing fpc0

{master:0}
root> Waiting (max 60 seconds) for system process `vnlrn' to stop...done
...
Syncing disks, vnodes remaining...2 4 2 4 3 2 1 1 0 0 0 done

syncing disks... All buffers synced.
Uptime: 14m50s
recorded reboot as normal shutdown
Rebooting...

U-Boot 1.1.6 (Apr 21 2011 - 13:58:42)

Board: EX4200-48PX 1.1
EPLD: Version 8.0 (0x82)
DRAM: Initializing (512 MB)
FLASH: 8 MB
NAND: No NAND device found!!!
0 MiB

Firmware Version: --- 01.00.00 ---
USB: scanning bus for devices... 2 USB Device(s) found
      scanning bus for storage devices... 1 Storage Device(s) found

ELF file is 32 bit
Consoles: U-Boot console

FreeBSD/PowerPC U-Boot bootstrap loader, Revision 2.2
(user@device1.example.com, Fri Feb 26 17:48:51 PST 2010)
Memory: 512MB
Loading /boot/defaults/loader.conf
/kernel data=0x9abfdc+0xb06e4 syms=[0x4+0x83b30+0x4+0xbd7c6]

Hit [Enter] to boot immediately, or space bar for command prompt.

```

```
Booting [/kernel] in 1 second... Booting [/kernel]...
Kernel entry at 0x800000e0 ...
GDB: no debug ports present
KDB: debugger backends: ddb
KDB: current backend: ddb
Copyright (c) 1996-2011, Juniper Networks, Inc.
All rights reserved.
Copyright (c) 1992-2006 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
The Regents of the University of California. All rights reserved.
JUNOS 11.4R1.2 #0: 2011-10-27 18:05:39 UTC
user@device1.example.com:/volume/build/junos/11.4/release/11.4R1.2/obj-powerpc/
bsd/kernels/JUNIPER-EX/kernel
can't re-use a leaf (all_slot_serialid)!
Timecounter "decrementer" frequency 50000000 Hz quality 0
cpu0: Freescale e500v2 core revision 2.2
cpu0: HID0 80004080<EMCP,TBEN,EN_MAS7_UPDATE>
real memory = 511705088 (488 MB)
avail memory = 500260864 (477 MB)
ETHERNET SOCKET BRIDGE initialising
Initializing EXSERIES platform properties ...
. . .
Automatic reboot in progress...
Media check on da0 on ex platforms
** /dev/da0s2a
FILE SYSTEM CLEAN; SKIPPING CHECKS
clean, 20055 free (31 frags, 2503 blocks, 0.0% fragmentation)
zeroizing /dev/da0s1a ...
. . .
zeroizing /dev/da0s3d ...
. . .
zeroizing /dev/da0s3e ...
. . .
zeroizing /dev/da0s4d ...
. . .
zeroizing /dev/da0s4e ...
. . .

syncing disks... All buffers synced.
Uptime: 3m40s
Rebooting...

U-Boot 1.1.6 (Apr 21 2011 - 13:58:42)

Board: EX4200-48PX 1.1
EPLD: Version 8.0 (0x82)
DRAM: Initializing (512 MB)
FLASH: 8 MB
NAND: No NAND device found!!!
0 MiB

Firmware Version: --- 01.00.00 ---
USB: scanning bus for devices... 2 USB Device(s) found
      scanning bus for storage devices... 1 Storage Device(s) found

ELF file is 32 bit
Consoles: U-Boot console

FreeBSD/PowerPC U-Boot bootstrap loader, Revision 2.2
(user@device1.example.com, Fri Feb 26 17:48:51 PST 2010)
Memory: 512MB
```

```

Loading /boot/defaults/loader.conf
/kernel data=0x9abfdc+0xb06e4 syms=[0x4+0x83b30+0x4+0xbd7c6]

Hit [Enter] to boot immediately, or space bar for command prompt.
Booting [/kernel] in 1 second... Booting [/kernel]...
Kernel entry at 0x800000e0 ...
GDB: no debug ports present
KDB: debugger backends: ddb
KDB: current backend: ddb
Copyright (c) 1996-2011, Juniper Networks, Inc.
All rights reserved.
Copyright (c) 1992-2006 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
The Regents of the University of California. All rights reserved.
JUNOS 11.4R1.2 #0: 2011-10-27 18:05:39 UTC
user@device1.example.com:/volume/build/junos/11.4/release/11.4R1.2/obj-powerpc/
bsd/kernels/JUNIPER-EX/kernel
can't re-use a leaf (all_slot_serialid)!
Timecounter "decrementer" frequency 500000000 Hz quality 0
cpu0: Freescale e500v2 core revision 2.2
cpu0: H1D0 80004080 <EMCP,TBEN,EN_MAS7_UPDATE>
real memory = 511705088 (488 MB)
avail memory = 500260864 (477 MB)
ETHERNET SOCKET BRIDGE initialising
Initializing EXSERIES platform properties ...
. . .
Automatic reboot in progress...
Media check on da0 on ex platforms
** /dev/da0s1a
FILE SYSTEM CLEAN; SKIPPING CHECKS
clean, 20064 free (48 frags, 2502 blocks, 0.1% fragmentation)
zeroizing /dev/da0s2a ...
. . .
Creating initial configuration...mgd: error: Cannot open configuration file:
/config/juniper.conf
mgd: warning: activating factory configuration
mgd: commit complete
mgd: -----
mgd: Please login as 'root'. No password is required.
mgd: To start Initial Setup, type 'ezsetup' at the JUNOS prompt.
mgd: To start JUNOS CLI, type 'cli' at the JUNOS prompt.
mgd: -----
Setting initial options: debugger_on_panic=NO debugger_on_break=NO.
Starting optional daemons: .
Doing initial network setup:
. . .

Amnesiac (ttyu0)

```

request vmhost halt

Syntax request vmhost halt
<re0 | re1>
<routing-engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Stop the host OS and Junos OS running on the router.

Options none—Stop the host OS and Junos OS on the router.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, stop the host OS and Junos OS running on the Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine on which the host OS and Junos OS needs to be stopped. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level maintenance

Related Documentation

- [request system halt](#)
- [vmhost on page 361](#)

List of Sample Output [request vmhost halt on page 473](#)

Sample Output

request vmhost halt

```
user@host> request vmhost halt
Halt the system ? [yes,no] (no) yes

Initiating vmhost halt... ok
Initiating Junos shutdown... shutdown: [pid 9756]
Shutdown NOW!
ok
Junos shutdown is in progress...
*** FINAL System shutdown message ***

System going down IMMEDIATELY

...
...
Operating system halted.
Please press any key to reboot.
```

request vmhost reboot

Syntax request vmhost reboot
<disk1>
<disk2>
<network>
<re0 | re1>
<routing engine>
<usb>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Reboot both the Junos OS and the host OS running on the router.

Options none—Reboot the router software immediately.

disk1—(Optional) Reboot both Junos OS and the host OS on the Routing Engine and boot the Routing Engine from the primary disk.

disk2—(Optional) Reboot both Junos OS and the host OS on the Routing Engine and boot the Routing Engine from backup disk.

network—(Optional) Reboot both Junos OS and the host OS on the Routing Engine and boot the Routing Engine from network by using the PXE boot method.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, reboot both Junos OS and the host OS on the Routing Engine in slot 0 (**re0**) or on the the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine on which Junos OS and the host OS are to be rebooted. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Both Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

usb—(Optional) Reboot both Junos OS and the host OS on the Routing Engine using the USB installation media.

Required Privilege Level maintenance

Related Documentation

- [request system reboot on page 389](#)
- [request vmhost power-on on page 529](#)

List of Sample Output [request vmhost reboot on page 475](#)

Sample Output

[request vmhost reboot](#)

```
user@host> request vmhost reboot
warning: Rebooting re0
Initiating vmhost reboot... ok
Initiating Junos shutdown... shutdown: [pid 3957]
Shutdown NOW!
ok

*** FINAL System shutdown message ***

System going down IMMEDIATELY
...
...
```

show chassis usb storage

Syntax	show chassis usb storage
Release Information	Command introduced in Junos OS Release 11.4 R2.
Description	Display the current status of any USB mass storage device and whether the USB ports are enabled or disabled.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none">• Installing Junos OS on SRX Series Devices Using a USB Flash Drive on page 77
List of Sample Output	show chassis hardware detail on page 476 show chassis usb storage on page 476

Sample Output

show chassis hardware detail

```
user@host> show chassis hardware detail
Hardware inventory:
Item          Version  Part number  Serial number  Description
Chassis
Routing Engine REV 01    750-043613   AAEC1923      RE-SRX240H2-POE
  usb0 (addr 1) DWC OTG root hub 0   vendor 0x0000  uhub0
  usb0 (addr 2) product 0x005a 90    vendor 0x0409  uhub1
  usb0 (addr 3) ST72682 High Speed Mode 64218 STMicroelectronics umass0
  usb0 (addr 4) Mass Storage Device 4096 JetFlash    umass1
FPC 0
  PIC 0
Power Supply 0 16x GE Base PIC
```

show chassis usb storage

```
user@host> show chassis usb storage
USB Disabled
```


show system autoinstallation status

Syntax	show system autoinstallation status
Release Information	<p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>Command supported in Junos OS Release 12.2 for ACX Series Universal Access Routers.</p> <p>Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.</p>
Description	(ACX Series routers, and EX Series switches only) Display autoinstallation status information.
Options	This command has no options.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • <i>ACX Series Autoinstallation Overview</i> • <i>Before You Begin Autoinstallation on an ACX Series Universal Access Router</i> • <i>Autoinstallation Configuration of ACX Series Universal Access Routers</i> • <i>USB Autoinstallation on ACX Series Routers</i> • <i>Verifying Autoinstallation on ACX Series Universal Access Routers</i> • <i>Verifying Autoinstallation on ACX Series Universal Access Routers</i> • <i>autoinstallation</i>
List of Sample Output	show system autoinstallation status on page 478
Output Fields	Table 48 on page 478 describes the output fields for the show system autoinstallation status command. Output fields are listed in the approximate order in which they appear.

Table 48: show system autoinstallation status Output Fields

Field Name	Field Description
Autoinstallation status	Display autoinstallation status information: <ul style="list-style-type: none">• Last committed file—File last committed for autoinstallation configuration.• Configuration server of last committed file—IP address or URL of the server configured to retrieve configuration information for the last committed configuration file.• Interface—Interface configured for autoinstallation.<ul style="list-style-type: none">• Name—Name of the interface.• State—Interface state.• Address acquisition—Display IP address acquired and protocol used for acquisition upon startup.<ul style="list-style-type: none">• Protocol—Protocol used for acquisition: BOOTP/DHCP or RARP.• Acquired address—IP address acquired from the DHCP server.

Sample Output

show system autoinstallation status

```
user@host> show system autoinstallation status
Autoinstallation status:
Master state: Active
Last committed file: None
Configuration server of last committed file: 10.0.0.0
Interface:
  Name: ge-0/0/1
  State: None
Address acquisition:
  Protocol: DHCP Client
  Acquired address: None
  Protocol: RARP Client
  Acquired address: None
```

show system autorecovery state

Syntax	show system autorecovery state
Release Information	Command introduced in Junos OS Release 11.2 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Description	Perform checks and show status of all autorecovered items.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> request system autorecovery state on page 370
List of Sample Output	show system autorecovery state on page 479
Output Fields	Table 49 on page 479 lists the output fields for the show system autorecovery state command. Output fields are listed in the approximate order in which they appear.

Table 49: show system autorecovery state Output Fields

Field Name	Field Description
File	The name of the file on which autorecovery checks are performed.
Slice	The disk partition on which autorecovery checks are performed.
Recovery Information	Indicates whether autorecovery information for the file or slice has been saved.
Integrity Check	Displays the status of the file's integrity check (passed or failed).
Action / Status	Displays the status of the item, or the action required to be taken for that item.

Sample Output

show system autorecovery state

```

user@host> show system autorecovery state

Configuration:
File          Recovery Information  Integrity Check  Action / Status
rescue.conf.gz Saved                Passed           None
Licenses:
File          Recovery Information  Integrity Check  Action / Status
JUNOS282736.lic Saved                Passed           None
JUNOS282737.lic Not Saved           Not checked     Requires save
BSD Labels:
Slice         Recovery Information  Integrity Check  Action / Status
s1            Saved                Passed           None
s2            Saved                Passed           None

```

s3
s4

Saved
Saved

Passed
Passed

None
None

show system boot-messages

List of Syntax	Syntax on page 481 Syntax (EX Series Switches) on page 481 Syntax (TX Matrix Router) on page 481 Syntax (TX Matrix Plus Router) on page 481 Syntax (MX Series Router) on page 481 Syntax (QFX Series) on page 481
Syntax	show system boot-messages
Syntax (EX Series Switches)	show system boot-messages <all-members> <local> <member <i>member-id</i> >
Syntax (TX Matrix Router)	show system boot-messages <all-chassis all-lcc lcc <i>number</i> scc>
Syntax (TX Matrix Plus Router)	show system boot-messages <all-chassis all-lcc lcc <i>number</i> sfc <i>number</i> >
Syntax (MX Series Router)	show system boot-messages <all-members> <local> <member <i>member-id</i> >
Syntax (QFX Series)	show system boot-messages infrastructure <i>name</i> interconnect-device <i>name</i> node-group <i>name</i>
Release Information	Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 9.0 for EX Series switches. sfc option introduced for the TX Matrix Plus router in Junos OS Release 9.6. Command introduced in Junos OS Release 11.1 for the QFX Series. Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
Description	Display initial messages generated by the system kernel upon startup. These messages are the contents of /var/run/dmesg.boot .
Options	none —Display all boot time messages. all-chassis —(TX Matrix routers and TX Matrix Plus routers only) (Optional) Display boot time messages for all of the chassis. all-lcc —(TX Matrix routers and TX Matrix Plus routers only) (Optional) On a TX Matrix router, display boot time messages for all T640 routers connected to a TX Matrix

router. On a TX Matrix Plus router, display boot time messages for all connected T1600 or T4000 LCCs.

all-members—(EX4200 switches and MX Series routers only) (Optional) Display boot time messages on all members of the Virtual Chassis configuration.

infrastructure *name*—(QFabric systems only) (Optional) Display boot time messages on the fabric control Routing Engine or fabric manager Routing engines.

interconnect-device *name*—(QFabric systems only) (Optional) Display boot time messages on the Interconnect device.

lcc *number*—(TX Matrix routers and TX Matrix Plus routers only) (Optional) On a TX Matrix router, display boot time messages for a specific T640 router connected to a TX Matrix router. On a TX Matrix Plus router, display boot time messages for a specific router connected to a TX Matrix Plus router.

Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.
- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.
- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

local—(EX4200 switches and MX Series routers only) (Optional) Display boot time messages on the local Virtual Chassis member.

member *member-id*—(EX4200 switches and MX Series routers only) (Optional) Display boot time messages on the specified member of the Virtual Chassis configuration. For EX4200 switches, replace *member-id* with a value from 0 through 9. For an MX Series Virtual Chassis, replace *member-id* with a value of 0 or 1.

node-group *name*—(QFabric systems only) (Optional) Display boot time messages on the Node group.

scc—(TX Matrix routers only) (Optional) Display boot time messages for the TX Matrix router (or switch-card chassis).

sfc *number*—(TX Matrix Plus routers only) (Optional) Display boot time messages for the TX Matrix Plus router. Replace *number* with 0.

Additional Information By default, when you issue the **show system boot-messages** command on the master Routing Engine of a TX Matrix router or a TX Matrix Plus router, the command is broadcast to all the master Routing Engines of the LCCs connected to it in the routing matrix. Likewise, if you issue the same command on the backup Routing Engine of a TX Matrix

or a TX Matrix Plus router, the command is broadcast to all backup Routing Engines of the LCCs that are connected to it in the routing matrix.

Required Privilege Level

view

Related Documentation

- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)

List of Sample Output

[show system boot-messages \(TX Matrix Router\) on page 483](#)
[show system boot-messages lcc \(TX Matrix Router\) on page 484](#)
[show system boot-messages \(TX Matrix Plus Router\) on page 485](#)
[show system boot-messages \(QFX3500 Switch\) on page 485](#)

Sample Output

show system boot-messages (TX Matrix Router)

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

JUNOS 4.1-20000216-Zf8469 #0: 2000-02-16 12:57:28 UTC
  tlim@device1.example.com:/p/build/20000216-0905/4.1/release_kernel/sys/compile/GENERIC
CPU: Pentium Pro (332.55-MHz 686-class CPU)
  Origin = "GenuineIntel" Id = 0x66a Stepping=10
  Features=0x183f9ff<FPU,VME,DE,PSE,TSC,MSR,PAE,MCE,CX8,SEP,MTRR,PGE,MCA,CMOV,<b16>,<b17>,MMX,<b24>>
Teknor CPU Card Recognized
real memory = 805306368 (786432K bytes)
avail memory = 786280448 (767852K bytes)
Probing for devices on PCI bus 0:
chip0 <generic PCI bridge (vendor=8086 device=7192 subclass=0)> rev 3 class 6000
0 on pci0:0:0
chip1 <Intel 82371AB PCI-ISA bridge> rev 1 class 60100 on pci0:7:0
chip2 <Intel 82371AB IDE interface> rev 1 class 10180 on pci0:7:1
chip3 <Intel 82371AB USB interface> rev 1 class c0300 int d irq 11 on pci0:7:2
smb0 <Intel 82371AB SMB controller> rev 1 class 68000 on pci0:7:3
pcic0 <TI PCI-1131 PCI-CardBus Bridge> rev 1 class 60700 int a irq 15 on pci0:13:0
TI1131 PCI Config Reg: [pci only][FUNC0 pci int]
pcic1 <TI PCI-1131 PCI-CardBus Bridge> rev 1 class 60700 int b irq 12 on pci0:13:1
TI1131 PCI Config Reg: [pci only][FUNC1 pci int]
fxp0 <Intel EtherExpress Pro 10/100B Ethernet> rev 8 class 20000 int a irq 12 on

pci0:16:0
chip4 <generic PCI bridge (vendor=1011 device=0022 subclass=4)> rev 4 class 6040
0 on pci0:17:0
fxp1 <Intel EtherExpress Pro 10/100B Ethernet> rev 8 class 20000 int a irq 10 on

pci0:19:0
Probing for devices on PCI bus 1:
```

```

mcs0 <Miscellaneous Control Subsystem> rev 12 class ff0000 int a irq 12 on pci1:
13:0
fxp2 <Intel EtherExpress Pro 10/100B Ethernet> rev 8 class 20000 int a irq 10 on

pci1:14:0
Probing for devices on the ISA bus:
sc0 at 0x60-0x6f irq 1 on motherboard
sc0: EGA color <16 virtual consoles, flags=0x0>
ed0 not found at 0x300
ed1 not found at 0x280
ed2 not found at 0x340
psm0 not found at 0x60
sio0 at 0x3f8-0x3ff irq 4 flags 0x20010 on isa
sio0: type 16550A, console
sio1 at 0x3e8-0x3ef irq 5 flags 0x20000 on isa
sio1: type 16550A
sio2 at 0x2f8-0x2ff irq 3 flags 0x20000 on isa
sio2: type 16550A
pcic0 at 0x3e0-0x3e1 on isa
PC-Card ctlr(0) TI PCI-1131 [CardBus bridge mode] (5 mem & 2 I/O windows)
pcic0: slot 0 controller I/O address 0x3e0
npx0 flags 0x1 on motherboard
npx0: INT 16 interface
fdc0: direction bit not set
fdc0: cmd 3 failed at out byte 1 of 3
fdc0 not found at 0x3f0
wdc0 at 0x1f0-0x1f7 irq 14 on isa
wdc0: unit 0 (wd0): <SunDisk SQFXB-80>, single-sector-i/o
wd0: 76MB (156672 sectors), 612 cyls, 8 heads, 32 S/T, 512 B/S
wdc0: unit 1 (wd1): <IBM-DCXA-210000>
wd1: 8063MB (16514064 sectors), 16383 cyls, 16 heads, 63 S/T, 512 B/S
wdc1 not found at 0x170
wdc2 not found at 0x180
ep0 not found at 0x300
fxp0: Ethernet address 00:a0:a5:12:05:5a
fxp1: Ethernet address 00:a0:a5:12:05:59
fxp2: Ethernet address 02:00:00:00:00:01
swapon: adding /dev/wd1s1b as swap device
Automatic reboot in progress...
/dev/rwd0s1a: clean, 16599 free (95 frags, 2063 blocks, 0.1% fragmentation)
/dev/rwd0s1e: clean, 9233 free (9 frags, 1153 blocks, 0.1% fragmentation)
/dev/rwd0s1a: clean, 16599 free (95 frags, 2063 blocks, 0.1% fragmentation)
/dev/rwd1s1f: clean, 4301055 free (335 frags, 537590 blocks, 0.0% fragmentation)

```

show system boot-messages lcc (TX Matrix Router)

```

user@host> show system boot-messages lcc 2
lcc2-re0:
-----
Copyright (c) 1996-2001, Juniper Networks, Inc.
All rights reserved.
Copyright (c) 1992-2001 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
The Regents of the University of California. All rights reserved.
JUNOS 7.0-20040912.0 #0: 2004-09-12 09:16:32 UTC

builder@device1.example.com:/build/benten-b/7.0/20040912.0/obj-i386/sys/compile/JUNIPER
Timecounter "i8254" frequency 1193182 Hz
Timecounter "TSC" frequency 601368936 Hz
CPU: Pentium III/Pentium III Xeon/Celeron (601.37-MHz 686-class CPU)
Origin = "GenuineIntel" Id = 0x68a Stepping = 10

```



```

Features=0x387f9ff<FPU,WE,DE,PSE,TSC,MSR,PAE,MCE,CX8,SEP,MTRR,PGE,MCA,CMOV,PAT,PSE36,PN,MMX,FXSR,SSE>
real memory = 2147467264 (2097136K bytes)
sio0: gdb debugging port
avail memory = 2084040704 (2035196K bytes)
Preloaded elf kernel "kernel" at 0xc06d9000.
DEVFS: ready for devices
Pentium Pro MTRR support enabled
md0: Malloc disk
DRAM Data Integrity Mode: ECC Mode with h/w scrubbing
npx0: <math processor> on motherboard
npx0: INT 16 interface
pcib0: <ServerWorks NB6635 3.0LE host to PCI bridge> on motherboard
pci0: <PCI bus> on pcib0
pcic-pci0: <TI PCI-1410 PCI-CardBus Bridge> irq 15 at device 1.0 on pci0
pcic-pci0: TI12XX PCI Config Reg: [pwr save][pci only]
fxp0: <Intel Embedded 10/100 Ethernet> port 0x1000-0x103f mem
0xfb800000-0xfb81ffff,0xfb820000-0xfb820fff irq 9 at device 3.0 on pci0
fxp1: <Intel Embedded 10/100 Ethernet> port 0x1040-0x107f mem
0xfb840000-0xfb85ffff,0xfb821000-0xfb821fff irq 11 at device 4.0 on pci0
...

```

show system boot-messages (TX Matrix Plus Router)

```

user@host> show system boot-messages
sfc0-re0:
-----
Copyright (c) 1996-2009, Juniper Networks, Inc.
All rights reserved.
Copyright (c) 1992-2006 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
    The Regents of the University of California. All rights reserved.
JUNOS 9.6B3.3 #0: 2009-06-17 19:52:08 UTC

builder@device1.example.com:/volume/build/junos/9.6/release/9.6B3.3/obj-i386/bsd/sys/compile/JUNIPER
MPTable: Timecounter "i8254" frequency 1193182 Hz quality 0 CPU: Intel(R) Xeon(R)
CPU          L5238 @ 2.66GHz (2660.01-MHz 686-class CPU)   Origin =
"GenuineIntel" Id = 0x1067a Stepping = 10   Features=0xbfebfbff
...
lcc1-re0:
-----
Copyright (c) 1996-2009, Juniper Networks, Inc.
All rights reserved.
Copyright (c) 1992-2006 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
    The Regents of the University of California. All rights reserved.
JUNOS 9.6-20090617.0 #0: 2009-06-17 04:15:14 UTC

builder@device1.example.com:/volume/build/junos/9.6/production/20090617.0/obj-i386/bsd/sys/compile/JUNIPER
Timecounter "i8254" frequency 1193182 Hz quality 0
CPU: Intel(R) Xeon(R) CPU          @ 1.86GHz (1862.01-MHz 686-class CPU)

Origin = "GenuineIntel" Id = 0x1067a Stepping = 10
Features=0xbfebfbff
...

```

show system boot-messages (QFX3500 Switch)

```

user@switch> show sytem boot-messages

```

```
getmemsize: msgbufp[size=32768] = 0x81d07fe4
```

```
System physical memory distribution:
```

```
-----
Total physical memory: 4160749568 (3968 MB)
Physical memory used: 3472883712 (3312 MB)
Physical memory allocated to kernel: 2130706432 (2032 MB)
Physical memory allocated to user BTLB: 1342177280 (1280 MB)
-----
```

```
Copyright (c) 1996-2010, Juniper Networks, Inc.
```

```
All rights reserved.
```

```
Copyright (c) 1992-2006 The FreeBSD Project.
```

```
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
```

```
The Regents of the University of California. All rights reserved.
```

```
JUNOS 11.1I #0: 2010-09-17 19:18:07 UTC
```

```
ssiano@device1.example.com:/c/ssiano/DEV_QFX_SI_BRANCH/03/20100917.399988/
```

```
obj-xlr/bsd/sys/compile/JUNIPER-DCTOR
```

```
WARNING: debug.mpsafenet forced to 0 as ipsec requires Giant
```

```
JUNOS 11.1I #0: 2010-09-17 19:18:07 UTC
```

```
ssiano@device1.example.com:/c/ssiano/DEV_QFX_SI_BRANCH/03/20100917.399988/
```

```
obj-xlr/bsd/sys/compile/JUNIPER-DCTOR
```

```
real memory = 3472883712 (3312MB)
```

```
avail memory = 1708171264 (1629MB)
```

```
cpuid: 0, btlb_cpumap:0xffffffff8
```

```
FreeBSD/SMP: Multiprocessor System Detected: 12 CPUs
```

```
ETHERNET SOCKET BRIDGE initialising
```

```
Initializing QFX platform properties ..
```

```
cpu0 on motherboard
```

```
: RMI's XLR CPU Rev. 0.3 with no FPU implemented
```

```
L1 Cache: I size 32kb(32 line), D size 32kb(32 line), eight way.
```

```
L2 Cache: Size 1024kb, eight way
```

```
pic_lbus0: <XLR Local Bus>
```

```
pic_lbus0: <XLR Local Bus> on motherboard
```

```
Enter qfx control ethernet probe addr:0xc5eeec00
```

```
gmac4: <XLR GMAC GE Ethernet> on pic_lbus0
```

```
me0: Ethernet address 00:1d:b5:f7:68:40
```

```
Enter qfx control ethernet probe addr:0xc5eeeb40
```

```
gmac5: <XLR GMAC GE Ethernet> on pic_lbus0
```

```
me1: Ethernet address 00:1d:b5:f7:68:41
```

```
Enter qfx control ethernet probe addr:0xc5eeea80
```

```
gmac6: <XLR GMAC GE Ethernet> on pic_lbus0
```

```
me1: Ethernet address 00:1d:b5:f7:68:42
```

```
sio0 on pic_lbus0
```

```
Entering sioattach
```

```
sio0: type 16550A, console
```

```
xls_setup_intr: skip irq 3, xlr regs are set up somewhere else.
```

```
gblmem0 on pic_lbus0
```

```
ehci0: <RMI XLS USB 2.0 controller> on pic_lbus0
```

```
ehci_bus_attach: allocated resource. tag=1, base=bef24000
```

```
xls_ehci_init: endian hardware swapping NOT enabled.
```

```
usb0: EHCI version 1.0
```

```
usb0 on ehci0
```

```
usb0: USB revision 2.0
```

```
uhub0: vendor 0x0000 EHCI root hub, class 9/0, rev 2.00/1.00, addr 1
```

```
uhub0: 2 ports with 2 removable, self powered
```

```
umass0: USB USBFlashDrive, rev 2.00/11.00, addr 2
```

```
pcib0: PCIe link 0 up
```

```
pcib0: PCIe link 2 up
```

```
pcib0: PCIe link 3 up
```

```
pcib0: <XLS PCI Host Controller> on pic_lbus0
```

```

pci0: <PCI bus> on pcib0
pcib1: <PCI-PCI bridge> at device 0.0 on pci0
pci1: <PCI bus> on pcib1
pci1: <network, ethernet> at device 0.0 (no driver attached)
pcib2: <PCI-PCI bridge> at device 1.0 on pci0
pcib3: <PCI-PCI bridge> at device 2.0 on pci0
pci2: <PCI bus> on pcib3
pci2: <network, ethernet> at device 0.0 (no driver attached)
pcib4: <PCI-PCI bridge> at device 3.0 on pci0
pci3: <PCI bus> on pcib4
pci3: <network, ethernet> at device 0.0 (no driver attached)
cfi device address space at 0xbc000000
cfi0: <AMD/Fujitsu - 8MB> on pic_lbus0
cfi device address space at 0xbc000000
i2c0: <I2C bus controller> on pic_lbus0
i2c1: <I2C bus controller> on pic_lbus0
qfx_fmn0 on pic_lbus0
pool offset 1503776768
xlr_lbus0: <XLR Local Bus Controller> on motherboard
qfx_bcpld_probe[124]
qfx_bcpld_probe[138]: dev_type=0x0
qfx_bcpld_probe[124]
qfx_bcpld0: QFX BCPLD probe success
qfx_bcpld0qfx_bcpld_attach[174]
qfx_bcpld_attach[207] : bus_space_tag=0x0, bus_space_handle=0xbd900000
qfx_bcpld_probe[124]
qfx_bcpld1: QFX BCPLD probe success
qfx_bcpld1qfx_bcpld_attach[174]
tor_bcpld_slave_attach[1245] : bus_space_tag=0x0, bus_space_handle=0xbda00000
Initializing product: 96 ..
bmeb: bmeb_lib_init done 0xc60a5000, addr 0x809c99a0
bme0:Virtual BME driver initializing
Timecounter "mips" frequency 1200000000 Hz quality 0
Timecounter "xlr_pic_timer" frequency 66666666 Hz quality 1
Timecounters tick every 1.000 msec
Loading the NETPFE fc module
IPsec: Initialized Security Association Processing.
SMP: AP CPU #3 Launched!
SMP: AP CPU #1 Launched!
SMP: AP CPU #2 Launched!
SMP: AP CPU #4 Launched!
SMP: AP CPU #5 Launched!
SMP: AP CPU #7 Launched!
SMP: AP CPU #6 Launched!
SMP: AP CPU #11 Launched!
SMP: AP CPU #10 Launched!
SMP: AP CPU #9 Launched!
SMP: AP CPU #8 Launched!
da0 at umass-sim0 bus 0 target 0 lun 0
da0: <USB USBFlashDrive 1100> Removable Direct Access SCSI-0 device
da0: 40.000MB/s transfers
da0: 3920MB (8028160 512 byte sectors: 255H 63S/T 499C)
Trying to mount root from ufs:/dev/da0s1a

```

show system auto-snapshot

Syntax	show system auto-snapshot
Release Information	Command introduced in Junos OS Release 12.3 for EX Series switches. Command introduced in Junos OS Release 12.1X45-D10 for SRX Series devices.
Description	Display automatic snapshot status information. When the automatic snapshot feature is enabled and the system reboots from the alternate root partition, the switch automatically takes a snapshot of the root file system in the alternate root partition and copies it onto the primary root partition. This automatic snapshot procedure takes place whenever the system reboots from the alternate partition, regardless of whether the reboot from the alternate partition is due to a command or due to a corruption of the primary partition.
Options	This command has no options.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • Understanding Resilient Dual-Root Partitions on Switches on page 111
List of Sample Output	show system auto-snapshot on page 489
Output Fields	Table 50 on page 488 describes the output fields for the show system auto-snapshot command. Output fields are listed in the approximate order in which they appear.

Table 50: show system auto-snapshot status Output Fields

Field Name	Field Description
Auto-snapshot configuration	Status of the configuration: <ul style="list-style-type: none"> • Enabled—If the system reboots from the alternate partition, the automatic snapshot feature automatically takes a snapshot of the alternate partition and copies it onto the primary partition. • Disabled—The system does not automatically take a snapshot of the alternate partition. You must use the manual snapshot command, request system snapshot, to take a snapshot of one partition and copy it onto the other.
Auto-snapshot state	Status of the automatic snapshot procedure: <ul style="list-style-type: none"> • Completed—The automatic snapshot procedure has completed copying the alternate partition to the primary partition and the alarm has been cleared. • Disabled—The automatic snapshot procedure is inactive. • In progress—The automatic snapshot procedure is in progress. It takes about 10 to 15 minutes to complete, depending upon disk size.

Sample Output

show system auto-snapshot

```
user@switch> show system auto-snapshot
Auto-snapshot Configuration: Enabled
Auto-snapshot State: Disabled
```

show system download

Syntax	<code>show system download <download-id></code>
Release Information	Command introduced in Junos OS Release 11.2 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Description	Display a brief summary of all the download instances along with their current state and extent of progress. If a download-id is provided, the command displays a detailed report of the particular download instance.
Options	<ul style="list-style-type: none"> download-id—(Optional) The ID number of the download instance.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> request system download start on page 376 Understanding Download Manager for SRX Series Devices on page 51
List of Sample Output	show system download on page 490 show system download 1 on page 491
Output Fields	Table 51 on page 490 lists the output fields for the show system download command. Output fields are listed in the approximate order in which they appear.

Table 51: show system download Output Fields

Field Name	Field Description
ID	Displays the download identification number.
Status	Displays the state of a particular download.
Start Time	Displays the start time of a particular download.
Progress	Displays the percentage of a download that has been completed.
URL	Displays the URL from which the file was downloaded.

Sample Output

show system download

```

user@host> show system download
Download Status Information:
ID  Status      Start Time      Progress  URL
1   Active      May 4 06:28:36  5%        ftp://ftp-server//tftpboot/1m_file

```

2	Active	May 4 06:29:07	3%	ftp://ftp-server//tftpboot/5m_file
3	Error	May 4 06:29:22	Unknown	ftp://ftp-server//tftpboot/badfile
4	Completed	May 4 06:29:40	100%	ftp://ftp-server//tftpboot/smallfile

show system download 1

```
user@host> show system download 1
```

```
Download ID      : 1
Status           : Active
Progress         : 6%
URL              : ftp://ftp-server//tftpboot/1m_file
Local Path       : /var/tmp/1m_file
Maximum Rate     : 1k
Creation Time    : May 4 06:28:36
Scheduled Time   : May 4 06:28:36
Start Time       : May 4 06:28:37
Error Count      : 0
```

show system license

Syntax	<code>show system license</code> <code><installed key-content <i>filename</i> keys revoked-info usage></code>
Release Information	Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 9.0 for EX Series switches. Command introduced in Junos OS Release 11.1 for the QFX Series. Command introduced in Junos OS Release 13.3 for the MX104 3D Universal Edge Routers. Customer ID added to output of data center users in Junos OS Release 15.1.
Description	Display licenses and information about how they are used.
Options	none —Display all license information. key-content <i>filename</i> —(Optional) Display license key contents of the specified filename. installed —(Optional) Display installed licenses only. keys —(Optional) Display a list of license keys. Use this information to verify that each expected license key is present. revoked-info —(Optional) Display information about revoked licenses. usage —(Optional) Display the state of licensed features.
Required Privilege Level	maintenance
Related Documentation	
List of Sample Output	show system license (vMX Routers with Juniper Agile Licensing) on page 493 show system license on page 494 show system license installed on page 495 show system license keys on page 495 show system license usage on page 495 show system license (MX104 Routers) on page 495 show system license installed (MX104 Routers) on page 496 show system license keys (MX104 Routers) on page 496 show system license usage (MX104 Routers) on page 496 show system license (MX104 Routers) on page 496 show system license installed (MX104 Routers) on page 497 show system license keys (MX104 Routers) on page 497 show system license usage (MX104 Routers) on page 497 show system license (MX104 Routers) on page 498 show system license installed (MX104 Routers) on page 498 show system license keys (MX104 Routers) on page 498

[show system license usage \(MX104 Routers\) on page 499](#)

[show system license \(QFX Series\) on page 499](#)

[show system license \(QFX5110 Switch with Disaggregated Feature License\) on page 499](#)

Output Fields Table 52 on page 493 lists the output fields for the **show system license** command. Output fields are listed in the approximate order in which they appear.

Table 52: show system license Output Fields

Field Name	Field Description
Feature name	Name assigned to the configured feature. You use this information to verify that all the features for which you installed licenses are present.
Licenses used	<p>Number of licenses used by a router or switch. You use this information to verify that the number of licenses used matches the number configured. If a licensed feature is configured, the feature is considered used.</p> <p>NOTE: In Junos OS Release 10.1 and later, the Licenses used column displays the actual usage count based on the number of active sessions or connections as reported by the corresponding feature daemons. This is applicable for scalable license-based features such as Subscriber Access (scale-subscriber), L2TP (scale-l2tp), Mobile IP (scale-mobile-ip), and so on.</p>
Licenses installed	<p>Information about the installed license key:</p> <ul style="list-style-type: none"> • License identifier—Identifier associated with a license key. • State—State of the license key: valid or invalid. An invalid state indicates that the key was entered incorrectly or is not valid for the specific device. • License version—Version of a license. The version indicates how the license is validated, the type of signature, and the signer of the license key. • Customer ID—Name of the customer license is for. Feature added as of Junos OS Release 15.1 for data center customers (for example QFX Series platform users). • Valid for device—Device that can use a license key. • Group defined—Group membership of a device. • Features—Feature associated with a license, such as data link switching (DLSw).
Licenses needed	Number of licenses required for features being used but not yet properly licensed.
Expiry	Amount of time left within the grace period before a license is required for a feature being used.

Sample Output

show system license (vMX Routers with Juniper Agile Licensing)

```

user@host> show system license
License usage:

```

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
VMX-SCALE	0	1	0	permanent
VMX-BANDWIDTH	0	130000	0	permanent
mobile-next-DPI-base	0	1000	0	permanent
mobile-next-policy-prepaid-scaling	0	1000	0	permanent
mobile-next-http-app-scaling	0	1000	0	permanent
mobile-next-scaling	0	1000	0	permanent

logical-system	0	1	0	permanent
ax411-wlan-ap	0	2	0	permanent
dynamic-vpn	0	2	0	permanent
scale-mobile-ip	0	1000	0	permanent
scale-l2tp	0	1000	0	permanent
scale-subscriber	0	64010	0	permanent

Licenses installed:

License identifier: RMS818090001

License version: 1

Software Serial Number: AID000000001

Customer ID: LABJuniperTest

License count: 1

Features:

VMX-SCALE - Max scale supported by the VMX

date-based, 2017-03-15 05:30:00 IST - 2017-05-14 05:30:00 IST

License identifier: RMS818020001

License version: 1

Software Serial Number: AID000000001

Customer ID: vMX-JuniperNetworks

License count: 1

Features:

VMX-SCALE - Max scale supported by the VMX

permanent

...

show system license

user@host> show system license

License usage:

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
subscriber-accounting	2	2	0	permanent
subscriber-authentication	1	2	0	permanent
subscriber-address-assignment	2	2	0	permanent
subscriber-vlan	2	2	0	permanent
subscriber-ip	0	2	0	permanent
scale-subscriber	2	3	0	permanent
scale-l2tp	4	5	0	permanent
scale-mobile-ip	1	2	0	permanent

Licenses installed:

License identifier: XXXXXXXXXX

License version: 2

Customer ID: ACME CORPORATION

Features:

subscriber-accounting - Per Subscriber Radius Accounting

permanent

subscriber-authentication - Per Subscriber Radius Authentication

permanent

subscriber-address-assignment - Radius/SRC Address Pool Assignment

permanent

subscriber-vlan - Dynamic Auto-sensed Vlan

permanent

subscriber-ip - Dynamic and Static IP

permanent

show system license installed

```

user@host> show system license installed
License identifier: XXXXXXXXXX
License version: 2
Features:
  subscriber-accounting - Per Subscriber Radius Accounting
                        permanent
  subscriber-authentication - Per Subscriber Radius Authentication
                        permanent
  subscriber-address-assignment - Radius/SRC Address Pool Assignment
                        permanent
  subscriber-vlan - Dynamic Auto-sensed Vlan
                        permanent
  subscriber-ip - Dynamic and Static IP
                        permanent

```

show system license keys

```

user@host> show system license keys
XXXXXXXXXX xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
          xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
          xxxxxx xxxxxx xxx

```

show system license usage

```

user@host> show system license usage
License usage:

```

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
subscriber-accounting	2	2	0	permanent
subscriber-authentication	1	2	0	permanent
subscriber-address-assignment	2	2	0	permanent
subscriber-vlan	2	2	0	permanent
subscriber-ip	0	2	0	permanent
scale-subscriber	2	3	0	permanent
scale-l2tp	4	5	0	permanent
scale-mobile-ip	1	2	0	permanent

show system license (MX104 Routers)

In the following output, ports 0 and 1 are activated by installing the license to activate the first two built-in ports.

```

user@host> show system license
License usage:

```

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
scale-subscriber	0	1000	0	permanent
scale-l2tp	0	1000	0	permanent
scale-mobile-ip	0	1000	0	permanent
MX104-2x10Gig-port-0-1	0	1	0	permanent

```

Licenses installed:
License identifier: XXXXXXXXXX
License version: 2
Features:
  MX104-2x10Gig-port-0-1 - MX104 2X10Gig Builtin Port(xe-2/0/0 & xe-2/0/1)

```

```
upgrade
permanent
```

show system license installed (MX104 Routers)

In the following output, ports 0 and 1 are activated by installing the license to activate the first two built-in ports.

```
user@host > show system license installed
License identifier: XXXXXXXXXX
License version: 2
Features:
MX104-2x10Gig-port-0-1 - MX104 2X10Gig Builtin Port(xe-2/0/0 & xe-2/0/1)
upgrade
permanent
```

show system license keys (MX104 Routers)

In the following output, ports 0 and 1 are activated by installing the license to activate the first two built-in ports.

```
user@host > show system license keys

XXXXXXXXXX xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
          xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
          xxxxxx xxxx
```

show system license usage (MX104 Routers)

In the following output, ports 0 and 1 are activated by installing the license to activate the first two built-in ports.

```
user@host > show system license usage
```

Feature name	Licenses used	Licenses installed	Expiry	needed	
scale-subscriber	0	1000		0	permanent
scale-l2tp	0	1000		0	permanent
scale-mobile-ip	0	1000		0	permanent
MX104-2x10Gig-port-0-1	0	1		0	permanent

show system license (MX104 Routers)

In the following output, ports 2 and 3 are activated by installing the license to activate the next two built-in ports after installing the license to activate the first two built-in ports.

```
user@host > show system license
License usage:
```

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
scale-subscriber	0	1000	0	permanent
scale-l2tp	0	1000	0	permanent
scale-mobile-ip	0	1000	0	permanent
MX104-2x10Gig-port-0-1	0	1	0	permanent
MX104-2x10Gig-port-2-3	0	1	0	permanent

```
Licenses installed:
License identifier: XXXXXXXXXX
```

```

License version: 2
Features:
  MX104-2x10Gig-port-0-1 - MX104 2X10Gig Builtin Port(xe-2/0/0 & xe-2/0/1)
upgrade
  permanent

License identifier: XXXXXXXXXX
License version: 2
Features:
  MX104-2x10Gig-port-2-3 - MX104 2X10Gig Builtin Port(xe-2/0/2 & xe-2/0/3)
upgrade
  permanent

```

show system license installed (MX104 Routers)

In the following output, ports 2 and 3 are activated by installing the license to activate the next two built-in ports after installing the license to activate the first two built-in ports.

```

user@host > show system license installed
License identifier: XXXXXXXXXX
License version: 2
Features:
  MX104-2x10Gig-port-0-1 - MX104 2X10Gig Builtin Port(xe-2/0/0 & xe-2/0/1)
upgrade
  permanent

License identifier: XXXXXXXXXX
License version: 2
Features:
  MX104-2x10Gig-port-2-3 - MX104 2X10Gig Builtin Port(xe-2/0/2 & xe-2/0/3)
upgrade
  permanent

```

show system license keys (MX104 Routers)

In the following output, ports 2 and 3 are activated by installing the license to activate the next two built-in ports after installing the license to activate the first two built-in ports.

```

user@host > show system license keys

XXXXXXXXXX xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
          xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
          xxxxxx xxxx

XXXXXXXXXX xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
          xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
          xxxxxx xxxx

```

show system license usage (MX104 Routers)

In the following output, ports 2 and 3 are activated by installing the license to activate the next two built-in ports after installing the license to activate the first two built-in ports.

```

user@host > show system license usage

```

Licenses	Licenses	Licenses	Expiry	
Feature name	used	installed	needed	
scale-subscriber	0	1000	0	permanent
scale-l2tp	0	1000	0	permanent
scale-mobile-ip	0	1000	0	permanent
MX104-2x10Gig-port-0-1	0	1	0	permanent
MX104-2x10Gig-port-2-3	0	1	0	permanent

show system license (MX104 Routers)

In the following output, ports 0,1,2, and 3 are activated by installing a single license key to activate all four built-in ports.

```
user@host > show system license
```

License usage:

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
scale-subscriber	0	1000	0	permanent
scale-l2tp	0	1000	0	permanent
scale-mobile-ip	0	1000	0	permanent
MX104-2x10Gig-port-0-1	0	1	0	permanent
MX104-2x10Gig-port-2-3	0	1	0	permanent

Licenses installed:

License identifier: XXXXXXXXXX

License version: 2

Features:

```
MX104-2x10Gig-port-0-1 - MX104 2X10Gig Builtin Port(xe-2/0/0 & xe-2/0/1)
upgrade
permanent
MX104-2x10Gig-port-2-3 - MX104 2X10Gig Builtin Port(xe-2/0/2 & xe-2/0/3)
upgrade
permanent
```

show system license installed (MX104 Routers)

In the following output, ports 0,1,2, and 3 are activated by installing a single license key to activate all four built-in ports.

```
user@host > show system license installed
```

License identifier: XXXXXXXXXX

License version: 2

Features:

```
MX104-2x10Gig-port-0-1 - MX104 2X10Gig Builtin Port(xe-2/0/0 & xe-2/0/1)
upgrade
permanent
MX104-2x10Gig-port-2-3 - MX104 2X10Gig Builtin Port(xe-2/0/2 & xe-2/0/3)
upgrade
permanent
```

show system license keys (MX104 Routers)

In the following output, ports 0,1,2, and 3 are activated by installing a single license key to activate all four built-in ports.

```
user@host > show system license keys
```

```
XXXXXXXXX  xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
```

```

xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
xxxxxx xxxxxx x

```

show system license usage (MX104 Routers)

In the following output, ports 0,1,2, and 3 are activated by installing a single license key to activate all four built-in ports.

```

user@host > show system license usage

```

Feature name	Licenses used	Licenses installed	Expiry needed	
scale-subscriber	0	1000	0	permanent
scale-l2tp	0	1000	0	permanent
scale-mobile-ip	0	1000	0	permanent
MX104-2x10Gig-port-0-1	0	1	0	permanent
MX104-2x10Gig-port-2-3	0	1	0	permanent

show system license (QFX Series)

```

user@switch> show system license
License usage:

```

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
qfx-edge-fab	1	1	1	permanent

```

Licenses installed:
License identifier: JUNOS417988
License version: 1
Features:
  qfx-edge-fab - QFX3000 Series QF/Node feature license
                permanent

```

show system license (QFX5110 Switch with Disaggregated Feature License)

```

user@switch> show system license
License usage:

```

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
bgp	0	1	0	2017-07-05 00:00:00 UTC
isis	0	1	0	2017-07-05 00:00:00 UTC
vxlan	0	1	0	2017-07-05 00:00:00 UTC
ovsdb	0	1	0	2017-07-05 00:00:00 UTC
jbs1	0	1	0	2017-07-02 00:00:00 UTC
upgrade1	0	1	0	2017-07-05 00:00:00 UTC

```

Licenses installed:
License identifier: JUNOS797095
License version: 4
Software Serial Number: 91730A00223925
Customer ID: Juniper
Features:
  JUNOS-BASE-SERVICES-CLASS-1 - QFX Junos Base Services license for Class 1 HW
                                date-based, 2016-07-01 00:00:00 UTC - 2017-07-02 00:00:00 UTC

```

```
License identifier: JUNOS797646
License version: 4
Software Serial Number: 91730A00224207
Customer ID: Juniper
Features:
  CLASS-1-JUNOS-BASE-ADVANCED-UPGRADE - Class 1 Junos Base to Advanced Services
Upgrade
  date-based, 2016-07-04 00:00:00 UTC - 2017-07-05 00:00:00 UTC

{master:0}
```


show system license (View)

Syntax	show system license <installed keys status usage>
Release Information	Command introduced in Junos OS Release 9.5. Logical system status option added in Junos OS Release 11.2.
Description	Display licenses and information about how licenses are used.
Options	<p>none—Display all license information.</p> <p>installed—(Optional) Display installed licenses only.</p> <p>keys—(Optional) Display a list of license keys. Use this information to verify that each expected license key is present.</p> <p>status—(Optional) Display license status for a specified logical system or for all logical systems.</p> <p>usage—(Optional) Display the state of licensed features.</p>
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • Adding New Licenses (CLI Procedure) on page 317
List of Sample Output	<p>show system license on page 502</p> <p>show system license installed on page 502</p> <p>show system license keys on page 503</p> <p>show system license usage on page 503</p> <p>show system license status logical-system all on page 503</p>
Output Fields	Table 53 on page 501 lists the output fields for the show system license command. Output fields are listed in the approximate order in which they appear.

Table 53: show system license Output Fields

Field Name	Field Description
Feature name	Name assigned to the configured feature. You use this information to verify that all the features for which you installed licenses are present.
Licenses used	Number of licenses used by the device. You use this information to verify that the number of licenses used matches the number configured. If a licensed feature is configured, the feature is considered used.

Table 53: show system license Output Fields (*continued*)

Field Name	Field Description
Licenses installed	Information about the installed license key: <ul style="list-style-type: none"> • License identifier—Identifier associated with a license key. • License version—Version of a license. The version indicates how the license is validated, the type of signature, and the signer of the license key. • Valid for device—Device that can use a license key. • Features—Feature associated with a license.
Licenses needed	Number of licenses required for features being used but not yet properly licensed.
Expiry	Time remaining in the grace period before a license is required for a feature being used.
Logical system license status	Displays whether a license is enabled for a logical system.

Sample Output

show system license

```
user@host> show system license
```

```
License usage:
```

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
av_key_kaspersky_engine 01:00:00 IST	1	1	0	2012-03-30
wf_key_surfcontrol_cpa 01:00:00 IST	0	1	0	2012-03-30
dynamic-vpn	0	1	0	permanent
ax411-wlan-ap	0	2	0	permanent

```
Licenses installed:
```

```
License identifier: JUNOS301998
```

```
License version: 2
```

```
Valid for device: AG4909AA0080
```

```
Features:
```

```
av_key_kaspersky_engine - Kaspersky AV  
date-based, 2011-03-30 01:00:00 IST - 2012-03-30 01:00:00 IST
```

```
License identifier: JUNOS302000
```

```
License version: 2
```

```
Valid for device: AG4909AA0080
```

```
Features:
```

```
wf_key_surfcontrol_cpa - Web Filtering  
date-based, 2011-03-30 01:00:00 IST - 2012-03-30 01:00:00 IST
```

show system license installed

```
user@host> show system license installed
```

```
License identifier: JUNOS301998
```

```
License version: 2
```

```
Valid for device: AG4909AA0080
```

```

Features:
  av_key_kaspersky_engine - Kaspersky AV
    date-based, 2011-03-30 01:00:00 IST - 2012-03-30 01:00:00 IST

License identifier: JUNOS302000
License version: 2
Valid for device: AG4909AA0080
Features:
  wf_key_surfcontrol_cpa - Web Filtering
    date-based, 2011-03-30 01:00:00 IST - 2012-03-30 01:00:00 IST

```

show system license keys

```

user@host> show system license keys

XXXXXXXXXX xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
          xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
          xxxxxx xxxxxx xxx

```

show system license usage

```

user@host> show system license usage


```

Feature name	Licenses used	Licenses installed	Licenses needed	Expiry
av_key_kaspersky_engine 01:00:00 IST	1	1	0	2012-03-30
wf_key_surfcontrol_cpa 01:00:00 IST	0	1	0	2012-03-30
dynamic-vpn	0	1	0	permanent
ax411-wlan-ap	0	2	0	permanent

show system license status logical-system all

```

user@host> show system license status logical-system all
Logical system license status:


```

logical system name	license status
root-logical-system	enabled
LSYS0	enabled
LSYS1	enabled
LSYS2	enabled

show system login logout

Syntax `show system login logout`

Release Information Command introduced in Junos OS Release 11.2.

Description Display the usernames locked after unsuccessful login attempts.

Required Privilege Level view and system

Related Documentation

- *lockout-period*
- [clear system login logout on page 369](#)

List of Sample Output [show system login logout on page 504](#)

Output Fields [Table 54 on page 504](#) lists the output fields for the **show system login logout** command. Output fields are listed in the approximate order in which they appear.

Table 54: show system login logout

Field Name	Field Description	Level of Output
User	Username	All levels
Lockout start	Date and time the username was locked	All levels
Lockout end	Date and time the username was unlocked	All levels

Sample Output

show system login logout

```
user@host> show system login logout
```

```
User          Lockout start      Lockout end
root          2011-05-11 09:11:15 UTC 2011-05-11 09:13:15 UTC
```

show system snapshot

List of Syntax [Syntax on page 505](#)
 [Syntax \(EX Series Switches\) on page 505](#)

Syntax show system snapshot

Syntax (EX Series Switches) show system snapshot
 <all-members | local | member *member-id*>
 <media (external | internal)>

Release Information Command introduced in Junos OS Release 7.6.
 Command introduced in Junos OS Release 10.0 for EX Series switches.
 Option **slice** deprecated for Junos OS with Upgraded FreeBSD in Junos OS Release 15.1.



NOTE: To determine which platforms run Junos OS with Upgraded FreeBSD, see the table listing the platforms currently running Junos OS with upgraded FreeBSD in “[Understanding Junos OS with Upgraded FreeBSD](#)” on page 17.

Description Display information about the backup software:

- On the routers, display information about the backup software, which is located in the **/altroot**, and **/altconfig** file systems or on the alternate media.
- On the switches, display information about the backup of the root file system (**/**) and directories **/altroot**, **/config**, **/var**, and **/var/tmp**, which are located either on an external USB flash drive or in internal flash memory.



NOTE: To back up software, use the **request system snapshot** command.

Options **none**—Display information about the backup software.

all-members | local | member *member-id*—(EX Series switch Virtual Chassis only)
 (Optional) Display the snapshot in a Virtual Chassis:

- **all-members**—Display the snapshot for all members of the Virtual Chassis.
- **local**—Display the snapshot on the member of the Virtual Chassis that you are currently logged into.
- **member *member-id***—Display the snapshot for the specified member of the Virtual Chassis.

media (external | internal)—(EX Series switch only) (Optional) Display the destination media location for the snapshot. The **external** option specifies the snapshot on an external mass storage device, such as a USB flash drive. The **internal** option specifies the snapshot on an internal memory source, such as internal flash memory. If no additional options are specified, the command displays the snapshot stored in both slices.

Required Privilege Level view

Related Documentation

- [request system snapshot on page 406](#)

List of Sample Output

- [show system snapshot \(Router\) on page 506](#)
- [show system snapshot media external \(Switch\) on page 506](#)
- [show system snapshot media internal \(Switch\) on page 507](#)

Output Fields Table 55 on page 506 lists the output fields for the **show system snapshot** command. Output fields are listed in the approximate order in which they appear.

Table 55: show system snapshot Output Fields

Field Name	Field Description
Creation date	Date and time of the last snapshot.
JUNOS version on snapshot	Junos OS release number of individual software packages.

Sample Output

show system snapshot (Router)

```
user@host> show system snapshot
Information for snapshot on hard-disk
Creation date: Oct 5 13:53:29 2005
JUNOS version on snapshot:
  jbase   : 7.3R2.5
  jcrypto: 7.3R2.5
  jdocs   : 7.3R2.5
  jkernel: 7.3R2.5
  jpfe    : M40-7.3R2.5
  jroute  : 7.3R2.5
```

show system snapshot media external (Switch)

```
user@switch> show system snapshot media external
Information for snapshot on      external (/dev/dals1a) (backup)
Creation date: Mar 19 03:37:18 2012
JUNOS version on snapshot:
  jbase   : ex-12.1I20120111_0048_user
  jcrypto-ex: 12.1I20120111_0048_user
  jdocs-ex: 12.1I20120111_0048_user
```

```

jroute-ex: 12.1I20120111_0048_user
jswitch-ex: 12.1I20120111_0048_user
jweb-ex: 12.1I20120111_0048_user
Information for snapshot on      external (/dev/da1s2a) (primary)
Creation date: Mar 19 03:38:25 2012
JUNOS version on snapshot:
  jbase   : ex-12.2I20120305_2240_user
  jcrypto-ex: 12.2I20120305_2240_user
  jdocs-ex: 12.2I20120305_2240_user
  jroute-ex: 12.2I20120305_2240_user
  jswitch-ex: 12.2I20120305_2240_user
  jweb-ex: 12.2I20120305_2240_user

```

show system snapshot media internal (Switch)

```

user@switch> show system snapshot media internal
Information for snapshot on internal (/dev/da0s1a) (backup)
Creation date: Mar 14 05:01:02 2011
JUNOS version on snapshot:
  jbase   : 11.1R1.9
  jcrypto-ex: 11.1R1.9
  jdocs-ex: 11.1R1.9
  jkernel-ex: 11.1R1.9
  jroute-ex: 11.1R1.9
  jswitch-ex: 11.1R1.9
  jweb-ex: 11.1R1.9
  jpfe-ex42x: 11.1R1.9
Information for snapshot on internal (/dev/da0s2a) (primary)
Creation date: Mar 30 08:46:27 2011
JUNOS version on snapshot:
  jbase   : 11.2-20110330.0
  jcrypto-ex: 11.2-20110330.0
  jdocs-ex: 11.2-20110330.0
  jkernel-ex: 11.2-20110330.0
  jroute-ex: 11.2-20110330.0
  jswitch-ex: 11.2-20110330.0
  jweb-ex: 11.2-20110330.0
  jpfe-ex42x: 11.2-20110330.0

```

show system snapshot (Junos OS with Upgraded FreeBSD)

Syntax (MX240, MX480, MX960, MX2010, MX2020 Routers and EX9200 Switches only)

show system snapshot

Release Information

Command introduced in Junos OS Release 15.1 for MX240, MX480, MX960, MX2010, and MX2020 routers and EX9200 switches only.
Command introduced in Junos OS Release 16.1R1 for VM host on MX240, MX480, MX960, MX2010, and MX2020 routers and PTX5000 routers.
Output for recovery snapshots provided in Junos Release 17.2 for all platforms using Junos OS with upgraded FreeBSD.

Description

Display information about the non-recovery backup software, which is located in the **junos** file system on the hard disk drive or solid-state drive (SSD).

Display information about recovery snapshot after the non-recovery information.



NOTE: To back up software, use the **request system snapshot** command.

Required Privilege Level

view

Related Documentation

- [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 414](#)
- [request system reboot \(Junos OS with Upgraded FreeBSD\) on page 398](#)
- [Understanding Junos OS with Upgraded FreeBSD on page 17](#)

List of Sample Output

[show system snapshot on page 508](#)
[show system snapshot \(updated as of Junos OS Release 17.2\) on page 509](#)

Output Fields

When you enter this command, you are provided feedback on the status of your request. If there are no snapshots available, the command returns null output.

Sample Output

show system snapshot

```
user@host> show system snapshot
Snapshot snap.20141219.122106:
Location: /packages/sets/snap.20141219.122106
Creation date: Dec 19 12:21:06 2014
Junos version: 15.1-20141216_ib_15_1_psd.0
```


Sample Output

show system snapshot (updated as of Junos OS Release 17.2)

```
user@host> show system snapshot
Non-recovery snapshots:

Snapshot snap.20170112.105151:
Location: /packages/sets/snap.20170112.105151
Creation date: Jan 12 10:51:51 2017
Junos version: 17.2I20170112_0239

Snapshot snap.20170112.112307:
Location: /packages/sets/snap.20170112.112307
Creation date: Jan 12 11:23:07 2017
Junos version: 17.2I20170112_0239


Snapshot snap.20170112.112314:
Location: /packages/sets/snap.20170112.112314
Creation date: Jan 12 11:23:14 2017
Junos version: 17.2I20170112_0239

Total non-recovery snapshots:      3

Recovery Snapshots:
Snapshots available on the OAM volume:
recovery.ufs
Date created: Wed Jan 11 15:59:35 PST 2017
Junos version: 17.2I20170111_2242

Total recovery snapshots: 1
```

show system snapshot (Upgraded FreeBSD)

Syntax	show system snapshot
Release Information	Command introduced in Junos OS Release 17.3 for SRX5400, SRX5600, and SRX5800 devices.
Description	<p>Display information about the non-recovery backup software, which is located in the junos file system on the hard disk drive or solid-state drive (SSD).</p> <p>Display information about recovery snapshot after the non-recovery information.</p>
	<div> NOTE: To back up software, use the request system snapshot command.</div>
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none">• request system snapshot (Junos OS with Upgraded FreeBSD) on page 414• request system reboot (Junos OS with Upgraded FreeBSD) on page 398• Understanding Junos OS with Upgraded FreeBSD on page 17
List of Sample Output	show system snapshot on page 510
Output Fields	When you enter this command, you are provided feedback on the status of your request. If there are no snapshots available, the command returns null output.

Sample Output

show system snapshot

```
user@host> show system snapshot
Non-recovery snapshots:
Snapshot snap.20170802.090921:
Location: /packages/sets/snap.20170802.090921
Creation date: Aug  2 09:09:21 2017
Junos version: 17.3R1

Snapshot snap.20170802.090951:
Location: /packages/sets/snap.20170802.090951
Creation date: Aug  2 09:09:51 2017
Junos version: 17.3R1

Total non-recovery snapshots: 2

Recovery Snapshots:
Snapshots available on the OAM volume:
recovery.ufs
```

Date created: Wed Aug 2 09:16:12 UTC 2017
Junos version: 17.3R1

Total recovery snapshots: 1

show system snapshot media

Syntax	<code>show system snapshot media <i>media-type</i></code>
Release Information	Command introduced in Junos OS Release 10.2 .
Description	Display the snapshot information for both root partitions on SRX Series devices
Options	<ul style="list-style-type: none">• <code>internal</code>— Show snapshot information from internal media.• <code>usb</code>— Show snapshot information from device connected to USB port.• <code>external</code>— Show snapshot information from the external CompactFlash card.
Required Privilege Level	View
Related Documentation	<ul style="list-style-type: none">• Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 201
List of Sample Output	show system snapshot media internal on page 512 show system snapshot media usb on page 512

Sample Output

show system snapshot media internal

```
show system snapshot media internal
Information for snapshot on      internal (/dev/da0s1a) (primary)
Creation date: Jan 15 10:43:26 2010
JUNOS version on snapshot:
  junos : 10.1B3-domestic
Information for snapshot on      internal (/dev/da0s2a) (backup)
Creation date: Jan 15 10:15:32 2010
JUNOS version on snapshot:
  junos : 10.2-20100112.0-domestic
```

show system snapshot media usb

```
show system snapshot media usb
Information for snapshot on      usb (/dev/dals1a) (primary)
Creation date: Jul 24 16:16:01 2009
JUNOS version on snapshot:
  junos : 10.0I20090723_1017-domestic
Information for snapshot on      usb (/dev/dals2a) (backup)
Creation date: Jul 24 16:17:13 2009
JUNOS version on snapshot:
  junos : 10.0I20090724_0719-domestic
```

show system storage partitions (EX Series Switches Only)

Syntax	show system storage partitions <all-members> <local> <member <i>member-id</i> >
Release Information	Command introduced in Junos OS Release 11.1 for EX Series switches.
Description	Display information about the disk partitions on EX Series switches.
Options	<p>none—Display partition information.</p> <p>all-members—(Virtual Chassis systems only) (Optional) Display partition information for all members of the Virtual Chassis.</p> <p>local—(Virtual Chassis systems only) (Optional) Display partition information for the local Virtual Chassis member.</p> <p>member <i>member-id</i>—(Virtual Chassis systems only) (Optional) Display partition information for the specified member of the Virtual Chassis configuration.</p>
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • Verifying Junos OS and Boot Loader Software Versions on an EX Series Switch on page 146
List of Sample Output	show system storage partitions on page 514
Output Fields	Table 56 on page 513 describes the output fields for the show system storage partitions command. Output fields are listed in the approximate order in which they appear.

Table 56: show system storage partitions Output Fields

Field Name	Field Description
Boot Media	Media (internal or external) from which the switch was booted.
Active Partition	Name of the active root partition.
Backup Partition	Name of the backup (alternate) root partition.
Currently booted from	Partition from which the switch was last booted.

Table 56: show system storage partitions Output Fields (*continued*)

Field Name	Field Description
Partitions information	<p>Information about partitions on the boot media:</p> <ul style="list-style-type: none"> • Partition—Partition identifier. • Size—Size of partition. • Mountpoint—Directory on which the partition is mounted.

Sample Output

show system storage partitions

```

user@switch> show system storage partitions
fpc0:
-----
Boot Media: internal (da0)
Active Partition: da0s1a
Backup Partition: da0s2a
Currently booted from: active (da0s1a)

Partitions information:
  Partition  Size  Mountpoint
  s1a        184M  /
  s2a        184M  altroot
  s3d        369M  /var/tmp
  s3e        123M  /var
  s4d         62M  /config
  s4e                unused (backup config)

```

show system storage partitions (View SRX Series)

Syntax	show system storage partitions
Release Information	Command introduced in Junos OS Release 10.2 .
Description	Display the partitioning scheme details on SRX300, SRX320, SRX340, SRX345, and SRX550HM devices.
Required Privilege Level	View
Related Documentation	<ul style="list-style-type: none"> • Example: Installing Junos OS on SRX Series Devices Using the Partition Option on page 120
List of Sample Output	show system storage partitions (single root partitioning) on page 515 show system storage partitions (USB) on page 515

show system storage partitions (dual root partitioning)

```
show system storage partitions
Boot Media: internal (da0)
Active Partition: da0s2a
Backup Partition: da0s1a
Currently booted from: active (da0s2a)
```

```
Partitions Information:
Partition  Size  Mountpoint
s1a       293M  altroot
s2a       293M  /
s3e       24M   /config
s3f       342M  /var
s4a       30M   recovery
```

show system storage partitions (single root partitioning)

```
show system storage partitions
Boot Media: internal (da0)
Partitions Information:
Partition  Size  Mountpoint
s1a       898M  /
s1e       24M   /config
s1f       61M   /var
```

show system storage partitions (USB)

```
show system storage partitions
Boot Media: usb (da1)
Active Partition: da1s1a
Backup Partition: da1s2a
Currently booted from: active (da1s1a)
```

```
Partitions Information:
```

Partition	Size	Mountpoint
s1a	293M	/
s2a	293M	altroot
s3e	24M	/config
s3f	342M	/var
s4a	30M	recovery

show version

Syntax	show version <brief detail> <node <i>node-id</i> local primary>
Release Information	Command introduced in Junos OS Release 10.4.
Description	Display the hostname and version information about the software running on the device.
Options	<p>none—Display standard information about the hostname and version of the software running on the device.</p> <p>brief—Display brief output.</p> <p>detail—Display detailed output.</p> <p>node <i>node-id</i>—Display the software version on a specific node. Range: 0 through 1</p> <p>local—Display the software version on the local node.</p> <p>primary—Display the software version on the primary node.</p>
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • Determining the Junos OS Version on page 48
List of Sample Output	show version on page 517

Sample Output

show version

```

user@host> show version
node0:
-----
Hostname: srx01
Model: srx1400
JUNOS Software Release [12.3I20141112_x_srx_12q3_x48_intgr.0-681573]
JUNOS wmi Daemon [12.1I20140304_0803_tjzhang]

```


CHAPTER 20

VM Host Software Administrative Commands

- request vmhost cleanup
- request vmhost file-copy
- request vmhost halt
- request vmhost hard-disk-test
- request vmhost power-off
- request vmhost power-on
- request vmhost reboot
- request vmhost snapshot
- request vmhost software abort
- request vmhost software add
- request vmhost software in-service-upgrade
- request vmhost software rollback
- request vmhost zeroize

request vmhost cleanup

Syntax request vmhost cleanup
<invoke-on>
<re0 | re1>
<routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Clean up temporary files, crash generated files, and log files located in the **/var/tmp**, **/var/crash**, and **/var/log** directories **respectively** on the host OS.

Options none—Clean up temporary files, crash generated files, and log files located in the **/var/tmp**, **/var/crash**, and **/var/log** directories on the host OS running on the Routing Engine on the local Virtual Chassis member.

invoke-on—(Optional) Clean up temporary files, crash generated files, and log files on all the Routing Engines or the other Routing Engine.

Clean up files in **/var/tmp**, **/var/crash**, and **/var/log** on the host OS running on a router that has dual Routing Engines. You can use the **all-routing-engine** option to clean up the files in these directories on the host OS running on all the Routing Engines or the **other-routing-engine** option to clean up the files in these directories on the host OS running on the other Routing Engine. If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, clean up files in **/var/tmp**, **/var/crash**, and **/var/log** on the host OS running on the Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine for which the files in **/var/tmp**, **/var/crash**, and **/var/log** on the host OS are to be cleaned up. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.

- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level maintenance

Related Documentation • [request system storage cleanup on page 452](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

request vmhost file-copy

Syntax `request vmhost file-copy (crash|log) j-node host file-name v-junos host file-name`

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Copy crash files or log files from the host OS to Junos OS. You can use these files for analysis and debugging purposes.

Options **crash**—Files in `/var/crash` on the host.
from j-node *filename*—Name of the host file to be copied.
log—Files in `/var/log` on the host.
to v-junos *filename*—Name of the Junos OS file to which the host file is copied.

Additional Information You can use the **show vmhost crash** and **show vmhost logs** commands to list or identify the files in the host OS to be copied to Junos OS.

Required Privilege Level maintenance

List of Sample Output [request vmhost file-copy on page 522](#)

Sample Output

request vmhost file-copy

```
user@host> request vmhost file-copy log from-jnode debug-20160214 to-vjunos /var/tmp/
/var/home/<user> # cd /var/tmp
:/var/tmp # ls -lrt debug-20160214
....
root@host:/var/tmp #
```

request vmhost halt

Syntax request vmhost halt
<re0 | re1>
<routing-engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Stop the host OS and Junos OS running on the router.

Options none—Stop the host OS and Junos OS on the router.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, stop the host OS and Junos OS running on the Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine on which the host OS and Junos OS needs to be stopped. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level maintenance

Related Documentation

- [request system halt](#)
- [vmhost on page 361](#)

List of Sample Output [request vmhost halt on page 524](#)

Sample Output

request vmhost halt

```
user@host> request vmhost halt
Halt the system ? [yes,no] (no) yes

Initiating vmhost halt... ok
Initiating Junos shutdown... shutdown: [pid 9756]
Shutdown NOW!
ok
Junos shutdown is in progress...
*** FINAL System shutdown message ***

System going down IMMEDIATELY

...
...
Operating system halted.
Please press any key to reboot.
```


request vmhost hard-disk-test

Syntax request vmhost hard-disk-test {disk *disk-name*|long|short|show-status}

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R2 for EX9200 switches.

Description Run memory and diagnostics monitoring test on the solid-state drive (SSD). The test output provides information about the various attributes of the SSD that is help monitor the status of the hard disk memory.

Options **disk *disk-name***—Name of the SSD.

long—Run extended Self-Monitoring, Analysis and Reporting Technology (SMART) tests of the SSD.

short—Run short SMART tests of the SSD.

show-status—Display the status of the test.

Required Privilege Level maintenance

Related Documentation • [vmhost on page 361](#)

List of Sample Output [request vmhost hard-disk-test on page 525](#)

Sample Output

request vmhost hard-disk-test

```
user@host> request vmhost hard-disk-test show-status disk /dev/sda
smartctl 5.42 2014-07-28 r3460
[x86_64-linux-3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt] (local build)Copyright (C)
2002-11 by Bruce Allen, http://smartmontools.sourceforge.net

=== START OF INFORMATION SECTION ===
Model Family:      StorFly Slim Sata SSD
Device Model:      StorFly VSF202CC050G-JUN
Serial Number:     P1T13003443810130041
Firmware Version:  0729-000
User Capacity:     50,020,540,416 bytes [50.0 GB]
```

```
Sector Size:      512 bytes logical/physical
Device is:        In smartctl database [for details use: -P show]
ATA Version is:   8
ATA Standard is:  ACS-2 (revision not indicated)
Local Time is:    Fri Jun 17 17:30:57 2016 IST
SMART support is: Available - device has SMART capability.
SMART support is: Enabled
```

```
=== START OF READ SMART DATA SECTION ===
```

```
SMART overall-health self-assessment test result: PASSED
```

```
General SMART Values:
```

```
Offline data collection status: (0x02) Offline data collection activity
                                     was completed without error.
                                     Auto Offline Data Collection: Disabled.
Self-test execution status:        (   0) The previous self-test routine completed
                                     without error or no self-test has ever
                                     been run.
```

```
...
...
...
```

request vmhost power-off

Syntax	<code>request vmhost power-off</code> <code><other routing-engine></code>
Release Information	Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers. Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers. Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Power off the Routing Engine on which Junos OS and the host OS are running. In a PTX3000, the Routing and Control Board is powered off.

Options none—Power off the Routing Engine immediately.

other-routing-engine—(Optional) Power off the other Routing Engine on which the Junos OS and the host OS are running. For example, if you issue the command from the master Routing Engine, the backup Routing Engine is powered off. Similarly, if you issue the command from the backup Routing Engine, the master Routing Engine is powered off.

Required Privilege Level maintenance

Related Documentation

- [request system power-off](#)
- [vmhost on page 361](#)

List of Sample Output [request vmhost power-off on page 527](#)

Sample Output

request vmhost power-off

```
user@host> request vmhost power-off
Power-off the vmhost ? [yes,no] (no) yes

Initiating vmhost shutdown... ok
Initiating Junos shutdown... shutdown: [pid 3884]
Shutdown NOW!
ok

*** FINAL System shutdown message from root@host ***
```

```
System going down IMMEDIATELY
...
...
```

request vmhost power-on

Syntax request vmhost power-on other-routing-engine

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Power on the Routing Engine on which Junos OS and the host OS are running. In a PTX3000, the Routing and Control Board is powered on.

Options **other-routing-engine**—Power on the other Routing Engine on which the Junos OS and the host OS are running. For example, if you issue the command from the master Routing Engine, the backup Routing Engine is powered on. Similarly, if you issue the command from the backup Routing Engine, the master Routing Engine is powered on.

Required Privilege Level maintenance

Related Documentation

- [request vmhost power-off on page 527](#)
- [request vmhost reboot on page 474](#)

List of Sample Output [request vmhost power-on on page 529](#)

Sample Output

request vmhost power-on

```
user@host> request vmhost power-on other-routing-engine
Routing Engine 1 power-on initiated, use "show chassis routing-engine" to verify
```

request vmhost reboot

Syntax request vmhost reboot
 <disk1>
 <disk2>
 <network>
 <re0 | re1>
 <routing engine>
 <usb>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Reboot both the Junos OS and the host OS running on the router.

Options none—Reboot the router software immediately.

disk1—(Optional) Reboot both Junos OS and the host OS on the Routing Engine and boot the Routing Engine from the primary disk.

disk2—(Optional) Reboot both Junos OS and the host OS on the Routing Engine and boot the Routing Engine from backup disk.

network—(Optional) Reboot both Junos OS and the host OS on the Routing Engine and boot the Routing Engine from network by using the PXE boot method.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, reboot both Junos OS and the host OS on the Routing Engine in slot 0 (**re0**) or on the the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine on which Junos OS and the host OS are to be rebooted. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Both Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

usb—(Optional) Reboot both Junos OS and the host OS on the Routing Engine using the USB installation media.

Required Privilege Level maintenance

Related Documentation

- [request system reboot on page 389](#)
- [request vmhost power-on on page 529](#)

List of Sample Output [request vmhost reboot on page 531](#)

Sample Output

[request vmhost reboot](#)

```
user@host> request vmhost reboot
warning: Rebooting re0
Initiating vmhost reboot... ok
Initiating Junos shutdown... shutdown: [pid 3957]
Shutdown NOW!
ok

*** FINAL System shutdown message ***

System going down IMMEDIATELY
...
...
```

request vmhost snapshot

Syntax request vmhost snapshot
 <partition>
 <re0 | re1>
 <recovery>
 <routing-engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Create a recovery snapshot of the currently running and active file system partitions on the backup disk to recover the primary disk in case of failure.

On the router, back up the snapshot of the host OS image along with the Junos OS image. In case of failure of the primary disk, you can boot from the image available in the backup disk and then recover the primary disk with the snapshot created using the **recovery** option.

Options none—Create a snapshot from the current disk to the target disk without partitioning the target disk. Contents on target disk is lost.

partition—(Optional) Create a snapshot from the current disk to target disk and partition the target disk. Contents on the target disk are lost

re0 | re1—(Optional) Create a snapshot from the current disk to target disk and partition the target disk on Routing Engine in slot 0 (**re0**) or on the Routing Engine in slot 1 (**re1**). The snapshot is taken without partitioning the target disk on corresponding Routing Engines. Contents on the target disk on the Routing Engines are lost.

recovery—(Optional) Recover the primary disk from the snapshot content stored in the backup disk. This option is applicable only when the Routing engine is booted from backup disk. Contents in the primary disk are lost.

routing-engine—(Optional) Specify the Routing Engine on which the snapshot is to be created. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Both Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.

- **master**—Master Routing Engine.
- **other**—Other Routing Engine.

Required Privilege Level maintenance

Related Documentation • [show vmhost snapshot on page 565](#)

List of Sample Output [request vmhost snapshot on page 533](#)
[request vmhost snapshot recovery on page 533](#)

Sample Output

request vmhost snapshot

```
user@host> request vmhost snapshot
warning: Existing data on the target may be lost
Proceed ? [yes,no] (no) yes

warning: Proceeding with vmhost snapshot
Current root details,           Device sda, Label: jrootb_P, Partition: sda4
Snapshot admin context from current boot disk to target disk ...
Proceeding with snapshot on secondary disk
Mounting device in preparation for snapshot...
Cleaning up target disk for snapshot ...
Creating snapshot on target disk from current boot disk ...
Snapshot created on secondary disk.
Software snapshot done
```

request vmhost snapshot recovery

```
user@host> request vmhost snapshot recovery
warning: Existing data on the target may be lost
Proceed ? [yes,no] (no) yes

warning: Proceeding with vmhost snapshot
Current root details,           Device sdb, Label: jrootb_S, Partition: sdb4
Snapshot admin context from current boot disk to target disk ...
Proceeding with snapshot on primary disk
Mounting device in preparation for snapshot...
Cleaning up target disk for snapshot ...
Creating snapshot on target disk from current boot disk ...
Primary disk is recovered now. Please issue "request vmhost reboot" to boot from
the primary disk.
Software snapshot done
```

request vmhost software abort

Syntax `request vmhost software abort in-service-upgrade`

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Abort unified in-service software upgrade (unified ISSU). The unified ISSU must be in progress and you must issue this command from a router session other than the one on which you issued the **request vmhost software in-service-upgrade** command to launch the unified ISSU.

Options **in-service-upgrade**—Abort unified ISSU.

Required Privilege Level maintenance

Related Documentation

- *request system software abort*
- [request vmhost software in-service-upgrade on page 538](#)

request vmhost software add

List of Syntax [MX Series on page 535](#)
 [PTX Series on page 535](#)

MX Series request vmhost software add *package-name*
 <no-validate>
 <re0 | re1>
 <reboot>
 <set [*package-names*]>

PTX Series request vmhost software add *package-name*
 <no-validate>
 <re0 | re1>
 <reboot>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1X65 for the PTX1000 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Install Junos OS and host software packages on the router.

For installing the host software as well as Junos OS, specify the package name **junos-vmhost-install-x.tgz** in the **request vmhost software add** command. Junos OS installation alone can be achieved by specifying the regular package name **junos-install-x.tgz** in the **request system software add** command. However, installation using the vmhost package is recommended as it upgrades both the host software and Junos OS.

Options none—Install Junos OS and host software packages on the Routing Engine on the local Virtual Chassis member.

package-name—Location from which the software package or bundle is to be installed.
 For example:

- ***/var/tmp/package-name***—For a software package or bundle that is being installed from a local directory on the router.
- ***protocol://hostname/pathname/package-name***—For a software package or bundle that is to be downloaded and installed from a remote location. Replace ***protocol*** with one of the following:

- **ftp**—File Transfer Protocol (FTP).
Use **ftp://hostname/pathname/package-name**. To specify authentication credentials, use **ftp://<username>:<password>@hostname/pathname/package-name**. To have the system prompt you for the password, specify **prompt** in place of the password. If a password is required, and you do not specify the password or **prompt**, an error message is displayed.
- **http**—Hypertext Transfer Protocol (HTTP).
Use **http://hostname/pathname/package-name**. To specify authentication credentials, use **http://<username>:<password>@hostname/pathname/package-name**. If a password is required and you omit it, you are prompted for it.
- **scp**—Secure Copy Protocol (SCP) (available only for Canada and U.S. version).
Use **scp://hostname/pathname/package-name**. To specify authentication credentials, use **scp://<username>:<password>@hostname/pathname/package-name**.



NOTE:

- The *pathname* in the protocol is the relative path to the user's home directory on the remote system and not the root directory.
- Do not use the **scp** protocol in the **request vmhost software add** command to download and install a software package or bundle from a remote location. The software upgrade is handled by the **mgd** process that does not support SCP.

To install a software package or bundle from a remote location:

1. Use the **file copy** command to copy the software package or bundle from the remote location to the **/var/tmp** directory on the hard disk:
file copy scp://source/package-name /var/tmp
2. Install the software package or bundle by using the **request vmhost software add** command:
request vmhost software add /var/tmp/package-name

no-validate—(Optional) When loading a software package or bundle with a different release, suppress the default behavior of the **validate** option.

re0 | re1—(Optional) Load a software package or bundle on the Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).



NOTE: The option **re1** is not supported on the PTX1000 Packet Transport Router.

reboot—(Optional) After adding the software package or bundle, reboot the system.

set [*package-names*]—(Optional)

Additional Information Before upgrading the software on the router, when you have a known stable system, issue the **request vmhost snapshot** command to back up the software. After you have upgraded the software on the router and are satisfied that the new package or bundle is successfully installed and running, issue the **request vmhost snapshot** command again to back up the new software to the backup disk.

After you run the **request vmhost snapshot** command, you cannot return to the previous version of the snapshot, because the previous snapshot is replaced by the new snapshot.

Required Privilege Level maintenance

Related Documentation

- [request system software add on page 420](#)
- [request vmhost software rollback on page 540](#)

List of Sample Output [request vmhost software add \(Multiple Packages\) on page 537](#)

Sample Output

[request vmhost software add \(Multiple Packages\)](#)

```
user@host> request vmhost software add set
[/var/tmp/junos-vmhost-install-ptx-x86-64-15.1F-20160518.0.tgz
/var/tmp/junos-vmhost-jdiag-15.1F-20160518.0.tgz] no-validate
Verified junos-vmhost-install-ptx-x86-64-15.1F-20160518.0 signed by
PackageDevelopmentEc_2016
Copied the config and other data to the aux disk.
Transfer junos-host-upgrade.sh
Transfer Done
Transfer /packages/db/pkginst.7286/junos-vmhost-install*.tgz
Transfer Done
Starting upgrade ...
Preparing for upgrade...
/tmp/pkg-ldX/unpack/install/
...
...
..
```

request vmhost software in-service-upgrade

Syntax `request vmhost software in-service-upgrade package-name`
 `<no-old-master-upgrade>`
 `<reboot>`

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Perform a unified in-service software upgrade (ISSU). A unified ISSU enables you to upgrade from one Junos OS release and host OS release to another with no disruption on the control plane and with minimal disruption of traffic. For an unified ISSU, graceful Routing Engine switchover (GRES) and nonstop active routing (NSR) must be enabled.

Options ***package-name***—Location from which the software package or bundle is to be installed.
 For example:

- ***/var/tmp/package-name***— For a software package or bundle that is being installed from a local directory on the router.
- ***protocol://hostname/pathname/package-name***—For a software package or bundle that is to be downloaded and installed from a remote location. Replace ***protocol*** with one of the following:
 - **ftp**—File Transfer Protocol (FTP).
 - **http**—Hypertext Transfer Protocol (HTTP).
 - **scp**—Secure Copy Protocol (SCP)(available only for Canada and U.S. version).

no-old-master-upgrade—(Optional) When the **no-old-master-upgrade** option is included, after the backup Routing Engine is rebooted with the new software package and a switchover occurs to make it the new master Routing Engine, the former master (new backup) Routing Engine is not upgraded to the new software. In this case, you must manually upgrade the former master (new backup) Routing Engine. If you do not include the **no-old-master-upgrade** option, the system automatically upgrades the former master Routing Engine.

reboot—(Optional) Automatically reboot the former master (new backup) Routing Engine after the ISSU. If you do not include the **reboot** option in the command, you

must manually reboot the former master (new backup) Routing Engine by using the **request vmhost reboot** command.

Additional Information The following conditions apply to unified ISSU:

- Unified ISSU is not supported on every platform. For a list of supported platforms, see *Unified ISSU System Requirements*.
- Unsupported PICs are restarted during a unified ISSU on certain routing devices. For information about supported PICs, see the *Junos OS High Availability Library for Routing Devices*.
- During a unified ISSU, any unsupported protocols running on the device causes packet loss. For information about supported protocols, see the *Junos OS High Availability Library for Routing Devices*.
- During a unified ISSU, you cannot bring any PICs online or take them offline on certain routing devices.

For more information, see the *Junos OS High Availability Library for Routing Devices*.

Required Privilege Level maintenance

Related Documentation

- *request system software in-service-upgrade*

List of Sample Output [request vmhost software in-service-upgrade on page 539](#)

Sample Output

[request vmhost software in-service-upgrade](#)

```
user@host> request vmhost software in-service-upgrade
/var/tmp/junos-vmhost-install-ptx-x86-64-15.1F5.6.tgz reboot
Chassis ISSU Check Done
[Feb 24 01:12:09]: Starting VMHOST ISSU
[Feb 24 01:12:09]:ISSU: Validating Image
FPC 2 will be offlined (In-Service-Upgrade not supported)
FPC 11 will be offlined (In-Service-Upgrade not supported)
MIC 11/0 will be offlined (In-Service-Upgrade not supported)
Do you want to continue with these actions being taken ? [yes,no] (no) yes

Junos Validation begin. Procedure will take few minutes.
...
...
```

request vmhost software rollback

Syntax request vmhost software rollback
<re0 | re1>
<routing-engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Roll back the Junos OS and the host OS software packages to the previous versions. You can revert to the previous versions of software packages that were loaded at the last successful **request vmhost software add** command.

Options none—Roll back the software packages of the Routing Engine on the local Virtual Chassis member.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, roll back the software packages in Routing Engine in slot 0 (**re0**) or software packages in the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine on which the software packages needs to be rolled back to the previous set of software packages. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Both Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—Other Routing Engine.

Required Privilege Level maintenance

Related Documentation

- [request vmhost software add on page 535](#)
- [request vmhost software abort on page 534](#)

List of Sample Output [request vmhost software rollback on page 541](#)

Sample Output

request vmhost software rollback

```
user@host> request vmhost software rollback
Current root details, Device sda, Label: jrootp_P, Partition: sda3
Finding alternate root for rollback
Rollback to software on jrootb_P ...
sh /etc/install/mk-mtre-rollback.sh jrootb_P b
Mounting device in preparation for rollback...
Updating boot partition for rollback...
Rollback complete, please reboot the node for it to take effect.
Cmos Write successfull
Cmos Write successfull for Boot_retry
Cmos Write successfull for Boot_retry
```

```
user@host> show vmhost version
Current root details, Device sda, Label: jrootp_P, Partition: sda3
Current boot disk: Primary
Current root set: p
UEFI Version: NGRE_v00.53.00.01
```

```
Primary Disk, Upgrade Time: Wed Feb 24 17:51:53 UTC 2016
Pending reboot.
```

```
Version: set p
VMHost Version: 2.951
VMHost Root: vmhost-x86_64-15.1I20160210_2212_builder
VMHost Core: vmhost-core-x86_64-15.1I20160210_2212_builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F5.5
```

```
Version: set b
VMHost Version: 2.953
VMHost Root: vmhost-x86_64-15.1F520160222_1052_builder
VMHost Core: vmhost-core-x86_64-15.1F520160222_1052_builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F5.6
```

```
user@host> request vmhost reboot
Reboot the vmhost ? [yes,no] (no) yes
```

```
warning: Rebooting re1
Initiating vmhost reboot... ok
Initiating Junos shutdown... shutdown: [pid 9733]
Shutdown NOW!
ok
Junos shutdown is in progress...
```

```
*** FINAL System shutdown message from root@nikon1 ***
```

```
System going down IMMEDIATELY
```

```
user@host> show vmhost version
Current root details, Device sda, Label: jrootb_P, Partition: sda4
Current boot disk: Primary
Current root set: b
```

UEFI Version: NGRE_v00.53.00.01

Primary Disk, Upgrade Time: Wed Feb 24 17:51:53 UTC 2016

Version: set p

VMHost Version: 2.951

VMHost Root: vmhost-x86_64-15.1I20160210_2212_builder

VMHost Core: vmhost-core-x86_64-15.1I20160210_2212_builder

kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt

Junos Disk: junos-install-x86-64-15.1F5.5

Version: set b

VMHost Version: 2.953

VMHost Root: vmhost-x86_64-15.1F520160222_1052_builder

VMHost Core: vmhost-core-x86_64-15.1F520160222_1052_builder

kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt

Junos Disk: junos-install-x86-64-15.1F5.6

request vmhost zeroize

Syntax request vmhost zeroize
<re0 | re1>
<routing-engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Remove all configuration information on the Routing Engines and reset all key values. If the device has dual Routing Engines, the command is broadcast to both Routing Engines on the device. The command removes all data files, including customized configuration and log files, by unlinking the files from their directories. The command removes all user-created files from the system including all plain-text passwords, secrets, and private keys for SSH, local encryption, local authentication, IPsec, RADIUS, TACACS+, and SNMP.

This command reboots the device and sets it to the factory-default configuration. After the reboot, you cannot access the device through the management Ethernet interface. Log in through the console as the root user and start the Junos OS CLI by typing **cli** at the prompt.

Options none—Remove all configuration information on all the Routing Engines and reset all key values.

re0 | re1—(Optional) Remove all configuration information on the Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Remove all configuration information on the specified Routing Engine. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Both Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—Other Routing Engine.

Required Privilege Level maintenance

Related Documentation

- [request vmhost snapshot on page 532](#)

List of Sample Output [request vmhost zeroize on page 544](#)

Sample Output

[request vmhost zeroize](#)

```
user@host> request vmhost zeroize
warning: Vmhost will reboot and may not boot without configuration
Erase all data, including configuration and log files? [yes,no] (no) yes

warning: Proceeding with vmhost zeroize
Zeroize secondary internal disk ...
mv: cannot stat '/tmp/zero-nZR/tgt_jlvmrootfs/etc/sysctl.conf': No such file or
directory
Proceeding with zeroize on secondary disk
Mounting device in preparation for zeroize...
Cleaning up target disk for zeroize ...
mv: cannot stat '/tmp/zero-nZR/sysctl.conf': No such file or directory
Zeroize done on target disk.
Zeroize of secondary disk completed
Zeroize primary internal disk ...
Proceeding with zeroize on primary disk
Mounting device in preparation for zeroize...
find: 'rm': No such file or directory
mv: cannot stat '/tmp/zero-9Dg/tgt_jlvmrootfs/etc/sysctl.conf': No such file or
directory
mv: cannot stat '/tmp/zero-9Dg/sysctl.conf': No such file or directory
Cleaning up target disk for zeroize ...
Zeroize done on target disk.
Zeroize of primary disk completed
Zeroize done
warning: Proceeding with vmhost reboot
Initiating vmhost reboot... ok
```

CHAPTER 21

VM Host Software Monitoring Commands

- `show vmhost bridge`
- `show vmhost crash`
- `show vmhost hard-disk-test`
- `show vmhost hardware`
- `show vmhost information`
- `show vmhost logs`
- `show vmhost management-if`
- `show vmhost netstat`
- `show vmhost processes`
- `show vmhost resource-usage`
- `show vmhost snapshot`
- `show vmhost status`
- `show vmhost uptime`
- `show vmhost version`

show vmhost bridge

Syntax show vmhost bridge
 <invoke-on>
 <re0 | re1>
 <routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display bridge table information. The bridge table provides information about the interfaces used for communication between host and guest operating systems.

Options **invoke-on**—(Optional) Display the bridge table information of Routing Engines on a router that has dual or redundant Routing Engines. You can use the **all-routing-engine** option to display the bridge table information of all the Routing Engines or the **other-routing-engine** option to display the bridge table information of the other Routing Engine. For example, If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display bridge table information about the Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine for which the bridge information is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and the backup Routing Engines.
- **local**— Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

List of Sample Output [show vmhost bridge on page 547](#)

Sample Output

`show vmhost bridge`

```
user@host> show vmhost bridge
Compute cluster: rainier-re-cc
Compute node: rainier-re-cn
Bridge Table
=====
bridge name          bridge id          STP enabled  interfaces
jnpr-int-br          8000.bee5a8cfdb9a  no           tap1
virbr0                8000.52540051f94b  yes          virbr0-nic
```

show vmhost crash

Syntax show vmhost crash

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display the number of times the host OS crashed. The crash dumps are available at `/var/crash`.

Required Privilege Level view

Related Documentation • [show system core-dumps](#)

List of Sample Output [show vmhost crash on page 548](#)

Sample Output

`show vmhost crash`

```
user@host> show vmhost crash
Compute cluster: rainier-re-cc

Compute node: rainier-re-cn

Crash Info
=====
total 0
```


show vmhost hard-disk-test

Syntax	<code>show vmhost hard-disk -test { disk <i>disk-name</i> status }</code>
Release Information	Command introduced in Junos OS Release 17.2R1 for the PTX5000, MX240, MX480, MX960 , MX2010, and MX2020 routers.
Description	Display memory and diagnostics monitoring test status on the solid-state drive (SSD). The test output provides information about the various attributes of the SSD that help to monitor the status of the hard disk memory. This command should be used only after initiating the disk test with the request vmhost hard-disk-test command.
Options	disk <i>disk-name</i> — Display the name of the SSD. status —Display the status of the test.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none"> • request vmhost hard-disk-test on page 525
List of Sample Output	show vmhost hard-disk-test on page 549

Sample Output

show vmhost hard-disk-test

```

user@host> show vmhost hard-disk-test status disk /dev/sda
smartctl 5.42 2014-07-28 r3460
[x86_64-linux-3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt] (local build)
Copyright (C) 2002-11 by Bruce Allen, http://smartmontools.sourceforge.net

=== START OF INFORMATION SECTION ===
Model Family:      UNIGEN SATA SSD
Device Model:      SATA SSD
Serial Number:     3AF607410C3800117282
Firmware Version:  S9FM01.3
User Capacity:     64,023,257,088 bytes [64.0 GB]
Sector Size:       512 bytes logical/physical
Device is:         In smartctl database [for details use: -P show]
ATA Version is:    8
ATA Standard is:   ACS-3 (revision not indicated)
Local Time is:     Sun Jan  8 08:02:22 2017 UTC
SMART support is:  Available - device has SMART capability.
SMART support is:  Enabled

=== START OF READ SMART DATA SECTION ===
SMART overall-health self-assessment test result: PASSED

General SMART Values:
Offline data collection status:  (0x00) Offline data collection activity
                                   was never started.

```

```

Self-test execution status:      (  0) Auto Offline Data Collection: Disabled.
                                   The previous self-test routine completed
                                   without error or no self-test has ever
                                   been run.

Total time to complete Offline  ( 30) seconds.
data collection:
Offline data collection
capabilities:      (0x1b) SMART execute Offline immediate.
                                   Auto Offline data collection on/off
support.
                                   Suspend Offline collection upon new
...
...
...

```

show vmhost hardware

Syntax show vmhost hardware
 <invoke-on>
 <re0 | re1>
 <routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display details of RAM and solid-state drives (SSDs) installed in the Routing Engine.

Options **none**—(Optional) Display information about hardware.

invoke-on—(Optional) Display the details of RAM and solid-state drives (SSDs) installed on a router that has dual Routing Engines. You can use the **all-routing-engine** option to display the hardware information of all the Routing Engines or the **other-routing-engine** option to display the hardware information of the other Routing Engine. For example, if you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display hardware information about the Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine for which the details of the installed RAM and solid-state drives (SSDs) is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**— If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

Related Documentation • *show chassis hardware*

List of Sample Output [show vmhost hardware on page 552](#)

Sample Output

[show vmhost hardware](#)

```
user@host> show vmhost hardware
Compute cluster: rainier-re-cc
```

```
Compute node: rainier-re-cn
Hardware inventory:
```

Item	Capacity	Part number	Serial number	Description
DIMM 0	16384 MB	36ADS2G72PZ-2G1A1	0x0CF49320	DDR4 2133 MHz
DIMM 1	16384 MB	36ADS2G72PZ-2G1A1	0x0CF4934C	DDR4 2133 MHz
DIMM 2	16384 MB	36ADS2G72PZ-2G1A1	0x0CF49329	DDR4 2133 MHz
DIMM 3	16384 MB	36ADS2G72PZ-2G1A1	0x0CF49352	DDR4 2133 MHz
Disk1	50.0 GB	StorFly-VSF202CC050G	P1T13003443810130041	SLIM SATA SSD
Disk2	50.0 GB	StorFly-VSF202CC050G	P1T13003443810130012	SLIM SATA SSD

show vmhost information

Syntax show vmhost information
 <invoke-on>
 <re0 | re1>
 <routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display information about the host—such as IP address of the host Routing Engine, host OS version, model number or name of the Routing Engine, and so on.

Options **invoke-on**—(Optional) Display information about the host on a router that has dual Routing Engines. You can use the **all-routing-engine** option to display information about the host of all the Routing Engines or the **other-routing-engine** option to display the information about the host of the other Routing Engine. If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine .

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display information about the host of Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine for which the information about the host is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and the backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

List of Sample Output [show vmhost information on page 554](#)

Sample Output

[show vmhost information](#)

```
user@host> show vmhost information
Compute cluster: rainier-re-cc
  Compute node      Model      Kernel release      Machine
Management IP
  rainier-re-cn     RAINIER    3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt x86_64
192.168.1.2/24
```

show vmhost logs

Syntax show vmhost logs
 <invoke-on>
 <re0 | re1>
 <routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display trace logs information of the host OS.

Options **invoke-on**—(Optional) Display the trace logs information of the host OS running on a router that has dual Routing Engines. You can use the **all-routing-engine** option to display the trace logs information of the host OS running on all the Routing Engines or the **other-routing-engine** option to display the trace logs information of the host OS running on the other Routing Engine. For example, If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display trace logs information of the host os running on the Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine for which the trace logs information of the host OS is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and the backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

List of Sample Output [show vmhost logs on page 556](#)

Sample Output

show vmhost logs

```
user@host> show vmhost logs
Compute cluster: rainier-re-cc

Compute node: rainier-re-cn

Logs Info
=====
total 104844
drwxr-xr-x. 2 root root      4096 Dec 10 19:56 sa
-rw-r--r--. 1 root root       400 Dec 10 19:56 postinstall.log
drwxr-xr-x. 2 root root      4096 Dec 10 19:56 audit
drwxr-xr-x. 5 root root      4096 Dec 10 19:56 glusterfs
drwxr-xr-x. 2 root root      4096 Dec 10 19:56 wdmd_disk_io
drwxr-xr-x. 2 root root      4096 Dec 10 19:56 openvswitch
drwxr-xr-x. 3 root root      4096 Dec 10 20:05 libvirt
-rw-r-----. 1 root root 228224 Dec 11 00:00 syslog-20151211.gz
-rw-r-----. 1 root root 987114 Dec 12 00:00 syslog-20151212.gz
-rw-r-----. 1 root root   3100 Dec 12 20:12 mail.log-20151213
-rw-r-----. 1 root root   3100 Dec 12 20:12 mail.info-20151213
-rw-r-----. 1 root root   5730 Dec 12 20:15 user.log-20151213
-rw-r-----. 1 root root 434831 Dec 12 23:52 kern.log-20151213
-rw-r-----. 1 root root 7349109 Dec 12 23:59 debug-20151213
-rw-r-----. 1 root root 955140 Dec 13 00:00 syslog-20151213.gz
-rw-r-----. 1 root root 266098 Dec 13 00:00 messages-20151213
-rw-r-----. 1 root root 10252576 Dec 13 00:00 auth.log-20151213
-rw-r--r--. 1 root root 26464360 Dec 13 04:11 daemon.log-20151213
-rw-r-----. 1 root root    242 Dec 13 04:12 mail.warn-20151213
-rw-r-----. 1 root root    242 Dec 13 04:12 mail.err-20151213
-rw-r-----. 1 root root 12373338 Dec 14 00:00 syslog-20151214
-rw-r-----. 1 root adm    1592 Dec 14 00:10 boot.4.gz
-rw-r-----. 1 root adm    1570 Dec 14 00:42 boot.3.gz
-rw-r-----. 1 root adm    1552 Dec 14 06:38 boot.2.gz
-rw-r-----. 1 root adm    1589 Dec 14 07:54 boot.1.gz
-rw-r-----. 1 root adm    5186 Dec 14 18:50 boot.0
-rw-r--r--. 1 root root    292 Dec 14 21:57 lastlog
-rw-r--r--. 1 root root   1638 Dec 14 21:57 sanlock.log
-rw-r-----. 1 root root   1098 Dec 14 21:57 mail.warn
-rw-r-----. 1 root root   8939 Dec 14 21:57 mail.log
-rw-r-----. 1 root root   8939 Dec 14 21:57 mail.info
-rw-r-----. 1 root root    968 Dec 14 21:57 mail.err
-rw-r-----. 1 root adm    5077 Dec 14 21:57 boot
-rw-rw-r--. 1 root root   61824 Dec 14 21:57 wtmp
--w-r-----. 1 root root   80275 Dec 14 21:57 resild
-rw-r-----. 1 root root   31314 Dec 14 21:59 user.log
-rw-r-----. 1 root root  951929 Dec 14 22:57 messages
-rw-r-----. 1 root root 1577908 Dec 14 22:57 kern.log
-rw-r-----. 1 root root  4810073 Dec 14 23:10 auth.log
-rw-r-----. 1 root root 11130442 Dec 14 23:14 syslog
-rw-r-----. 1 root root  7305132 Dec 14 23:14 debug
-rw-r--r--. 1 root root 21884828 Dec 14 23:14 daemon.log
```

show vmhost management-if

Syntax show vmhost management-if

Release Information Command introduced in Junos OS Release 15.1F6.



NOTE: The command is supported on the routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines only.

Description Display the administrative status, speed and operational mode of the host interface eth0, which serves as a management interface.

Required Privilege Level view

List of Sample Output [show vmhost management-if on page 557](#)

Sample Output

show vmhost management-if

```
user@host> show vmhost management-if
Administrative status: Up
Link status: Up
Link speed: 1000Mb/s
Link operational mode: Full
```

show vmhost netstat

Syntax `show vmhost netstat`
 `<invoke-on>`
 `<re0 | re1>`
 `<routing engine>`

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display network statistics information for the host OS. The statistics contains information related to the interfaces used for the communication between the host and the guest, such as the IP address of the destination, IP address of the gateway, mask, flags, and so on.

Options **invoke-on**—(Optional) Display the network statistics for the host OS on a router that has dual Routing Engines. You can use the **all-routing-engine** option to display the network statistics information for the host OS running on all the Routing Engines or the **other-routing-engine** option to display the network statistics information for the host OS running on the other Routing Engine. For example, If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine .

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display the network statistics information for the host OS running on the Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine for which the network statistics information for the host OS is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and the backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

List of Sample Output [show vmhost netstat on page 559](#)

Sample Output

`show vmhost netstat`

```
user@host> show vmhost netstat
Compute cluster: rainier-re-cc

Compute node: rainier-re-cn

Netstat
=====
Kernel IP routing table
Destination      Gateway          Genmask          Flags   MSS Window  irtt Iface
0.0.0.0          10.216.63.254   0.0.0.0          UG      0 0        0 eth0
10.216.48.0      0.0.0.0         255.255.240.0    U       0 0        0 eth0
192.168.1.0      0.0.0.0         255.255.255.0    U       0 0        0
jnpr-int-br
192.168.122.0    0.0.0.0         255.255.255.0    U       0 0        0 virbr0
```

show vmhost processes

Syntax show vmhost processes
 <invoke-on>
 <re0 | re1>
 <routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Description Display information about the host processes that are running on the router.

Options **invoke-on**—(Optional) Display information about the host processes that are running on a router with dual Routing Engines. You can use the **all-routing-engine** option to display information about the host processes running on all the Routing Engines or the **other-routing-engine** option to display information about the host processes running on the other Routing Engine. For example, If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine .

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display information about the host processes running on the Routing Engine in slot 0 (re0) or on the Routing Engine in slot 1 (re1).

routing-engine—(Optional) Specify the Routing Engine for which the information about the host processes is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and the backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

Related Documentation • [show system processes](#)

List of Sample Output [show vmhost processes on page 561](#)

Sample Output

show vmhost processes

```
user@host> show vmhost processes
Compute cluster: rainier-re-cc
```

```
Compute node: rainier-re-cn
  UID      PID  PPID  C  STIME TTY      TIME CMD
  root      1    0    0  21:56 ?      00:00:02 init [3]
  root      2    0    0  21:56 ?      00:00:00 [kthreadd]
  root      3    2    0  21:56 ?      00:00:04 [ksoftirqd/0]
  root      5    2    0  21:56 ?      00:00:00 [kworker/0:0H]
  root      7    2    0  21:56 ?      00:00:00 [posixcpumr/0]
  root      8    2    0  21:56 ?      00:00:00 [kclksetdelayd]
  root      9    2    0  21:56 ?      00:00:00 [rcub/0]
  root     10    2    0  21:56 ?      00:00:04 [rcu_preempt]
  root     11    2    0  21:56 ?      00:00:00 [rcu_sched]
  root     12    2    0  21:56 ?      00:00:00 [rcu_bh]
  root     13    2    0  21:56 ?      00:00:03 [rcuc/0]
  root     14    2    0  21:56 ?      00:00:00 [kcmosdelayd]
  root     15    2    0  21:56 ?      00:00:00 [migration/0]
  root     16    2    0  21:56 ?      00:00:00 [migration/1]
  root     17    2    0  21:56 ?      00:00:03 [rcuc/1]
  root     18    2    0  21:56 ?      00:00:04 [ksoftirqd/1]
  root     19    2    0  21:56 ?      00:00:00 [posixcpumr/1]
  root     20    2    0  21:56 ?      00:00:00 [kworker/1:0]
  root     21    2    0  21:56 ?      00:00:00 [kworker/1:0H]
  root     22    2    0  21:56 ?      00:00:00 [migration/2]
  root     23    2    0  21:56 ?      00:00:10 [rcuc/2]
  root     24    2    0  21:56 ?      00:00:02 [ksoftirqd/2]
  root     25    2    0  21:56 ?      00:00:00 [posixcpumr/2]
  root     26    2    0  21:56 ?      00:00:00 [kworker/2:0]
  root     27    2    0  21:56 ?      00:00:00 [kworker/2:0H]
  root     28    2    0  21:56 ?      00:00:00 [migration/3]
  root     29    2    0  21:56 ?      00:00:01 [rcuc/3]
  root     30    2    0  21:56 ?      00:00:01 [ksoftirqd/3]
  root     31    2    0  21:56 ?      00:00:00 [posixcpumr/3]
  root     32    2    0  21:56 ?      00:00:00 [kworker/3:0]
  root     33    2    0  21:56 ?      00:00:00 [kworker/3:0H]
  root     34    2    0  21:56 ?      00:00:00 [migration/4]
  root     35    2    0  21:56 ?      00:00:01 [rcuc/4]
  root     36    2    0  21:56 ?      00:00:01 [ksoftirqd/4]
  root     37    2    0  21:56 ?      00:00:00 [posixcpumr/4]
  root     38    2    0  21:56 ?      00:00:00 [kworker/4:0]
  root     39    2    0  21:56 ?      00:00:00 [kworker/4:0H]
  root     40    2    0  21:56 ?      00:00:00 [migration/5]
  root     41    2    0  21:56 ?      00:00:01 [rcuc/5]
```

show vmhost resource-usage

Syntax show vmhost resource-usage
 <invoke-on>
 <re0 | re1>
 <routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display the current usage of solid-state drive (SSD), RAM, and CPU resources of the host OS.

Options **invoke-on**—(Optional) Display information about resources used by the host OS running on a router that has dual Routing Engines. You can use the **all-routing-engine** option to display information about resources used by the host OS on all the Routing Engines or the **other-routing-engine** option to display information about resources used by the host OS on the other Routing Engine. For example, If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display information about resources used by the host OS on the Routing Engine in slot 0 (re0) or on the Routing Engine in slot 1 (re1).

routing-engine—(Optional) Specify the Routing Engine for which the information about resources used by the host OS is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and the backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

List of Sample Output [show vmhost resource-usage on page 563](#)

Sample Output

show vmhost resource-usage

```

user@host> show vmhost resource-usage
Compute cluster: rainier-re-cc

Compute node: rainier-re-cn
CPU Usage
=====
23:15:09 CPU %usr %nice %sys %iowait %irq %soft %steal %guest
%idle
23:15:09 all 0.36 0.00 1.16 0.07 0.00 0.01 0.00 74.07
24.34
23:15:09 0 1.42 0.00 4.46 0.28 0.00 0.03 0.00 0.00
93.81
23:15:09 1 1.43 0.00 3.87 0.30 0.00 0.03 0.00 0.00
94.38
23:15:09 2 0.02 0.00 0.42 0.00 0.00 0.01 0.00 98.33
1.22
23:15:09 3 0.00 0.00 0.14 0.00 0.00 0.00 0.00 98.65
1.21
23:15:09 4 0.00 0.00 0.09 0.00 0.00 0.00 0.00 98.71
1.19
23:15:09 5 0.00 0.00 0.10 0.00 0.00 0.00 0.00 98.71
1.19
23:15:09 6 0.00 0.00 0.11 0.00 0.00 0.00 0.00 98.70
1.19
23:15:09 7 0.00 0.00 0.12 0.00 0.00 0.00 0.00 98.68
1.19

Memory Usage
=====
total used free shared buffers cached
Mem: 63851 51388 12462 0 35 123
Swap: 0 0 0

Disk Usage
=====
Filesystem Size Used Avail Use% Mounted on
rootfs 3.3G 127M 3.0G 5% /
/dev/sda4 2.0G 1.7G 166M 91% /.old
tmpfs 32G 1.2M 32G 1% /.old/run
none 32G 4.0K 32G 1% /.old/dev
tmpfs 32G 0 32G 0% /.old/tmp
tmpfs 32G 0 32G 0% /.old/tdev
/dev/mapper/jvg_P-jlvmrootrw 3.3G 127M 3.0G 5% /.old/tdev/.union/.s
unionfs 3.3G 127M 3.0G 5% /
none 32G 4.0K 32G 1% /dev
tmpfs 32G 180K 32G 1% /run
tmpfs 32G 8.0K 32G 1% /var/volatile
/dev/mapper/jvg_P-jlvmjunos 32G 13G 18G 43% /junos
/dev/mapper/jvg_P-jlvmvm 6.1G 2.7G 3.1G 47% /vm
/dev/mapper/jvg_P-jlvm spare 287M 2.1M 266M 1% /spare
cgroup 32G 0 32G 0% /sys/fs/cgroup
unionfs 3.3G 127M 3.0G 5% /run/named-chroot/etc/bind

tmpfs 32G 180K 32G 1%
/run/named-chroot/var/run/named

```

tmpfs	32G	180K	32G	1%	
/run/named-chroot/var/run/bind					
unionfs	3.3G	127M	3.0G	5%	
/run/named-chroot/var/cache/bind					
unionfs	3.3G	127M	3.0G	5%	
/run/named-chroot/etc/localtime					
none	32G	4.0K	32G	1%	/run/named-chroot/dev/random
none	32G	4.0K	32G	1%	/run/named-chroot/dev/zero
none	32G	4.0K	32G	1%	/run/named-chroot/dev/null

show vmhost snapshot

Syntax show vmhost snapshot
 <invoke-on>
 <re0 | re1>
 <routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display snapshot details including Linux host kernel version, software version, and other package version details for both the sets of software in the backup disk.

Options **invoke-on**—(Optional) Display the host snapshot information of Routing Engines on a router that has dual Routing Engines. You can use the **all-routing-engine** option to display the host snapshot information of all the Routing Engines or the **other-routing-engine** option to display the host snapshot information of the other Routing Engine. For example, If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display host snapshot information about the Routing Engine in slot 0 (**re0**) or the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine for which the host snapshot details is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

- Related Documentation**
- [request vmhost software rollback on page 540](#)
 - [request vmhost snapshot on page 532](#)

List of Sample Output [show vmhost snapshot on page 566](#)

Sample Output

show vmhost snapshot

```
user@host> show vmhost snapshot
UEFI      Version: NGRE_v00.53.00.01

Secondary Disk, Snapshot Time: Tue Dec  8 19:49:09 UTC 2015

Version: set p
VMHost Version: 2.897
VMHost Root: vmhost-x86_64-15.1I20151203_0011_rbu-builder
VMHost Core: vmhost-core-x86_64-15.1I20151203_0011_rbu-builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F-20151204.0

Version: set b
VMHost Version: 2.897
VMHost Root: vmhost-x86_64-15.1I20151203_0011_rbu-builder
VMHost Core: vmhost-core-x86_64-15.1I20151203_0011_rbu-builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F-20151204.0
```

show vmhost status

Syntax show vmhost status
 <invoke-on>
 <re0 | re1>
 <routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display information about the status of communication between the host OS and the guest OS. The following status outputs are displayed:

- **Online**—Communication between the host OS and the guest OS is good.
- **Offline**—Communication with the host is lost. Any state other than **Online** is considered as **Offline**.

Options **invoke-on**—(Optional) Display the status of communication between the host OS and the guest OS running on a router with dual Routing Engines. You can use the **all-routing-engine** option to display the status of host-to-guest communication on all the Routing Engines or the **other-routing-engine** option to display the status of host-to-guest communication on the other Routing Engine. For example, If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display the status of communication between the host OS and the guest OS on the Routing Engine in slot 0 (**re0**) or on the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine for which the status of communication between the host OS and the guest OS is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.

- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

List of Sample Output [show vmhost status on page 568](#)

Sample Output

[show vmhost status](#)

```
user@host> show vmhost status
Compute cluster: rainier-re-cc
Compute Node: rainier-re-cn, Online
```

show vmhost uptime

Syntax show vmhost uptime
 <invoke-on>
 <re0 | re1>
 <routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display the current time and information such as how long the host OS has been running, number of users, average load, and reason for the last reboot that occurred.

Options **invoke-on**—(Optional) Display the uptime information about the host on a router with dual Routing Engines. You can use the **all-routing-engine** option to display the uptime information about the host on all the Routing Engines or the **other-routing-engine** option to display the uptime information about the host on the other Routing Engine. For example, If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display the uptime information about the host on the Routing Engine in slot 0 (**re0**) or on the Routing Engine in slot 1 (**re1**).

routing-engine—(Optional) Specify the Routing Engine for which the uptime information about the host is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

**Related
Documentation**

- *show chassis routing-engine*

List of Sample Output [show vmhost uptime on page 570](#)

Sample Output

[show vmhost uptime](#)

```
user@host> show vmhost uptime
Vmhost Current time: 2015-12-14 23:16:01+00:00
Vmhost Uptime:
    23:16:01 up 1:19, 0 users, load average: 6.22, 6.14, 6.07
Vmhost last reboot reason: 0x2000:hypervisor reboot
```

show vmhost version

Syntax show vmhost version
 <invoke-on>
 <re0 | re1>
 <routing engine>

Release Information Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
 Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
 Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.



NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.

Description Display host version information including Linux host kernel version, host software version, and other package version details for both the sets of software in the primary disk.

Options **invoke-on**—(Optional) Display the version information of the host running on a router with dual Routing Engines. You can use the **all-routing-engine** option to display the version information of the host software running on all the Routing Engines or the **other-routing-engine** option to display the version information of the host software running on other Routing Engine. For example, If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

re0 | re1—(Optional) On routers that support dual or redundant Routing Engines, display version information of the host software on the Routing Engine in slot 0 (re0) or on the Routing Engine in slot 1 (re1).

routing-engine—(Optional) Specify the Routing Engine for which the version information of the host software is to be displayed. The following options are available:

- **backup**—Backup Routing Engine.
- **both**—Master and backup Routing Engines.
- **local**—Routing Engine on the local Virtual Chassis member.
- **master**—Master Routing Engine.
- **other**—If you issue the command from the master Routing Engine, the backup Routing Engine is referred to as the other Routing Engine.

Required Privilege Level view

List of Sample Output [show vmhost version on page 572](#)

Sample Output

show vmhost version

```
user@host> show vmhost version
Current root details,          Device sda, Label: jrootb_P, Partition: sda4
Current boot disk: Primary
Current root set: b
UEFI    Version: NGRE_v00.53.00.01

Primary Disk, Upgrade Time: Mon Dec 14 21:55:38 UTC 2015

Version: set p
VMHost Version: 2.900
VMHost Root: vmhost-x86_64-15.1F420151130_1049_builder
VMHost Core: vmhost-core-x86_64-15.1F420151130_1049_builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F4.10

Version: set b
VMHost Version: 2.901
VMHost Root: vmhost-x86_64-15.1I20151210_0011_rbu-builder
VMHost Core: vmhost-core-x86_64-15.1I20151210_0011_rbu-builder
kernel: 3.10.79-ovp-rt74-WR6.0.0.20_preempt-rt
Junos Disk: junos-install-x86-64-15.1F-20151211.0
```