



Junos[®] OS

Installation and Upgrade Guide



Modified: 2018-04-09

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 © 2018 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 <https://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
Chapter 1	Software Overview	29
	Junos OS Overview	29
	One Operating System	30
	One Modular Software Architecture	30
	Secure Boot	30
	Junos OS with Upgraded FreeBSD	31
	Release Information for Junos OS with Upgraded FreeBSD	32
	Changes in Package Names for Junos OS with Upgraded FreeBSD	33
	Linux-Based Platforms Package Names	33
	EX Series Switches Package Names	34
	MX Series Routers Package Names	35
	QFX Series and EX4600 Switches Package Names	36
	SRX5400, SRX5600, and SRX5800 Devices Package Names	37
	Changes in Commands and Statements in Junos OS with Upgraded FreeBSD	37
	Changes in Disk Volumes for Junos OS with Upgraded FreeBSD	39
	/junos Volume	40
	/oam Volume	40
	Changes in Use of Snapshots for Junos OS with Upgraded FreeBSD	41
	Recovery Snapshots	41
	Non-Recovery Snapshots	42
	Junos OS Installation Package Names	43
	Junos OS Installation Packages Prefixes	44
	Junos OS Release Numbers	45
	Junos OS Editions	47
	FIPS 140-2 Security Compliance	47
	Boot Sequence on Devices with Routing Engines	48
	Boot Order for Devices	48
	Bootling from an Alternate Boot Device	50

	Configuration Files	50
	Configuration File Selection Sequence	51
	Remote Storage of Configuration Files	52
	Understanding Download Manager for SRX Series Devices	52
	Overview	52
	Using Download Manager to Upgrade Junos OS	53
	Handling Errors	53
	Considerations	54
Chapter 2	Installing, Upgrading, and Downgrading Software	55
	Software Installation Overview	56
	Installation Type Overview	58
	Standard Installation	58
	Category Change Installation	58
	Recovery Installation	58
	Understanding Software Installation on EX Series Switches	59
	Overview of the Software Installation Process	59
	Software Package Security	59
	Installing Software on a Virtual Chassis	60
	Installing Software on Switches with Redundant Routing Engines	60
	Installing Software Using Automatic Software Download	60
	Autoinstalling a Configuration File on an EX2200 or EX3300 Switch from a Disk-on-Key USB Memory Stick	61
	Installing Software on an EX2300 or EX3400 Switch	61
	Configuration Image Validation on EX Series Switches	61
	Troubleshooting Software Installation	61
	Understanding Junos OS Upgrades for SRX Series Devices	61
	Understanding Junos OS Upgrades	62
	Junos OS Upgrade Methods on the SRX Series Devices	62
	Overview of Upgrading to 64-bit Junos OS	63
	Upgrading Redundant Routing Engines from 32-bit to 64-bit Junos OS	64
	Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using One Slot	66
	Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using Two Slots	66
	Overview of CoS Upgrade Requirements (Junos OS Release 11.1 or 11.2 to a Later Release)	68
	Understanding How to Back Up an Installation on Switches	70
	Understanding System Snapshot on QFX Switches	70
	Understanding System Snapshot on EX Series Switches	70
	Backing Up the Existing Installation on Routers	72
	Ensuring Sufficient Disk Space for Junos OS Upgrades on SRX Devices	74
	Verifying Available Disk Space on SRX Series Devices	74
	Cleaning Up the System File Storage Space	75

Downloading Software	76
Downloading Software by Using a Browser	76
Installing Software Using the Command-Line Interface	77
Installing Software Packages on QFX Devices	79
Installing a Standard Software Package on QFX5100, QFX5110, QFX5200, QFX5210, and EX4600 Switches	79
Installing a Standard Software Package on QFX10002 Switches	80
Upgrading Software from Junos OS Release 15.1X53-D3X to Junos OS Release 15.1X53-D60, 15.1X53-D61.7, 15.1X53-D62, and 15.1X53-D63 on QFX10008 and QFX10016 Switches	82
Installing a Software Package on QFX10008 and QFX10016 Switches	84
Preparing the Switch for Installation	85
Installing Software on the Backup Routing Engine	85
Installing Software on the Master Routing Engine	86
Installing the Software Package on a Device with a Single Routing Engine	88
Installing Software on an EX Series Switch with a Single Routing Engine (CLI Procedure)	90
Installing the Software Package on a Router with Redundant Routing Engines	92
Preparing the Router for the Installation	92
Installing Software on the Backup Routing Engine	93
Installing Software on the Remaining Routing Engine	95
Finalizing the Installation	96
Installing Software on an EX Series Switch with Redundant Routing Engines (CLI Procedure)	98
Preparing the Switch for the Software Installation	99
Installing Software on the Backup Routing Engine	100
Installing Software on the Default Master Routing Engine	101
Returning Routing Control to the Default Master Routing Engine (Optional)	102
Upgrading Junos OS with Upgraded FreeBSD	103
Determining Which Package or Packages to Install	103
Installing Junos OS with Upgraded FreeBSD Over Plain Junos OS	106
Installing Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Different Release	108
Downgrading Junos OS from Upgraded FreeBSD	109
Downgrading from Junos OS with Upgraded FreeBSD to Junos OS	109
Downgrading from Junos OS with Upgraded FreeBSD Release 17.4 or Later to an Earlier Release of Junos OS with Upgraded FreeBSD	110
Downgrading from Junos OS with Upgraded FreeBSD Release 17.3 or Earlier to an Earlier Release of Junos OS with Upgraded FreeBSD	111
Installing Junos OS Software with Junos Automation Enhancements	112
Upgrading Software by Using Automatic Software Download	117
Upgrading the Loader Software on the Line Cards in a Standalone EX8200 Switch or an EX8200 Virtual Chassis	117
Example: Installing Junos OS Upgrade Packages on SRX Series Devices	120
Reverting the Junos OS Software Image Back to the Previous Version	123

	Upgrading Jloader Software on QFX Series Devices	125
	Jloader Software Version 1.1.4 Guidelines	127
	Upgrading Jloader Software on a QFX3500 Switch	128
	Upgrading Jloader Software on a QFabric System	131
Chapter 3	Using the USB Flash Drive and Boot Loader to Upgrade Software	139
	Preparing the USB Flash Drive to Upgrade Junos OS on SRX Series Devices . . .	139
	Installing Junos OS on SRX Series Devices Using a USB Flash Drive	141
	Booting an EX Series Switch Using a Software Package Stored on a USB Flash Drive	142
	Upgrading the Boot Loader on SRX Series Devices	143
	Installing Junos OS on SRX Series Devices from the Boot Loader Using a TFTP Server	144
	Installing Junos OS on SRX Series Devices from the Boot Loader Using a USB Storage Device	147
	Verifying Junos OS and Boot Loader Software Versions on an EX Series Switch	147
	Verifying the Number of Partitions and File System Mountings	148
	Verifying the Loader Software Version	149
	Verifying Which Root Partition Is Active	149
	Verifying the Junos OS Version in Each Root Partition	150
Chapter 4	Upgrading the BIOS and Firmware	153
	Before You Begin Installing or Upgrading the Firmware	153
	Understanding BIOS Upgrades on SRX Series Devices	155
	Understanding Manual BIOS Upgrade Using the Junos CLI	155
	Understanding Auto BIOS Upgrade Methods on SRX Series Devices	156
	Disabling Auto BIOS Upgrade on SRX Series Devices	157
	Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC (PTX-5-100G-WDM)	158
	Upgrading Firmware on the 5-Port 100-Gigabit DWDM OTN PIC (PTX-5-100G-WDM)	159
	Installing Firmware on the 100-Gigabit DWDM OTN MIC (MIC3-100G-DWDM)	161
	Upgrading Firmware on the 100-Gigabit DWDM OTN MIC (MIC3-100G-DWDM)	162
Chapter 5	Reinstalling Software	165
	Checklist for Reinstalling Junos OS	165
	Log the Software Version Information	167
	Log the Hardware Version Information	168
	Log the Chassis Environment Information	169
	Log the System Boot-Message Information	169
	Log the Active Configuration	171
	Log the Interfaces on the Router	171
	Log the BGP, IS-IS, and OSPF Adjacency Information	172
	Log the System Storage Information	173
	Back Up the Currently Running and Active File System	174

	Reinstall Junos OS	175
	Reconfigure Junos OS	175
	Configure Host Names, Domain Names, and IP Addresses	175
	Protecting Network Security by Configuring the Root Password	177
	Check Network Connectivity	179
	Copy Backup Configurations to the Router	179
	Configure Host Names, Domain Names, and IP Addresses	179
	Protecting Network Security by Configuring the Root Password	181
	Check Network Connectivity	183
	Copy Backup Configurations to the Router	183
	After You Reinstall Junos OS	183
	Compare Information Logged Before and After the Reinstall	183
	Back Up the New Software	184
	Compare Information Logged Before and After the Reinstall	184
	Back Up the New Software	185
Chapter 6	Configuring Dual-Root Partitions on Switches	187
	Configuring Dual-Root Partitions	187
	Resilient Dual-Root Partition Scheme (Junos OS Release 10.4R3 and Later)	188
	Automatic Fixing of Corrupted Primary Root Partition with the Automatic Snapshot Feature	188
	Earlier Partition Scheme (Junos OS Release 10.4R2 and Earlier)	189
	Understanding Upgrading or Downgrading Between Resilient Dual-Root Partition Releases and Earlier Releases	190
	Dual-Root Partitioning Scheme on SRX Series Devices	190
	Boot Media and Boot Partition on SRX Series Devices	191
	Important Features of the Dual-Root Partitioning Scheme	192
	Understanding Automatic Recovery of the Primary Junos OS Image with Dual-Root Partitioning	193
	Understanding How the Primary Junos OS Image with Dual-Root Partitioning Recovers Devices	194
	Understanding How Junos OS Release 10.0 or Later Upgrades with Dual-Root Partitioning	196
	Example: Installing Junos OS on SRX Series Devices Using the Partition Option	196
	Reinstalling the Single-Root Partition on SRX Series Devices	200
Chapter 7	Storage Media	203
	Routing Engines and Storage Media	203
	System Memory and Storage Media on Routers	204
	System Memory	205
	Storage Media	206
	System Memory and Storage Media for SRX Series Services Gateways	207
	SRX Series Device Overview	207
	System Memory	208

	Storage Media	208
	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) . . .	210
	Storage Media Names for SRX Series Devices	212
	Repartitioning Routing Engine System Storage to Increase the Swap Partition	212
	Accessing USB Storage on PTX1000 Routers	213
Chapter 8	Performing a Recovery Installation	215
	Creating a Snapshot and Using It to Boot an EX Series Switch	215
	Creating a Snapshot on a USB Flash Drive and Using It to Boot the Switch	216
	Creating a Snapshot and Using It to Boot a QFX3500 and QFX3600 Series Switch	216
	Creating a Snapshot on a USB Flash Drive and Using It to Boot the Switch	216
	Creating a Snapshot on an Internal Flash Drive and Using It to Boot the Switch	217
	Creating a Snapshot on the Alternate Slice of the Boot Media	218
	Creating a Snapshot and Using It to Boot a Device	218
	Creating a Snapshot on an External USB Flash Drive and Using It to Boot the Device	218
	Example: Creating a Snapshot and Using It to Boot an SRX Series Device	220
	Creating an Emergency Boot Device for Routers	223
	Creating an Emergency Boot Device for QFX Series Switches	224
	Performing a Recovery Installation Using an Emergency Boot Device	226
	Performing a Recovery Installation	228
	Recovering from a Failed Software Installation	229
	Installing and Recovering Software Using the Open Network Install Environment (ONIE)	231
	Understanding the Open Network Install Environment	233
	Downloading Software Files with a Browser	233
	Connecting to the Console Port	234
	Backing Up the Current Configuration Files	234
	Uninstalling the Existing Version of Junos OS	234
	Installing a Junos OS Software Package That Resides on a Webserver or DHCP Server with DHCP Options Configured	235
	Installing Junos OS Software Using Secure Copy Protocol (SCP)	236
	Installing Junos OS Software Using FTP or TFTP Without a Webserver . . .	237
	Installing Junos OS Software Using DHCP Server with No DHCP Options Configured	238
	Installing Junos OS Software Using Webserver Without DHCP Configured	239
	Installing Junos OS Software Using USB Media	240
	Verifying Software Installation	241
	Troubleshooting Boot Problems	241
	Creating an Emergency Boot Device	242

	Performing a Recovery Installation	243
	Understanding Integrity Check and Autorecovery of Configuration, Licenses, and	
	Disk Information on SRX Series Devices	244
	Overview	244
	How Autorecovery Works	244
	How to Use Autorecovery	245
	Data That Is Backed Up in an Autorecovery	245
	Troubleshooting Alarms	245
	Considerations	246
	Saving a Rescue Configuration File	246
	Restoring a Saved Configuration	247
	Copy Saved Files to the Router	247
	Loading and Committing the Configuration File	248
	Reverting to the Default Factory Configuration by Using the request system	
	zeroize Command	249
	Reverting to the Rescue Configuration	249
	Restarting and Halting SRX Series Devices	250
	Rebooting SRX Series Devices	250
	Halting SRX Series Devices	252
	Bringing Chassis Components Online and Offline on SRX Series Devices . .	254
	Restarting the Chassis on SRX Series Devices	255
Chapter 9	Zero Touch Provisioning	257
	Understanding Zero Touch Provisioning	257
	Configuring Zero Touch Provisioning	261
	Configuring Zero Touch Provisioning on SRX Series Devices	266
	Understanding Zero Touch Provisioning on SRX Series Devices	267
	Understanding ZTP on SRX Series Devices	267
	Network Activator Overview	267
	Limitations	270
	Configuring Zero-Touch Provisioning on an SRX Series Device	270
	Understanding Factory-Default Configuration on SRX Series Device for Zero	
	Touch Provisioning	273
	Monitoring Zero Touch Provisioning	274
	Using the Console to Monitor Zero Touch Provisioning	274
	Using System Log Alerts to Monitor Zero Touch Provisioning	275
	Using Error Messages to Monitor Zero Touch Provisioning	275
	Using System Log Files to Monitor Zero Touch Provisioning	275
	Using the show dhcp client binding Command	276
	Using the show dhcp client statistics Command	276
Chapter 10	Automatic Installation of Configuration Files	279
	Autoinstallation Overview	279
	Automatic Installation of Configuration Files	280
	Supported Autoinstallation Interfaces and Protocols	280
	Typical Autoinstallation Process on a New Device	281
	Understanding Autoinstallation of Configuration Files	285
	Typical Uses for Autoinstallation	285
	Autoinstallation Configuration Files and IP Addresses	285

	Typical Autoinstallation Process on a New Switch	286
	Configuring Autoinstallation of Configuration Files (CLI Procedure)	287
	Verifying Autoinstallation Status	289
	Example: Configuring Autoinstallation on SRX Series Devices	290
	Autoinstalling a Configuration File from a Disk-on-Key USB Memory Stick onto an EX2200 or EX3300 Switch	293
	Configuring Autoinstallation on JNU Satellite Devices	295
	Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group	298
	Supported Autoinstallation Interfaces and Protocols	298
	Typical Autoinstallation Process on a New Router	298
	Autoinstallation of Satellite Devices in a Junos Node Unifier Group	300
	Verifying Autoinstallation on JNU Satellite Devices	301
Chapter 11	Installation, Upgrade, and Recovery of VM Host Support on Devices with Routing Engines	303
	Routing Engines with VM Host Support	303
	Architecture of Routing Engines with VM Host Support	304
	Salient Features of the Routing Engines with VM Host Support	305
	Platform Virtualization	305
	Hardware Assisted Paravirtualized Guest Junos OS	306
	Guest Junos OS to Serve as the Administrative Framework	306
	Storage Partitioning and Redundancy	306
	NTP and Time Zone	309
	Autorecovery	309
	Handling Reboot and Power Off	309
	Routers with VM Host Support-Boot Process	310
	Booting for the First Time	310
	Boot Sequence	311
	Understanding Console Port	311
	Understanding Hostnames Synchronization	311
	VM Host Installation	312
	VM Host Upgrade	313
	VM Host Rollback	314
	VM Host Snapshot	316
	Disk Recovery Using the VM Host Snapshot	316
	Copying VM Host Installation Package to the PXE Boot Server	316
	Creating an Emergency Boot Device for Routing Engines with VM Host Support	318
	Upgrading the SSD Firmware on Routing Engines with VM Host Support	320
	Disabling Autorecovery on Routing Engines with VM Host Support	322
	VM Host Operations and Management	322
Chapter 12	Installing and Managing Software Licenses	325
	Junos OS Feature Licenses	325
	License Key Components for the EX Series Switch	326
	Understanding Software Licenses for EX Series Switches	327
	Purchasing a Software Feature License	327
	Features Requiring a License on EX2200 Switches	328
	Features Requiring a License on EX2300 Switches	329
	Features Requiring a License on EX3300 Switches	329

Features Requiring a License on EX3400 Switches	331
Features Requiring a License on EX4300 Switches	332
Features Requiring a License on EX4600 Switches	333
Features Requiring a License on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches	334
License Warning Messages	335
License Enforcement	336
Software Feature Licenses	337
Software Features That Require Licenses on M Series, MX Series, and T Series Routers	338
Software Features That Require Licenses on M Series Routers Only	341
Software Features That Require Licenses on MX Series Routers Only	342
Software Feature Licenses for SRX Series Devices	348
Software Features That Require Licenses on EX Series Switches	348
Software Features That Require Licenses on the QFX Series	349
Disaggregated Software Features That Require Licenses on the QFX Series	352
Disaggregated Software Feature Licenses on QFX5200 Switches	353
Junos OS Feature License Keys	354
Release-Tied License Keys and Upgrade Licenses on MX Series Routers	354
Licensable Ports on MX5, MX10, and MX40 Routers	355
Port Activation on MX104 Routers	356
Managing Licenses for the EX Series Switch (CLI Procedure)	357
Adding New Licenses	358
Deleting Licenses	358
Saving License Keys	358
Monitoring Licenses for the EX Series Switch	358
Displaying Installed Licenses and License Usage Details	359
Displaying Installed License Keys	360
Generating License Keys	360
Adding New Licenses (CLI Procedure)	362
Installing a License Using a Configuration Statement	362
Installing Licenses Using the CLI Directly	363
Installing Licenses Using a Configuration File	364
Installing a License Using an Operational Command	366
Adding a License to a Device with a Single Routing Engine	366
Adding a License to a Device with Dual Routing Engines	366
Deleting License Keys (CLI)	367
Using the Operational Command to Delete Licenses	367
Using a Configuration Command to Delete Licenses	368
Saving License Keys (CLI)	369
Verifying Junos OS License Installation (CLI)	369
Displaying Installed Licenses	369
Displaying License Usage	371
License Modes for Enhanced MPCs Overview	372
Configuring the License Mode for Specific Enhanced MPCs on MX Series Routers	373
Example: Configuring the License Mode for MPC5E	374

Chapter 13	Troubleshooting Software Installation	381
	Troubleshooting Software Installation	381
	Recovering from a Failed Software Upgrade on an EX Series Switch	381
	Rebooting from the Inactive Partition	382
	Freeing Disk Space for Software Installation	383
	Installation from the Boot Loader Generates 'cannot open package' Error	383
	Troubleshooting a Switch That Has Booted from the Backup Junos OS Image	384
	Disk Space Management for Junos OS Installation	385
	Verifying PIC Combinations	386
Chapter 14	Configuration Statements	389
	auto-configuration	390
	auto-configuration (System)	391
	auto-image-upgrade	393
	auto-snapshot	394
	autoinstallation	395
	autoinstallation (JNU Satellite Devices)	396
	bootp	397
	commit	398
	configuration-servers	399
	delete-after-commit (JNU Satellites)	400
	file (App Engine Virtual Machine Management Service)	401
	flag (App Engine Virtual Machine Management Service)	403
	interfaces (Autoinstallation)	404
	level (App Engine Virtual Machine Management Service)	405
	license	406
	traceoptions (App Engine Virtual Machine Management Service)	408
	traceoptions (System License)	409
	vmhost	411
	vmhost management-if disable	412
	vmhost management-if link-mode	413
	vmhost management-if speed	414
	usb	415
	vmhost	416
Chapter 15	Operational Commands	417
	clear system login lockout	419
	request system autorecovery state	420
	request system download abort	422
	request system download clear	423
	request system download pause	424
	request system download resume	425
	request system download start	426
	request system firmware upgrade	428
	request system halt	430
	request system license add	432
	request system license delete	434

request system license save	435
request system license update	436
request system partition compact-flash	437
request system power-off	438
request system reboot	440
request system reboot (EX Series)	446
request system reboot (Junos OS with Upgraded FreeBSD)	449
request system recover	452
request system scripts add	454
request system scripts delete	455
request system scripts rollback	456
request system snapshot	457
request system snapshot (Junos OS with Upgraded FreeBSD)	465
request system snapshot (Maintenance)	468
request system software abort in-service-upgrade (ICU)	471
request system software add	472
request system software add (Maintenance)	486
request system software configuration-backup	487
request system software configuration-restore	488
request system software delete	489
request system software download	493
request system software recover-from-restore-point	495
request system software restore-point	496
request system software rollback	498
request system software rollback (SRX Series)	503
request system software validate	504
request system software validate on (Junos OS with Upgraded FreeBSD)	508
request system storage cleanup	511
request system storage cleanup (SRX Series)	522
request system zeroize	525
show chassis usb storage	531
show system autoinstallation status	532
show system autorecovery state	534
show system boot-messages	536
show system auto-snapshot	543
show system download	545
show system license	547
show system license (View)	556
show system login lockout	559
show system rollback	560
show system software restore-point-status	562
show system software usb-software-version	563
show system snapshot	565
show system snapshot (Junos OS with Upgraded FreeBSD)	568
show system snapshot media	570
show system storage partitions	571
show version	574

Chapter 16	VM Host Administration Commands	577
	request vmhost cleanup	578
	request vmhost file-copy	580
	request vmhost halt	581
	request vmhost hard-disk-test	583
	request vmhost power-off	585
	request vmhost power-on	587
	request vmhost reboot	588
	request vmhost snapshot	590
	request vmhost software abort	592
	request vmhost software add	593
	request vmhost software in-service-upgrade	597
	request vmhost software rollback	599
	request vmhost zeroize	602
Chapter 17	VM Host Monitoring Commands	605
	show vmhost bridge	606
	show vmhost crash	608
	show vmhost hard-disk-test	609
	show vmhost hardware	611
	show vmhost information	613
	show vmhost logs	615
	show vmhost management-if	617
	show vmhost netstat	618
	show vmhost processes	620
	show vmhost resource-usage	622
	show vmhost snapshot	625
	show vmhost status	627
	show vmhost uptime	629
	show vmhost version	631

List of Figures

Chapter 1	Software Overview	29
	Figure 1: Configuration Selection Sequence	51
Chapter 2	Installing, Upgrading, and Downgrading Software	55
	Figure 2: Connecting to the Console Port on a Junos OS Device	57
	Figure 3: Upgrading to 64-bit Junos OS with Redundant Routing Engines	64
	Figure 4: Upgrading a Single Routing Engine to 64-bit Junos OS Using Two Slots	67
Chapter 7	Storage Media	203
	Figure 5: Routing Engines	205
	Figure 6: SRX240 Device Front Panel	207
	Figure 7: SRX650 Device System Routing Engine	207
	Figure 8: SRX345 Device Front Panel	207
	Figure 9: SRX1500 Device Front Panel	207
	Figure 10: SRX4200 Services Gateway Front Panel	208
	Figure 11: SRX4600 Services Gateway Front Panel	208
	Figure 12: SRX5800 Device Routing Engine	208
Chapter 9	Zero Touch Provisioning	257
	Figure 13: Components Involved in Initial Provisioning of Remote Device	268
	Figure 14: Workflow for Initial Provisioning	269
	Figure 15: Entering Activation Code for ZTP	271
	Figure 16: Initiating ZTP Process (Software Image Downloading)	271
	Figure 17: Completing ZTP Process	272
	Figure 18: Configuring System Root-Authentication Password	273
Chapter 11	Installation, Upgrade, and Recovery of VM Host Support on Devices with Routing Engines	303
	Figure 19: Architecture of RE-MX-X6, RE-MX-X8 and RE-PTX-X8 Routing Engines	305
	Figure 20: SSD Partitioning	307
	Figure 21: Host partition table for Routing Engines with 200-GB SSDs	308
	Figure 22: Partitioning of the guest VM	308
	Figure 23: Host partition table for Routing Engines on MX2010 and MX2020 routers with 100GB SSD	308
	Figure 24: Guest VM partition on MX2010 and MX2020 Routers	308

List of Tables

	About the Documentation	xxi
	Table 1: Notice Icons	xxiv
	Table 2: Text and Syntax Conventions	xxiv
Chapter 1	Software Overview	29
	Table 3: New and Changed Commands and Statements for Junos OS with Upgraded FreeBSD	38
	Table 4: Deprecated Commands and Statements for Junos OS with Upgraded FreeBSD	38
	Table 5: Installation Package Prefixes	44
	Table 6: Software Release Types	46
	Table 7: show system download Output Fields	53
Chapter 2	Installing, Upgrading, and Downgrading Software	55
	Table 8: Caveats for Downgrading from Junos OS Release 15.1X53-D60 to Previous Software Releases	81
	Table 9: Caveats for Downgrading from Junos OS Release 15.1X53-D60 to Previous Software Releases	83
	Table 10: Upgrade Path to Junos OS with the Upgraded FreeBSD	105
	Table 11: Junos OS and Jloader Software Compatibility Matrix for the QFX3500 Switch and QFX3500 Node Device	125
	Table 12: Junos OS and Jloader Software Compatibility Matrix for the QFX3008-I Interconnect Device	126
	Table 13: Junos OS and Jloader Software Compatibility Matrix for the QFX3600-I Interconnect Device and QFX3600 Node Device	126
	Table 14: Uboot Software Release and Jloader Software Compatibility Matrix	126
Chapter 3	Using the USB Flash Drive and Boot Loader to Upgrade Software	139
	Table 15: Environment Variables Settings	145
Chapter 4	Upgrading the BIOS and Firmware	153
	Table 16: CLI Commands for Manual BIOS Upgrade	155
Chapter 5	Reinstalling Software	165
	Table 17: Checklist for Reinstalling Junos OS	165
Chapter 6	Configuring Dual-Root Partitions on Switches	187
	Table 18: Resilient Dual-Root Partition Scheme	188
	Table 19: Earlier Partition Scheme	190
	Table 20: Storage Media on SRX Series Devices	191
Chapter 7	Storage Media	203

	Table 21: Routing Engines and Storage Media Names (ACX Series, M Series, MX Series, T Series, TX Matrix, TX Matrix Plus, and JCS 1200 Routers)	210
	Table 22: Storage Media Names	212
Chapter 8	Performing a Recovery Installation	215
	Table 23: Autorecovery Alarms	245
Chapter 10	Automatic Installation of Configuration Files	279
	Table 24: Interfaces and Protocols for IP Address Acquisition During Autoinstallation	280
Chapter 11	Installation, Upgrade, and Recovery of VM Host Support on Devices with Routing Engines	303
	Table 25: Hardware Specifications of the RE-MX-X6, RE-MX-X8, RE-PTX-X8, and RCBPTX, Routing Engines	304
Chapter 12	Installing and Managing Software Licenses	325
	Table 26: Junos OS EFL Part Number on EX2200 Switches	328
	Table 27: Junos OS EFL Part Number on EX2300 Switches	329
	Table 28: Junos OS EFL Part Number on EX3300 Switches	330
	Table 29: Junos OS AFL Part Number on EX3300 Switches	331
	Table 30: Junos OS EFL Part Number on EX3400 Switches	331
	Table 31: Junos OS EFL Part Number on EX4300 Switches	332
	Table 32: Junos OS AFL Part Number on EX4300 Switches	333
	Table 33: Junos OS AFL Part Number on EX4600 Switches	333
	Table 34: Junos OS AFL Part Number on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches	334
	Table 35: Junos OS Feature License Model Number for M Series, MX Series, and T Series Routers	338
	Table 36: Junos OS Feature License Model Number for M Series Routers	341
	Table 37: Junos OS Feature License Model Number for MX Series Routers	342
	Table 38: Standard Junos OS Feature Licenses and Model Numbers for QFX Series Devices	350
	Table 39: Disaggregated Junos OS Feature Licenses and Associated SKU's	353
	Table 40: Upgrade Licenses for Enhancing Port Capacity	356
	Table 41: Port Activation License Model for MX104 Routers	357
	Table 42: License Variants for MPCs	372
Chapter 15	Operational Commands	417
	Table 43: request system storage cleanup Output Fields	514
	Table 44: request system storage cleanup Output Fields	522
	Table 45: show system autoinstallation status Output Fields	533
	Table 46: show system autorecovery state Output Fields	534
	Table 47: show system auto-snapshot status Output Fields	543
	Table 48: show system download Output Fields	545
	Table 49: show system license Output Fields	548
	Table 50: show system license Output Fields	556
	Table 51: show system login logout	559
	Table 52: show system software restore-point status Output Fields	562
	Table 53: show system software usb-software-version Output Fields	563
	Table 54: show system snapshot Output Fields	566

Table 55: show system storage partitions Output Fields	572
--	-----

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 <https://www.juniper.net/documentation/>.

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 <https://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 <https://www.juniper.net/documentation/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 <https://www.juniper.net/documentation/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 <https://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <https://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: <https://www.juniper.net/customers/support/>
- Search for known bugs: <https://prsearch.juniper.net/>
- Find product documentation: <https://www.juniper.net/documentation/>
- Find solutions and answer questions using our Knowledge Base: <https://kb.juniper.net/>
- Download the latest versions of software and review release notes: <https://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <https://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum: <https://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <https://www.juniper.net/cm/>

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

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 <https://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 <https://www.juniper.net/support/requesting-support.html>.

CHAPTER 1

Software Overview

- [Junos OS Overview on page 29](#)
- [Junos OS with Upgraded FreeBSD on page 31](#)
- [Junos OS Installation Package Names on page 43](#)
- [FIPS 140-2 Security Compliance on page 47](#)
- [Boot Sequence on Devices with Routing Engines on page 48](#)
- [Configuration Files on page 50](#)
- [Understanding Download Manager for SRX Series Devices on page 52](#)

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 30](#)
- [One Modular Software Architecture on page 30](#)
- [Secure Boot on page 30](#)

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.

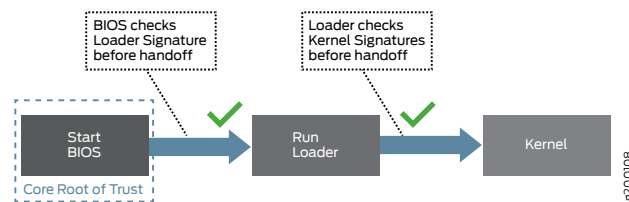
Juniper Networks routing platforms run only binaries supplied by Juniper Networks, and currently do not support third-party binaries. Each Junos OS image includes a digitally signed manifest of executables that are registered with the system only if the signature can be validated. Junos OS will not execute any binary without a registered signature. This feature protects the system against unauthorized software and activity that might compromise the integrity of your device.

Secure Boot

Secure Boot is a significant system security enhancement based on the UEFI standard (see www.uefi.org). It works by safeguarding the BIOS itself from tampering or modification and then maintaining that protection throughout the boot process.

The Secure Boot process begins with Secure Flash, which ensures that unauthorized changes cannot be made to the firmware. Authorized releases of Junos OS carry a digital signature produced by either Juniper Networks directly or one of its authorized partners. At each point of the boot-up process, each component verifies the next link is sound by checking the signature to ensure that the binaries have not been modified. The boot process cannot continue unless the signature is correct. This "chain of trust" continues until the operating system takes control. In this way, overall system security is enhanced, increasing resistance to some firmware-based persistent threats.

Figure 1 shows a simplified version of this "chain of trust."



For information on which Junos OS releases and hardware support Secure Boot, see [Feature Explorer](#) and enter **Secure Boot**.

Related Documentation

- [Junos OS Editions on page 47](#)
- [Junos OS Installation Packages Prefixes on page 44](#)

Junos OS with Upgraded FreeBSD

Junos OS with upgraded FreeBSD provides a clean-slate implementation of Junos OS on top of a pristine (minimally modified) and current version of FreeBSD. Basing Junos OS on an upgraded FreeBSD kernel makes Junos OS more efficient and takes advantage of the sophisticated processing and security features in the upgraded FreeBSD kernel.

Both virtualized Routing Engines (or VM hosts) and non-virtualized Routing Engines run Junos OS with upgraded FreeBSD. For more information on VM hosts, see [“Routing Engines with VM Host Support” on page 303](#) and [“VM Host Installation” on page 312](#).

The major processing changes in Junos OS with upgraded FreeBSD are as follow:

- 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.
- An Upgraded FreeBSD image also includes FIPS mode as a configuration mode, eliminating the need for a separate FIPS package.
- Better scaling numbers in platforms with multiple CPU cores.

- No adverse effect on the routing engine performance on platforms with no additional virtual CPUs (VCPU).
- Storage space for JUNOS root partition is 4 G and 16 G for *var* and *config* partitions, which is same as in the legacy JUNOS image.

Limitations:

The following limitations exist on the upgraded FreeBSD for SRX Series devices:

- Upgraded FreeBSD is supported only on Routing Engine 1800X4 type.
- The underlying FreeBSD is 64 bits, while there are specific 32-bit processes and utilities.
- ISSU is not supported from an older version of FreeBSD to an upgraded FreeBSD. However, it is supported between upgraded FreeBSDs.
- If you downgrade from Junos OS Release 17.4 to any previous releases, all the data in the disk will be lost, and system boots up with default factory configuration.

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). Nonrecovery snapshots (but not recoverable image snapshots) are available for the package sets to preserve package content lists.

For other changes in Junos OS with upgraded FreeBSD, see the following subsections:

- [Release Information for Junos OS with Upgraded FreeBSD on page 32](#)
- [Changes in Package Names for Junos OS with Upgraded FreeBSD on page 33](#)
- [Changes in Commands and Statements in Junos OS with Upgraded FreeBSD on page 37](#)
- [Changes in Disk Volumes for Junos OS with Upgraded FreeBSD on page 39](#)
- [Changes in Use of Snapshots for Junos OS with Upgraded FreeBSD on page 41](#)

Release Information for Junos OS with Upgraded FreeBSD

Junos OS with upgraded FreeBSD was first introduced in Junos OS Release 15.1 running on bare metal.

In Junos OS Release 16.1, with the release of virtualized Routing Engines RE-MX-X6, RE-MX-X8, and RE-PTX-X8, Junos OS with upgraded FreeBSD could run as a guest virtual machine (VM) on a Linux VM host.



NOTE: VM hosts only run virtualized Junos OS with upgraded FreeBSD.

To find which platforms support Junos OS with upgraded FreeBSD, see [Feature Explorer](#) and enter one of the following:

- For non-virtualized, enter **freebsd** and select **Junos kernel upgrade to FreeBSD 10+**.
- For virtualized, enter **virtualization** and select **Virtualization of the Routing Engine**.



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 103](#)
- **Guest VM:** [“VM Host Installation” on page 312](#)

Changes in Package Names for Junos OS with Upgraded FreeBSD

Junos OS with upgraded FreeBSD is based on an upgraded FreeBSD kernel and has been released on a platform-by-platform basis starting in Junos OS Release 15.1. Package-naming conventions changed in certain ways with the release of Junos OS with upgraded FreeBSD, depending on the hardware platform.

Junos OS with upgraded FreeBSD packages use XML description files instead of scripts.

Installation package names for VM hosts begin with the **junos-vmhost-install** prefix.

For information on and examples of other installation package names for Junos OS with upgraded FreeBSD, see the following subsections:

- [Linux-Based Platforms Package Names on page 33](#)
- [EX Series Switches Package Names on page 34](#)
- [MX Series Routers Package Names on page 35](#)
- [QFX Series and EX4600 Switches Package Names on page 36](#)
- [SRX5400, SRX5600, and SRX5800 Devices Package Names on page 37](#)

Linux-Based Platforms Package Names

The following are components of the Junos OS with upgraded FreeBSD package-naming conventions for Linux-based packages such as those for SRX Series, ACX Series, NFX Series, OCX Series, and PTX Series:

- **Prefix**—Linux-based devices use the **jinstall-host** prefix for Junos OS with upgraded FreeBSD.
- **Platform**—This field indicates the major product group, such as **acx**, **nfx**, **ocx**, or **ptx**.
- **Product**—This field indicates the specific product.
- **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 **17.3R1.3**.
- Edition—The edition field is null (empty) for standard (domestic) images. For jurisdictions with limits on dataplane encryption, this field is set to **limited**.

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

- **jinstall-host-acx5k-17.2R1.13-signed.tgz**
- **jinstall-host-nfx-2-flex-x86-32-17.2R1.13-secure-signed.tgz**

EX Series Switches Package Names

There are multiple conventions for naming installation packages for EX Series switches.

- The EX9200 switch is based on the MX Series routers and has the same package-naming convention as the MX Series routers. See [“MX Series Routers Package Names” on page 35](#).
- The EX4600 switch is based on the QFX5100 platform and has the same package-naming convention as the QFX5100 platform. See [“QFX Series and EX4600 Switches Package Names” on page 36](#).
- The components of the Junos OS with Upgraded FreeBSD package-naming conventions for EX2300 and EX3400 switches are as follows:
 - Prefix—This is **junos-arm**. This prefix takes the place of the **jinstall** prefix used in earlier releases of Junos OS.
 - 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. Media keywords follow the term **media** in the package name. Values for the media keyword include the following:
 - usb** for images installed from a USB drive
 - net** for images installed from the loader prompt
 - 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 standard (domestic) images. For jurisdictions with limits on dataplane encryption, this field is set to **limited**.

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 software package names include the following:

- **junos-arm-32-15.1X53-D50.2.tgz**—Image for an EX2300 or EX3400 platform for jurisdictions without limits on dataplane encryption.
- **junos-arm-32-15.1X53-D50.2-limited.tgz** —Image for an EX2300 or EX3400 platform for jurisdictions with limits on dataplane encryption.
- **junos-install-media-usb-arm-32-15.1X53-D50.2.img**—Image stored on and installed from a USB drive for a EX2300 or EX3400 platform for jurisdictions without limits on dataplane encryption.
- **junos-install-media-net-arm-32-15.1X53-D50.2.tgz**—Image stored on the tftp server and installed from a loader prompt for a EX2300 or EX3400 platform for jurisdictions without limits on dataplane encryption.

MX Series Routers Package Names

The components of the Junos OS with Upgraded FreeBSD package-naming conventions for MX Series routers and EX9200 switches are as follows:

- **Prefix**—This is **junos-install**. This prefix takes the place of the **jinstall** prefix used in earlier releases of Junos OS.
- **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. Media keywords follow the term **media** in the package name. Values for the media keyword include the following:
 - usb** for images installed from a USB drive
 - net** for images installed from the loader prompt
- **Platform**—This field indicates the major product group, such as **ex92xx** or **mx**.
- **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 **17.3R1.3**.
- **Edition**—The edition field is null (empty) for standard (domestic) images. For jurisdictions with limits on dataplane encryption, this field is set to **limited**.

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 for jurisdictions without limits on dataplane encryption.
- **junos-install-mx-x86-32-15.1R1.9-limited.tgz**—Image for a supported MX Series platform used for jurisdictions with limits on dataplane encryption.
- **junos-install-media-usb-mx-x86-32-15.1R1.9.tgz**—Image stored on and installed from a USB drive for a supported MX Series platform for jurisdictions without limits on dataplane encryption.
- **junos-install-ex92xx-x86-64-17.2R1.13-limited.tgz**—Image for an EX9200 platform for jurisdictions with limits on dataplane encryption.
- **junos-install-media-usb-ex92xx-17.2R1.13.img.gz**—Image stored on and installed from a USB for an EX9200 platform for jurisdictions without limits on dataplane encryption.

QFX Series and EX4600 Switches Package Names

The components of the Junos OS with upgraded FreeBSD package-naming conventions for QFX Series and EX4600 switches installation packages are as follows:

- Prefix—Linux-based devices use the **jinstall-host** prefix for Junos OS with upgraded FreeBSD.
- Platform—This field indicates the major product group, such as **ex-4600** or **qfx**.
- Product—This field indicates the specific product, such as **5e** or **10-f** or **10-m**.
- 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 **17.3R1.3**.
- Edition—The edition field is null (empty) for standard (domestic) images. For jurisdictions with limits on dataplane encryption, this field is set to **limited**.

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

- **jinstall-host-ex-4600-17.2R1.13-limited-signed.tgz**
- **jinstall-host-ex-4600-17.2R1.13-signed.tgz**
- **jinstall-host-qfx-5e-x86-64-17.2R1.13.tgz**
- **jinstall-host-qfx-10-f-flex-x86-64-17.2R1.13-secure-signed.tgz**
- **jinstall-host-qfx-10-m-x86-64-17.2R1.13-secure-limited-signed.tgz**
- **jinstall-host-qfx-5-17.2R1.13-limited-signed.tgz**

SRX5400, SRX5600, and SRX5800 Devices Package Names

The components of the Junos OS with upgraded FreeBSD package-naming conventions for SRX5400, SRX5600, and SRX5800 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**.



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-srx5000-x86-64-17.3R1.9.tgz**—An image for a SRX5400, SRX5600, and SRX5800 devices.
- **junos-install-media-usb-srx5000-x86-64-17.3R1.9.img.gz**—An image stored on and installed from a USB flash drive for SRX5400, SRX5600, and SRX5800 devices.

Changes in Commands and Statements in Junos OS with Upgraded FreeBSD

There is now a separate Operation, Administration, and Maintenance (OAM) volume (**oam**) distinct from the Junos OS volume (**junos**).

One major change between Junos OS and Junos OS with upgraded FreeBSD is the distinction between recovery snapshots and nonrecovery snapshots.

The upgraded FreeBSD kernel requires changes to several commands and statements and their related parameters. The new and changed actions are summarized in [Table 3 on page 38](#). For details on the changes listed in [Table 3 on page 38](#), see the topics covering the specific command or statement.

For changed actions for VM hosts, see “[VM Host Operations and Management](#)” on [page 322](#).

Table 3: New and Changed Commands and Statements for Junos OS with Upgraded FreeBSD

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

The upgraded FreeBSD kernel also requires that several commands and statements in Junos OS be deprecated in Junos OS with upgraded FreeBSD. The deprecated commands and statements are summarized in [Table 4 on page 38](#). The date of deprecation is the release date for that platform supporting Junos OS with upgraded FreeBSD. To find which platforms support Junos OS with upgraded FreeBSD, see [Feature Explorer](#) and enter one of the following:

- For non-virtualized, enter **freebsd** and select **Junos kernel upgrade to FreeBSD 10+**.
- For virtualized, enter **virtualization** and select **Virtualization of the Routing Engine**.

Table 4: Deprecated Commands and Statements for Junos OS with Upgraded FreeBSD

Deprecated Command or Configuration Statement	Release Deprecated
Deprecated Command	

Table 4: Deprecated Commands and Statements for Junos OS with Upgraded FreeBSD (continued)

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

- See Also**
- [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 465](#)
 - [show system snapshot \(Junos OS with Upgraded FreeBSD\) on page 568](#)
 - [request system reboot \(Junos OS with Upgraded FreeBSD\) on page 449](#)

Changes in Disk Volumes for Junos OS with Upgraded FreeBSD

In computer data storage, a volume or logical drive is a single accessible storage area with a single file system, typically (though not necessarily) resident on a single partition of a hard disk.

Junos OS with upgraded FreeBSD has two volumes: **dev/gpt/junos** (**/junos** for short) and a separate operations, administration, and maintenance (OAM) volume **dev/gpt/oam** (**/oam** for short).

- [/junos Volume on page 40](#)
- [/oam Volume on page 40](#)

/junos Volume

The **/junos** volume is used for running device software and holds configuration information and logs.

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. For more information on non-recovery snapshots, see [“Changes in Use of Snapshots for Junos OS with Upgraded FreeBSD” on page 41](#).

/oam Volume

The compact flash drive is the **/oam** volume. 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. For more information on recovery snapshots, see [“Changes in Use of Snapshots for Junos OS with Upgraded FreeBSD” on page 41](#).

Changes in Use of Snapshots for Junos OS with Upgraded FreeBSD

Snapshots taken with Junos OS with upgraded FreeBSD are not the same as snapshots taken with Junos OS (as in legacy Junos OS). The two are not compatible with each other. A recovery snapshot on a USB taken from a router running Junos OS based on the older FreeBSD kernel is not supported for recovery after the router is upgraded to Junos OS with upgraded FreeBSD.

Junos OS with upgraded FreeBSD has two types of snapshots: recovery snapshots (which are not the same thing as recovery snapshots taken using the older Junos OS) and non-recovery snapshots. Recovery snapshots and non-recovery snapshots have different content, locations, and purposes, so it is important that they are created and maintained properly. We recommend that you generate both a non-recovery and a recovery snapshot after you successfully upgrade to Junos OS with upgraded FreeBSD, and refresh these snapshots periodically.

- [Recovery Snapshots on page 41](#)
- [Non-Recovery Snapshots on page 42](#)

Recovery Snapshots

The major characteristics of recovery snapshots are as follow:

- 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.

A recovery snapshot is automatically taken when, for the first time, you upgrade from a pre-FreeBSD-based Junos OS release to Junos OS with upgraded FreeBSD. Therefore, unless someone manually deletes the recovery snapshot, there should always be a recovery snapshot.

If a device does not have a recovery snapshot, then the only way to recover the device would be to do a media install (network or USB).

Helpful commands for recovery snapshots are:

- **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.
- **show system snapshot**—As of Junos OS Release 17.2, use this command to list the recovery snapshot.
- Previous to Junos OS Release 17.2, use the following shell command to see if a recovery snapshot exists on the device:

```
# oamctl list-snapshots
```

Non-Recovery Snapshots

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

- Non-recovery snapshots reside on the **/junos** volume.
- Non-recovery snapshots 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.
- 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.

Non-recovery snapshots consume little space, except for the **config.tgz** file.

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.

The snapshot script (which is the script that generates output for non-recovery snapshots) does not generate XML output. In such cases, the `<output>` tag is used.

```
user@host> request system snapshot | display xml
<rpc-reply xmlns:junos="http://xml.juniper.net/junos/18.1I0/junos">
  <output>
    NOTICE: Snapshot snap.20180105.165049 created successfully
  </output>
  <cli>
    <banner></banner>
  </cli>
</rpc-reply>
```

This is documented in `<rpc-reply>` in the *Junos XML Management Protocol Developer Guide*.

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.

- See Also**
- [show system snapshot \(Junos OS with Upgraded FreeBSD\) on page 568](#)
 - [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 465](#)

- Related Documentation**
- [Upgrading Junos OS with Upgraded FreeBSD on page 103](#)

Junos OS Installation Package Names

The installation package is used to upgrade or downgrade from one release to another. When installed, the installation package completely reinstalls the software, rebuilds the Junos OS file system, and can erase system logs and other auxiliary information from the previous installation. The installation package does, however, retain the configuration files from the previous installation.

Junos OS installation packages have the following general pattern:

prefix-release-edition-signed.extension

For enhanced automation variants of Junos OS, the installation package looks like this:

prefix-flex-release-edition-signed.extension

The **signed** part of the filename indicates that the 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.

Starting in 2015, the word **signed** appears less frequently after the edition in the filename. But you might still see it in software installation packages. Whether **signed** appears or not, all Junos OS images from Junos OS Release 15.1 on are signed for validation.

Extensions are **tgz**, **gz**, **img**, **iso**, etc.

For more detail on the makeup of the installation package filename, see the following sections:

- [Junos OS Installation Packages Prefixes on page 44](#)
- [Junos OS Release Numbers on page 45](#)
- [Junos OS Editions on page 47](#)

Junos OS Installation Packages Prefixes

The first part of the installation package filename is a combination of a standard prefix and product designation. [Table 5 on page 44](#) 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 5 on page 44](#) also indicates the prefixes used for the different platforms running Junos OS with upgraded FreeBSD.

SRX Series packages use the following naming convention for package prefixes:

junos-product

[Table 5 on page 44](#) lists several examples of installation package prefixes for the SRX Series.

[Table 5 on page 44](#) does not list packages you do not use with the **request system software add** command. For information on packages you load onto a USB drive to install, see Chapter 10 of the *Software Installation and Upgrade Guide*.

Table 5: 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.
jinstall-host-nfx-2*	<p>Junos OS with upgraded FreeBSD for NFX2xx platforms that are Linux based; this prefix indicates the image includes the host and Junos OS.</p> <p>For details and specific examples of package names for Junos OS with upgraded FreeBSD, see “Changes in Package Names for Junos OS with Upgraded FreeBSD” on page 33.</p>
jinstall-host-qfx*	<p>Junos OS with upgraded FreeBSD for QFX platforms that are Linux based; this prefix indicates the image includes the host and Junos OS. For example, jinstall-host-qfx-5-release.tgz is the package name for Junos OS on the QFX5100.</p> <p>For details and specific examples of package names for Junos OS with upgraded FreeBSD, see “Changes in Package Names for Junos OS with Upgraded FreeBSD” on page 33.</p>

Table 5: Installation Package Prefixes (continued)

Installation Package Prefix	Description
jinstall-ocx-flex*	OCX Series switches.
jinstall-ppc*	Junos OS for the ACX Series, MX5, MX10, MX40, MX80, and MX104 routers.
junos-arm	Junos OS with Upgraded FreeBSD for EX2300 and EX3400 switches. For details and specific examples of package names for Junos OS with upgraded FreeBSD, see “Changes in Package Names for Junos OS with Upgraded FreeBSD” on page 33.
junos-install*	Junos OS with upgraded FreeBSD for EX Series switches and MX Series routers that support Junos OS with upgraded FreeBSD. For details and specific examples of package names for Junos OS with upgraded FreeBSD, see “Changes in Package Names for Junos OS with Upgraded FreeBSD” on page 33.
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.
junos-install-media-pxe-srxhe-x86* (PXE)	Junos OS for SRX4600.
junos-install-media-usb-srxhe-x86* (USB)	
junos-srxhe-x86* (CLI)	

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

web page, you download the Junos OS installation package for a particular Junos OS release number.

Given the format of

m.nZb.s

The software release number 17.2R1.13, for example, maps to this format as follows:

- *m* is the main release number of the product, for example, 17.
- *n* is the minor release number of the product, for example, 2.
- *Z* is the type of software release, for example, R for FRS or maintenance release.

For types of software releases, see [Table 6 on page 46](#).

- *b* is the build number of the product, for example, 1, indicating the FRS rather than a maintenance release..
- *s* is the spin number of the product, for example, 13.

Table 6: Software Release Types

Release Type	Description
R	First revenue ship (FRS) or maintenance release software. R1 is FRS. R2 onward are maintenance releases.
F	Feature velocity release. Feature velocity releases are only in Junos OS Release 15.1.
B	Beta release software.
I	Internal release software. These are private software releases for verifying fixes.
S	Service release software, which are released to customers to solve a specific problem—this release will be maintained along with the life span of the underlying release. The service release number is added after the R number, for example, 14.2R3-S4.4. Here S4 represents the 4th service release on top of 14.2R3 and is the 4th respin.
X	<p>Special (eXception) release software. X releases follow a numbering system that differs from the standard Junos OS release numbering.</p> <p>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 https://kb.juniper.net/InfoCenter/index?page=home.</p>



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.

Junos OS Editions

Editions show up in the installation package name after the release number string and before *signed*.

In releases earlier than Junos OS Release 15.1, installation packages came in several major software package categories or editions, such as domestic, worldwide, or Federal Information Processing Standard (FIPS). For those still using packages with names including these terms, here is what they indicate:

- **domestic**—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 router or switch. Later images use a null, or empty, edition field for this category.
- **limited**—Junos OS for all other customers. This edition does not include any high-encryption capabilities for data leaving the router or switch. Sometimes referred to as the *Export* edition, starting in Junos OS Release 15.1R1, this category is renamed to the limited edition.
- **fips**—Junos OS that provides advanced network security for customers who need software tools to configure a network of Juniper Networks routers and switches in a Federal Information Processing Standards (FIPS) 140-2 environment. For more information about Junos-FIPS, see FIPS 140-2 Security Compliance. In later images, FIPS, instead of being a separate edition, is an option you select on installation.

Starting with Junos OS 15.1, a simplified edition scheme was started:

- Junos OS with a null (empty) edition field is the standard image for Junos OS.
- **limited**—Version has no cryptographic support and is intended for countries in the Eurasian Customs Union (EACU). These countries have import restrictions on software containing data-plane encryption.

Related Documentation

- [Changes in Package Names for Junos OS with Upgraded FreeBSD on page 33](#)

FIPS 140-2 Security Compliance

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

Starting in Junos OS Release 15.1, Junos-FIPS is packaged in a domestic image only: a single Junos OS image supports both domestic and FIPS features. Users that have the FIPS credentials and permission to login can flip between a regular Junos image and FIPS image.



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.

Boot Sequence on Devices with Routing Engines

Juniper Networks devices start using the installed Junos OS. Bootable copies of Junos OS are stored in various locations: the internal flash disk, the hard drive, the removable media. The following subsections discuss the order of locations checked for a valid bootable operating system.

- [Boot Order for Devices on page 48](#)
- [Booting from an Alternate Boot Device on page 50](#)

Boot Order for Devices

Information about the boot order for the various devices with Routing Engines is given in this section in alphabetical order of the device families.



NOTE: For information about which Routing Engines are supported by each device, see

https://www.juniper.net/documentation/en_US/release-independent/junos/topics/reference/general/routing-engine-m-mx-t-series-support-by-chassis.html.

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)

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 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)

The M Series and MX Series (except for the MX80 routers and the MX104 routers) 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 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

The 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 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

Booting from an Alternate Boot Device



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

- [Hardware Overview of SRX Series Services Gateways on page 207](#)
- [Storage Media Names for SRX Series Devices on page 212](#)
- [Routing Engine Specifications](#)

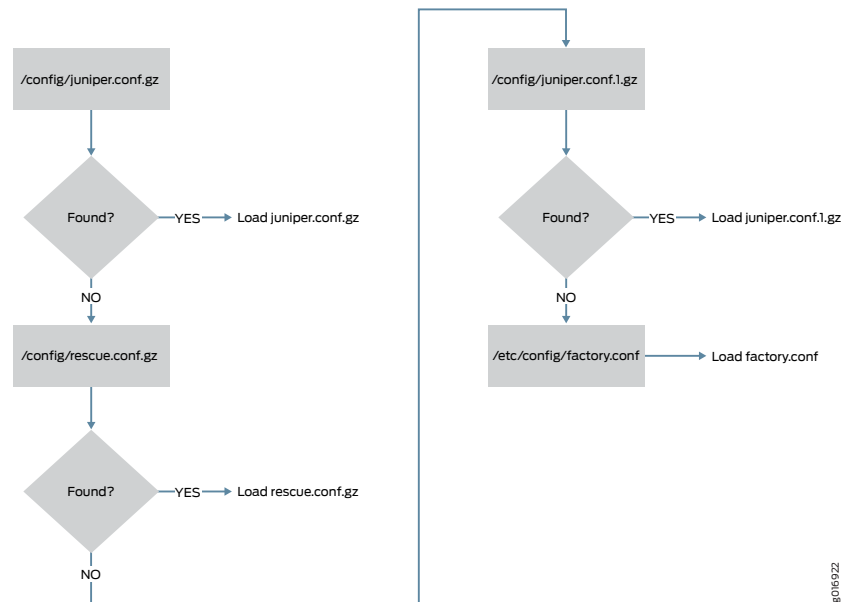
Configuration Files

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 51.

Figure 1: Configuration Selection Sequence



The configuration file selection process occurs in the following sequence:

1. **/config/juniper.conf.gz**—Active configuration file.
2. **/config/rescue.conf.gz**—Rescue configuration file. This file is created by the router or switch administrator.
3. **/config/juniper.conf.1.gz**—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 the **juniper.conf.gz** file. The last three committed configurations are stored in the files **juniper.conf.1.gz** through **juniper.conf.3.gz**. The rescue configuration is stored in the **rescue.conf.gz** file. These files are located in the **/config** directory. The remaining archived configuration files **juniper.conf.4.gz** through **juniper.conf.49.gz** are saved in the **/var/db/config** directory. Configuration files are located in the SRX series compact flash drive.

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

```
user@host>file list /config
/config:
.snap/
db_ext
db_ext-
juniper.conf+.gz
juniper.conf.1.gz
juniper.conf.2.gz
juniper.conf.3.gz
juniper.conf.gz
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 248](#).

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

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 53](#)
- [Handling Errors on page 53](#)
- [Considerations on page 54](#)

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 7 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 7: 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.

Table 7: show system download Output Fields (continued)

Output Field	Description
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 58](#)

CHAPTER 2

Installing, Upgrading, and Downgrading Software

- [Software Installation Overview on page 56](#)
- [Installation Type Overview on page 58](#)
- [Understanding Software Installation on EX Series Switches on page 59](#)
- [Understanding Junos OS Upgrades for SRX Series Devices on page 61](#)
- [Overview of Upgrading to 64-bit Junos OS on page 63](#)
- [Overview of CoS Upgrade Requirements \(Junos OS Release 11.1 or 11.2 to a Later Release\) on page 68](#)
- [Understanding How to Back Up an Installation on Switches on page 70](#)
- [Backing Up the Existing Installation on Routers on page 72](#)
- [Ensuring Sufficient Disk Space for Junos OS Upgrades on SRX Devices on page 74](#)
- [Downloading Software on page 76](#)
- [Installing Software Packages on QFX Devices on page 79](#)
- [Installing the Software Package on a Device with a Single Routing Engine on page 88](#)
- [Installing Software on an EX Series Switch with a Single Routing Engine \(CLI Procedure\) on page 90](#)
- [Installing the Software Package on a Router with Redundant Routing Engines on page 92](#)
- [Installing Software on an EX Series Switch with Redundant Routing Engines \(CLI Procedure\) on page 98](#)
- [Upgrading Junos OS with Upgraded FreeBSD on page 103](#)
- [Downgrading Junos OS from Upgraded FreeBSD on page 109](#)
- [Installing Junos OS Software with Junos Automation Enhancements on page 112](#)
- [Upgrading Software by Using Automatic Software Download on page 117](#)
- [Upgrading the Loader Software on the Line Cards in a Standalone EX8200 Switch or an EX8200 Virtual Chassis on page 117](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 120](#)
- [Reverting the Junos OS Software Image Back to the Previous Version on page 123](#)
- [Upgrading Jloader Software on QFX Series Devices on page 125](#)

Software Installation Overview

A device is delivered with the Juniper Networks operating system (Junos OS) preinstalled. As new features and software fixes become available, you can upgrade your software to use them.

When you power on the device, it starts (boots) using the installed software. 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 edition installed, only the release number of the software.

You upgrade (or downgrade) the version of Junos OS on a device by copying a software installation package to your device or other system on your local network and then using the CLI to install the new software on the device. You then reboot the device, which boots from the upgraded software.

To prepare for installation, back up the system, select the install package you require, and download it from the Juniper Networks download page.

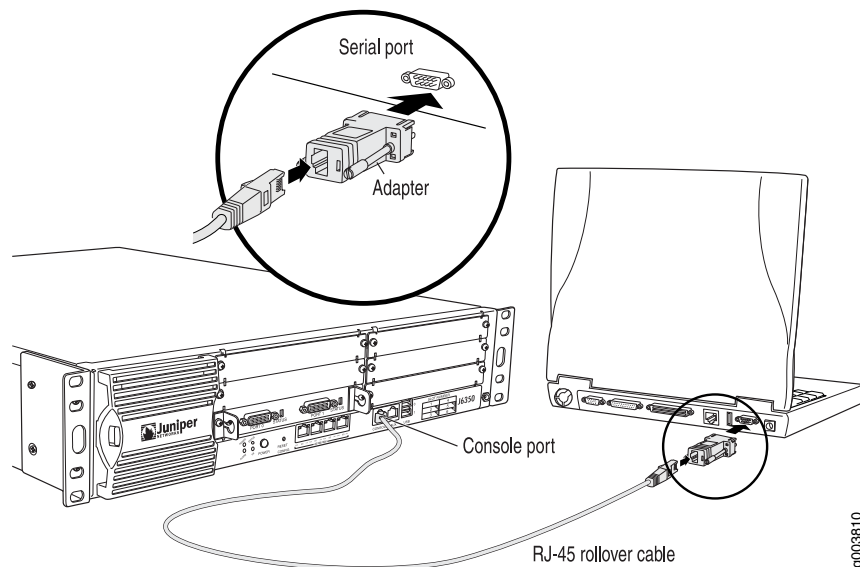
We recommend that you upgrade all individual software packages using an out-of-band connection from the console or management Ethernet interface, because in-band connections can be lost during the upgrade process.

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 2 on page 57](#).

Figure 2: Connecting to the Console Port on a Junos OS Device



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.

During a successful upgrade, the installation package removes all files from the **/var/tmp** directory of the device 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 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.

If you encounter any difficulties during software installation or an upgrade, you can use the recovery installation procedure to install the Junos OS on the device.

After a successful upgrade, you should back up the new existing configuration to a secondary device.

Related Documentation

- [Recovering from a Failed Software Installation on page 229](#)
- [Initial Configuration Overview for Junos OS Devices](#)
- [Accessing a Junos OS Device the First Time](#)

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 44](#).

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 44](#)
- [Software Naming Convention for SRX Series Devices](#)

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 59](#)
- [Software Package Security on page 59](#)
- [Installing Software on a Virtual Chassis on page 60](#)
- [Installing Software on Switches with Redundant Routing Engines on page 60](#)
- [Installing Software Using Automatic Software Download on page 60](#)
- [Autoinstalling a Configuration File on an EX2200 or EX3300 Switch from a Disk-on-Key USB Memory Stick on page 61](#)
- [Installing Software on an EX2300 or EX3400 Switch on page 61](#)
- [Configuration Image Validation on EX Series Switches on page 61](#)
- [Troubleshooting Software Installation on page 61](#)

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.

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 a 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 or on different members. 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*.

- 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.

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*.

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 \(CLI Procedure\)](#)” on page 90 to complete the software installation.

Configuration Image Validation on EX Series Switches

If you upgrade or downgrade the Junos OS image on a switch that supports configuration image validation, the system validates that the existing configuration is compatible with the new image before the actual upgrade or downgrade commences. See *Validating the Configuration Image Before Upgrading or Downgrading the Software* for details about using image validation. See [Feature Explorer](#) for information about which EX Series switches support image validation.

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 381.



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

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*
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 120](#)
- *Installing Junos OS Upgrade Packages on SRX Series Devices from a Remote Server*

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.

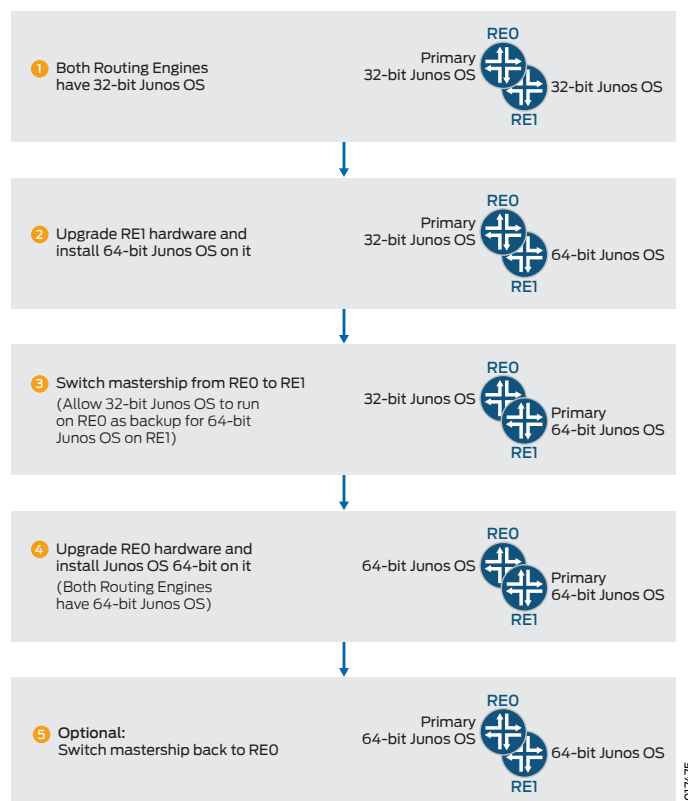
The following upgrade scenarios are covered in this overview:

- [Upgrading Redundant Routing Engines from 32-bit to 64-bit Junos OS on page 64](#)
- [Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using One Slot on page 66](#)
- [Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using Two Slots on page 66](#)

Upgrading Redundant Routing Engines from 32-bit to 64-bit Junos OS

For a diagram of this procedure, see [Figure 3 on page 64](#). For the purposes of this procedure, slot 0 has the primary Routing Engine initially.

Figure 3: 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
```

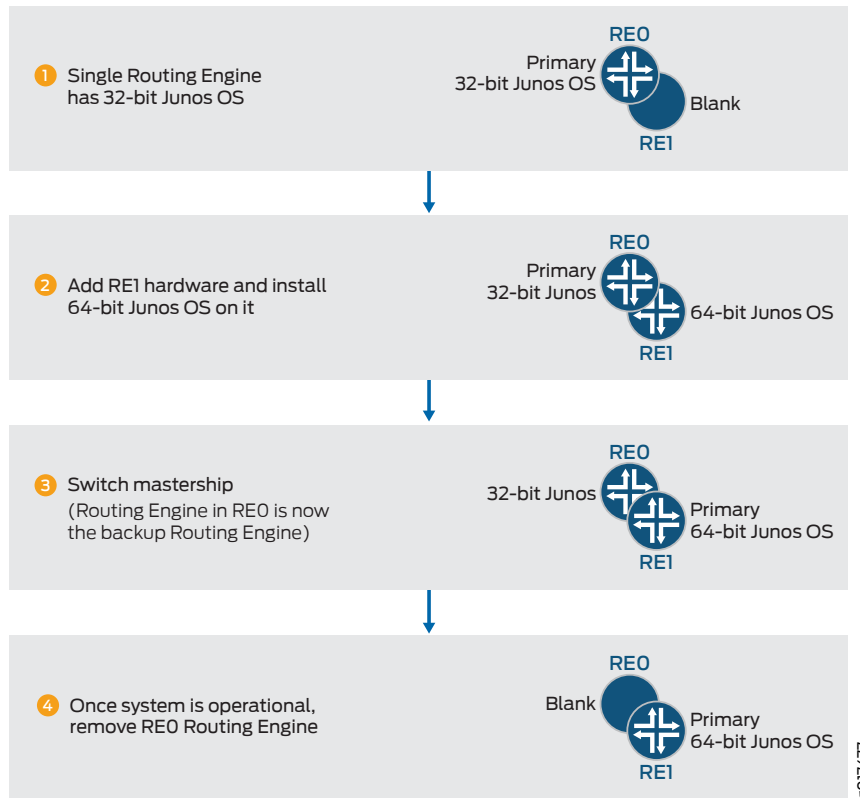
Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using Two Slots

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 66.

For a diagram of this procedure, see [Figure 4 on page 67](#). In this procedure, slot 1 is initially empty.

Figure 4: 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 92](#)
- [Installing the Software Package on a Router with a Single Routing Engine on page 88](#)

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 on page 79](#)
- *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)*

Understanding How to Back Up an Installation on Switches

You can create copies of the software running on a switch using the system snapshot feature. The system snapshot feature takes a “snapshot” of the files currently used to run the switch—the complete contents of the `/config` and `/var` directories, which include the running Junos OS, the active configuration, and the rescue configuration—and copies all of these files into an alternate (internal, meaning internal flash, or an external, meaning USB flash) memory source. You can then use this snapshot to boot the switch at the next boot up or as a backup boot option.

- [Understanding System Snapshot on QFX Switches on page 70](#)
- [Understanding System Snapshot on EX Series Switches on page 70](#)

Understanding System Snapshot on QFX Switches



NOTE: On QFX3500 and QFX3600 switches running Enhanced Layer 2 Software, all of the directories that reside in the “/” partition are read only.



NOTE: System snapshot is not supported on QFX10000 switches.

You can only use snapshots to move files to external memory if the switch was booted from internal memory, or to move files to internal memory if the switch was booted from external memory. You cannot create a snapshot in the memory source that booted the switch even if the snapshot is being created on a different partition in the same memory source.

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.

System snapshots on the switch have the following limitations:

- You cannot use snapshots to move files to any destination outside of the switch other than an installed external USB flash drive.
- Snapshot commands are always executed on a local switch.

Understanding System Snapshot on EX Series Switches

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.

Backing Up the Existing Installation on Routers

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, that is, all stored files except the **juniper.conf** and SSH files are removed. Creating a backup has the following advantages:

- The device can boot from a 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 to the secondary storage device.

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 103](#). For information on which platforms use Junos OS with upgraded FreeBSD, see [“Release Information for Junos OS with Upgraded FreeBSD” on page 32](#).

For information about backing up the existing installation on SRX Series Services Gateways, see *Backing Up the Existing Installation on SRX Series Devices*.

After a successful upgrade, remember to back up the new current configuration to the secondary device.

On routers, you should back up the existing 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 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.

On ACX Series routers, when you issue the **request system snapshot slice alternate** command, the command backs up the files to the router's NAND flash.

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

```
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 the files to an external USB storage device, you need to run the following command:

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

On ACX5000 line of routers, when you issue the **request system snapshot slice alternate** command, the command backs up the files to an external USB storage device.



NOTE: ACX5000 line of routers do not have a NAND flash memory and alternate slice for Junos OS. Junos OS for ACX5000 line of routers runs as a VM on a host image.

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

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

If an external USB is not inserted to the router, then the command shows the following error:

```
user@host> request system snapshot slice alternate
fpc0:
```

```
error: usb (/dev/da0) media missing or invalid
-----
```

Related Documentation

- [request system snapshot on page 457](#)
- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)
- [Upgrading Junos OS with Upgraded FreeBSD on page 103](#)
- [Backing Up the Existing Installation on SRX Series Devices](#)

Ensuring Sufficient Disk Space for Junos OS Upgrades on SRX Devices

Before you begin upgrading Junos OS on an SRX Series device, perform the following tasks:

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-b1ocks	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.

Downloading Software

You can download the software in one of two ways:

- [Downloading Software by Using a Browser on page 76](#)
- [Installing Software Using the Command-Line Interface on page 77](#)

Downloading Software by Using a Browser

You download the software package you need from the Juniper Networks Support website at <https://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://userregistration.juniper.net/entitlement/setupAccountInfo.do>.

To download the software image:

1. From your browser, go to <https://www.juniper.net/support/>.

The Online Support (CSC) page opens.

2. Click the **Download Software** link.

The Download Software page appears.

3. Select the software package that you want to download. You can select software that supports a specific platform or technology.
4. On the page that appears, click the **Software** tab and select the Junos OS installation package to download.
5. Log in with your username and password.
6. On the Download Software page that appears, the following options are available:
 - If you want to download the software on your local host, click the **CLICK HERE** link and save the file to your system. If you want to place the file on a remote system, you must make sure that the file can be accessible by the router, switch, or services gateway by using HTTP, FTP, or SCP. Proceed with the installation. See [“Installing Software Using the Command-Line Interface” on page 77](#) for more details.
 - If you want to download the software on your device, use the following procedure to download and install the software on the device.
 - a. Click **Copy** to copy the generated URL to the clipboard.



NOTE: The URL string generated remains active only for 15 minutes.

- b. Log in to your device.

- c. In operational mode, enter the **file copy "URL" destination** command.

In the command, paste the copied URL string (for *URL*) and then enter **/var/tmp** (as the destination on your hard disk).

Example:



NOTE: Ensure that the URL string is enclosed within quotation marks. Also ensure that there is sufficient free space available on the device.

The software image is downloaded on your device.

- d. (Optional) Validate the software image by using the **request system software validate package-name** command.

Example:

```
user@host> request system software validate /var/tmp/
junos-install-mx-x86-32-17.3R1.10.tgz
```

For more details, see [request system software validate](#).

- e. Install the software by using the **request system software add package-name** command.

Example:

```
user@host> request system software add /var/tmp/
junos-install-mx-x86-32-17.3R1.10.tgz
```

Your software is installed on the device.

See Also • [Junos Platforms - Download Software](#)

Installing Software Using the Command-Line Interface

Download the software package you need from the Juniper Networks Support website at <https://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://userregistration.juniper.net/entitlement/setupAccountInfo.do>.

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 by 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
```

After your credentials are validated, the FTP session opens.

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

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

Following is an example of an *installation-package* name:

junos-install-mx-x86-32-17.3R1.10.tgz

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

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

To transfer the package by 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
```

After your credentials are validated, the FTP session opens.

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

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

Following is an example of an *installation-package* name:

junos-install-mx-x86-32-17.3R1.10.tgz

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

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

See Also • [Junos Platforms - Download Software](#)

Installing Software Packages on QFX Devices

We recommend that you connect to the console port while installing the installation package so you can respond to any required user input and detect any errors that may occur.

Before you install the new installation package, back up your current configuration files because the upgrade process removes all of the stored files on the switch.

To back up your current configuration files, enter the **save** command:

```
user@switch# save filename
```

Executing this command saves a copy of your configuration files to a remote location such as an external USB device.

Installation procedures are in the following subsections:

- [Installing a Standard Software Package on QFX5100, QFX5110, QFX5200, QFX5210, and EX4600 Switches on page 79](#)
- [Installing a Standard Software Package on QFX10002 Switches on page 80](#)
- [Upgrading Software from Junos OS Release 15.1X53-D3X to Junos OS Release 15.1X53-D60, 15.1X53-D61.7, 15.1X53-D62, and 15.1X53-D63 on QFX10008 and QFX10016 Switches on page 82](#)
- [Installing a Software Package on QFX10008 and QFX10016 Switches on page 84](#)

Installing a Standard Software Package on QFX5100, QFX5110, QFX5200, QFX5210, and EX4600 Switches



NOTE: Before you install the software, back up any critical files in `/var/home`. For more information regarding how to back up critical files, contact Customer Support at <https://www.juniper.net/support>.



NOTE: On QFX5100 and EX4600 switches, the Host OS is not upgraded automatically, so you must use the **force-host** option if you want the Junos OS and Host OS versions to be the same.

However, pay attention to these notes regarding Junos OS and Host OS versions:

- The Junos OS and Host OS versions do not need to be the same.
- During an ISSU, the Host OS cannot be upgraded.
- Upgrading the Host OS is not required for every software upgrade, as noted above.



NOTE: On QFX5100 and EX4600 switches, you must use the **force-host** option if you are downgrading from Junos OS Release 14.1X53-D40 to any release earlier than 14.1X53-D40 otherwise the switch will issue core dumps.

If the installation package resides locally on the switch, execute the **request system software add <pathname> <source> reboot** command.

For example:

```
user@switch> request system software add /var/tmp/jinstall-host-qfx-5-17.2R1.n-signed.tgz
reboot
```

If the Install Package resides remotely, execute the **request system software add <pathname> <source> reboot** command.

For example:

```
user@switch> request system software add
ftp://ftpsrvr/directory/jinstall-host-qfx-5-17.2R1.n-signed.tgz reboot
```

After the reboot has finished, verify that the new version of software has been properly installed by executing the **show version** command.

```
user@switch> show version
```

Installing a Standard Software Package on QFX10002 Switches



NOTE: Before you install the software, back up any critical files in **/var/home**. For more information regarding how to back up critical files, contact Customer Support at <https://www.juniper.net/support>.



NOTE: If you want to downgrade from Junos OS Release 15.1X53-D60 to a previous release, pay attention to these caveats:

Table 8: Caveats for Downgrading from Junos OS Release 15.1X53-D60 to Previous Software Releases

Junos OS Software Releases	Using the CLI	Using a USB Stick
15.1X53-D33	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded.	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded.
15.1X53-D32	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded. NOTE: You must downgrade to Junos OS Release 15.1X53-D33 before you downgrade to Junos OS Release 15.1X53-D30.	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded. NOTE: You must downgrade to Junos OS Release 15.1X53-D33 before you downgrade to Junos OS Release 15.1X53-D32.
15.1X53-D30	No	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded. NOTE: You must downgrade to Junos OS Release 15.1X53-D33 before you downgrade to Junos OS Release 15.1X53-D30.
Releases prior to 15.1X53-D30	No	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded. NOTE: You must downgrade to Junos OS Release 15.1X53-D33 before you downgrade to Junos OS Release 15.1X53-D30.

Install the software in one of two ways:

- If the installation package resides locally on the switch, execute the **request system software add <pathname><source> reboot** command.

For example:

```
user@switch> request system software add  
/var/tmp/jinstall-host-qfx-10-17.2R1.n-secure-signed.tgz reboot
```

- If the Install Package resides remotely, execute the **request system software add <pathname><source> reboot** command.

For example:

```
user@switch> request system software add  
ftp://ftpserver/directory/jinstall-host-qfx-10-17.2R1.n-secure-signed.tgz reboot
```

After the reboot has finished, verify that the new version of software has been properly installed by executing the **show version** command.

```
user@switch> show version
```

Upgrading Software from Junos OS Release 15.1X53-D3X to Junos OS Release 15.1X53-D60, 15.1X53-D61.7, 15.1X53-D62, and 15.1X53-D63 on QFX10008 and QFX10016 Switches



NOTE: Before you install the software, back up any critical files in `/var/home`. For more information regarding how to back up critical files, contact Customer Support at <https://www.juniper.net/support>.



NOTE: If you want to downgrade from Junos OS Release 15.1X53-D60 to a previous release, pay attention to these caveats:

Table 9: Caveats for Downgrading from Junos OS Release 15.1X53-D60 to Previous Software Releases

Junos OS Software Releases	Using the CLI	Using a USB Stick
15.1X53-D33	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded.	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded.
15.1X53-D32	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded. NOTE: You must downgrade to Junos OS Release 15.1X53-D33 before you downgrade to Junos OS Release 15.1X53-D30.	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded. NOTE: You must downgrade to Junos OS Release 15.1X53-D33 before you downgrade to Junos OS Release 15.1X53-D32.
15.1X53-D30	No	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded. NOTE: You must downgrade to Junos OS Release 15.1X53-D33 before you downgrade to Junos OS Release 15.1X53-D30.
Releases prior to 15.1X53-D30	No	Yes, but the configuration is lost, and there is no workaround. We recommend that you save the configuration to an external location, such as a remote server, and then load the configuration after you have successfully downgraded. NOTE: You must downgrade to Junos OS Release 15.1X53-D33 before you downgrade to Junos OS Release 15.1X53-D30.

The switch contains two routing engines, so you will need to install the software on each routing engine (re0 and re1).

If the installation package resides locally on the switch, execute the **request system software add <pathname><source>** command.

To install the software on re0:

```
user@switch> request system software add
/var/tmp/jinstall-host-qfx-10-m-15.1X53-D60.4-secure-domestic-signed.tgz re0
```

If the Install Package resides remotely from the switch, execute the **request system software add <pathname><source> re0** command.

For example:

```
user@switch> request system software add
ftp://ftpserver/directory/jinstall-host-qfx-10-m-15.1X53-D60.4-secure-domestic-signed.tgz re0
```

To install the software on re1:

```
user@switch> request system software add
/var/tmp/jinstall-host-qfx-10-m-15.1X53-D60.4-secure-domestic-signed.tgz re1
```

If the Install Package resides remotely from the switch, execute the **request system software add <pathname><source> re1** command.

For example:

```
user@switch> request system software add
ftp://ftpserver/directory/jinstall-host-qfx-10-m-15.1X53-D60.4-secure-domestic-signed.tgz re1
```

Reboot both routing engines.

For example:

```
user@switch> request system reboot both-routing-engines
```

After the reboot has finished, verify that the new version of software has been properly installed by executing the **show version** command.

```
user@switch> show version
```

Installing a Software Package on QFX10008 and QFX10016 Switches

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



NOTE: Before you install the software, back up any critical files in `/var/home`. For more information regarding how to back up critical files, contact Customer Support at <https://www.juniper.net/support>.



WARNING: If graceful Routing Engine switchover (GRES), nonstop bridging (NSB), 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.

To upgrade the software, perform the following tasks:

1. [Preparing the Switch for Installation on page 85](#)
2. [Installing Software on the Backup Routing Engine on page 85](#)
3. [Installing Software on the Master Routing Engine on page 86](#)

Preparing the Switch for Installation

Perform the following steps before installing the software:

1. Log in to the master 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 switch.
2. From the command line, enter configuration mode:

`user@switch> configure`
3. Disable Routing Engine redundancy:

`user@switch# delete chassis redundancy`
4. Disable nonstop-bridging:

`user@switch# delete protocols layer2-control nonstop-bridging`
5. Save the configuration change on both Routing Engines:

`user@switch# commit synchronize`
6. Exit the CLI configuration mode:

`user@switch# exit`

Installing Software on the Backup Routing Engine

After the switch 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 master Routing Engine. This enables the master Routing Engine to continue operations, minimizing disruption to your network.

After making sure that the new software version is running correctly on the backup Routing Engine, you are ready to switch routing control to the backup Routing Engine, and then upgrade or downgrade the software version on the other Routing Engine.

1. Log in to the console port on the other Routing Engine (currently the backup).

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

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

```
user@switch> request system software add validate  
/var/tmp/jinstall-host-qfx-10-17.2R1.n-secure-signed.tgz
```

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

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

```
user@switch> request system reboot
```



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

To abort the installation, do not reboot your switch. Instead, finish the installation and then issue the **request system software delete <package-name>** command. This is your last chance to stop the installation.

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

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

4. Log in and issue the **show version** command to verify the version of the software installed.

```
user@switch> show version
```

Installing Software on the Master 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 master Routing Engine software:

1. Log in to the master Routing Engine console port.

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

2. Transfer routing control to the backup Routing Engine:

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

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

3. Verify that the backup Routing Engine (slot 1) is the master Routing Engine:

```
user@switch> 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)
```

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

```
user@switch> request system software add validate
/var/tmp/jinstall-host-qfx-10-17.2R1.n-secure-signed.tgz
```

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

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

```
user@switch> request system reboot
```



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

To abort the installation, do not reboot your system. Instead, finish the installation and then issue the **request system software delete jinstall <package-name>** 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 switch then reboots from the boot device on which the software was just installed. When the reboot is complete, the switch displays the login prompt.

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

6. Log in and issue the **show version** command to verify the version of the software installed.
7. Transfer routing control back to the master Routing Engine:

```
user@switch> 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 master Routing Engine (slot 0) is indeed the master Routing Engine:

```
user@switch> 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)
```

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

To upgrade the router or switch software:

1. Download the software packages you need from the Juniper Networks Support Web site at <https://www.juniper.net/support/>. For information about downloading software packages, see “[Downloading Software](#)” on page 76.
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
```

3. 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](#).

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.

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](#).

4. 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.

5. Log in and issue the **show version** command to verify the version of the software installed.
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.



NOTE: On an ACX router, you must issue the **request system snapshot slice alternate** command.

Related Documentation

- [Repartitioning Routing Engine System Storage to Increase the Swap Partition on page 212](#)

Installing Software on an EX Series Switch with a Single Routing Engine (CLI Procedure)

You can use this procedure to upgrade Junos OS on a single routing engine in any EX Series switch, including all switches that do not support redundant Routing Engines. You can also use this procedure to upgrade software on all EX Series Virtual Chassis, with the exception of the EX8200 Virtual Chassis.

This procedure can be used to upgrade the following switches or Virtual Chassis:

- EX2200 switch
- EX2300 switch
- EX3200 switch
- EX3300 switch
- EX3400 switch
- EX4200 switch
- EX4300 switch
- EX4500 switch
- EX4550 switch
- EX6200 switch (single Routing Engine upgrade only)
- EX8200 switch (single Routing Engine upgrade only)
- All Virtual Chassis except EX8200 Virtual Chassis

To upgrade software on an EX6200 or EX8200 switch running two Routing Engines, see [“Installing Software on an EX Series Switch with Redundant Routing Engines \(CLI Procedure\)” on page 98](#) or *Upgrading Software on an EX6200 or EX8200 Standalone Switch Using Nonstop Software Upgrade (CLI Procedure)*.

To upgrade software on an EX8200 Virtual Chassis, see *Installing Software for All Devices in an EX8200 Virtual Chassis*.

To install software upgrades on a switch with a single Routing Engine:

1. Download the software package.
2. (Optional) Back up the current software configuration to a second storage option. See the [Junos OS Installation and Upgrade Guide](#) for instructions on performing this task.
3. (Optional) Copy the software package to the switch. We recommend that you use FTP to copy the file to the `/var/tmp` directory.

This step is optional because Junos OS can also be upgraded when the software image is stored at a remote location. These instructions describe the software upgrade process for both scenarios.

4. Install the new package on the switch:

```
user@switch> request system software add package
```

Replace *package* with one of the following paths:

- For a software package in a local directory on the switch—`/var/tmp/package.tgz`.
- For a software package on a remote server:
 - `ftp://hostname/pathname/package.tgz`
 - `http://hostname/pathname/package.tgz`

where *package.tgz* is, for example, `jinstall-ex-4200-9.4R1.8-domestic-signed.tgz`.

Include the optional **member** option to install the software package on only one member of an EX4200 Virtual Chassis:

```
user@switch> request system software add source member member-id reboot
```

Other members of the Virtual Chassis are not affected. To install the software on all members of the Virtual Chassis, do not include the **member** option.



NOTE: To abort the installation, do not reboot your device; instead, finish the installation and then issue the `request system software delete package.tgz` command, where *package.tgz* is, for example, `jinstall-ex-4200-10.2R1.8-domestic-signed.tgz`. This is your last chance to stop the installation.

The `request system software delete package.tgz` command is not available on EX2300 and EX3400 switches.

5. Reboot to start the new software:

```
user@switch> request system reboot (EX Series)
```

6. After the reboot has completed, log in and verify that the new version of the software is properly installed:

```
user@switch> show version
```

7. To ensure that the resilient dual-root partitions feature operates correctly, execute the following command to copy the new Junos OS image into the alternate root partition:

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

To update the alternate root partitions on all members of a Virtual Chassis, use this command:

```
user@switch> request system snapshot slice alternate all-members
```

Resilient dual-root partitions allow the switch to boot transparently from the alternate root partition if the system fails to boot from the primary root partition.



NOTE: EX2300 and EX3400 switches have two volumes: JUNOS volume and OAM (recovery) volume. To store a snapshot (non-recovery) on JUNOS volume, use the command `request system snapshot`. To create snapshot (recovery) on the OAM volume, use the command `request system snapshot recovery`.

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 deactivate GRES (if it is enabled). 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 92](#)
2. [Installing Software on the Backup Routing Engine on page 93](#)
3. [Installing Software on the Remaining Routing Engine on page 95](#)
4. [Finalizing the Installation on page 96](#)

Preparing the Router for the Installation

To prepare the router for software installation:

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:

```
{master}  
user@host> configure  
Entering configuration mode  
  
{master} [edit]  
user@host#
```

3. Disable nonstop-bridging if it is enabled:

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

4. Disable Routing Engine redundancy if enabled:

```
{master}[edit]
user@host# (delete | deactivate) chassis redundancy graceful-switchover
```

5. Save the configuration change on both Routing Engines:

```
{master}[edit]
user@host# commit synchronize
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete
```

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.

Before you start this procedure, decide which software package you need and download it to the `/var/tmp` directory of the primary Routing Engine. For information on which packages to use for which upgrades, see [“Junos OS Installation Package Names” on page 43](#).

To install software on the backup Routing Engine:

1. Log in to the console port on the current primary Routing Engine in slot 0.
2. Install the new software package on the backup Routing Engine (re1) using the **request system software add** command:

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

Installation and validation take about 15 minutes.



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 backup Routing Engine to start the new software:

```
user@host> request system reboot other-routing-engine
Rebooting re1
user@host>
```

You must reboot the device to load the new installation of Junos OS on the router. You can combine steps 2 and 3 by adding **reboot** to the **request system software add** command. But if you do the steps separately, make sure you reboot the Routing Engine you just added system software to.



NOTE: To abort the installation, do not reboot your device. Instead, finish the installation and then issue the **request system software delete software-package-name** 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. Issue the **show version invoke-on other-routing-engine** command to verify the new software is installed.

```
user@host> show version invoke-on other-routing-engine
re1:
-----
Hostname: host1
Model: mx240
Junos: package-name
. . .
user@host>
```

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 76](#).

The **jweb** installation module adds a router management graphical user interface that you can use to view and configure your router.

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.

To install software on the primary Routing Engine:

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

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

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

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
```

Installation and validation take about 15 minutes.



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
```

You must reboot the device to load the new installation of Junos OS on the router. You can combine steps 3 and 4 by adding **reboot** to the **request system software add** command. But if you do the steps separately, make sure you reboot the Routing Engine you just added system software to.



NOTE: To abort the installation, do not reboot your device. Instead, finish the installation and then issue the **request system software delete software-package-name** 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 backup Routing Engine (slot 0) and issue the **show version** command to verify the version of the software installed.

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

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 76](#).

The **jweb** installation module adds a router management graphical user interface that you can use to view and configure your router.

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.

To finalize the redundant Routing Engines upgrade:

1. Restore the configuration that existed before you started this procedure (from [“Preparing the Router for the Installation” on page 92](#)):

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

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

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

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

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

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

```
{master}
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)
```

5. 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, use the **request system snapshot recovery** command. To find which platforms in which releases use Junos OS with upgraded FreeBSD, see [Feature Explorer](#) and enter **Junos kernel upgrade to FreeBSD 10+**. For more information, see “Changes in Use of Snapshots for Junos OS with Upgraded FreeBSD” on page 41.
- For Junos OS, use the **request system snapshot** command:

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

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 system snapshot** 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 212](#)

Installing Software on an EX Series Switch with Redundant Routing Engines (CLI Procedure)

For an EX6200 switch or an EX8200 switch with redundant Routing Engines, you can minimize disruption to network operation during a Junos OS upgrade by upgrading the Routing Engines separately, starting with the backup Routing Engine.



NOTE: If your EX8200 switch is running Junos OS Release 10.4R3 or later, you can upgrade the software packages on both Routing Engines with a single command and with minimal network disruption by using nonstop software upgrade (NSSU) instead of this procedure. See *Upgrading Software on an EX6200 or EX8200 Standalone Switch Using Nonstop Software Upgrade (CLI Procedure)*.



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 disable GRES before you begin the software installation by using the `deactivate chassis redundancy graceful-switchover` command in configuration mode. 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.

To upgrade the software package on an EX6200 switch or an EX8200 switch with one installed Routing Engine, see [“Installing Software on an EX Series Switch with a Single Routing Engine \(CLI Procedure\)” on page 90](#).

To upgrade redundant Routing Engines, you first install the new Junos OS release on the backup Routing Engine while keeping the currently running software version on the master Routing Engine. After making sure that the new software version is running correctly on the backup Routing Engine, you switch device control to the backup Routing Engine. Finally, you install the new software on the new backup Routing Engine.

To upgrade Junos OS on the switch, perform the following tasks:

1. [Preparing the Switch for the Software Installation on page 99](#)
2. [Installing Software on the Backup Routing Engine on page 100](#)
3. [Installing Software on the Default Master Routing Engine on page 101](#)
4. [Returning Routing Control to the Default Master Routing Engine \(Optional\) on page 102](#)

Preparing the Switch for the Software Installation

Perform the following steps before installing the software:

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

For information on logging in to the Routing Engine through the console port, see *Connecting and Configuring an EX Series Switch (CLI Procedure)*.

2. Enter the Junos OS CLI configuration mode:

- a. Start the CLI from the shell prompt:

```
user@switch:RE% cli
```

You will see:

```
{master}
user@switch>
```

- b. Enter configuration mode:

```
user@switch> configure
```

You will see:

```
{master}[edit]
user@switch#
```

3. Disable nonstop active routing (NSR) (supported on switches running Junos OS Release 10.4 or later):

```
{master}[edit]
user@switch# delete routing-options nonstop-routing
```

4. Disable nonstop bridging:

```
{master}[edit]
user@switch# delete nonstop-bridging nonstop-bridging
```

5. Disable graceful Routing Engine switchover (GRES):

```
{master}[edit]
user@switch# deactivate chassis redundancy graceful-switchover
```

6. Save the configuration change on both Routing Engines:

```
{master}[edit]
```

```
user@switch# commit synchronize
```



NOTE: To ensure the most recent configuration changes are committed before the software upgrade, perform this step even if nonstop active routing and graceful Routing Engine switchover were previously disabled.

7. Exit the CLI configuration mode:

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

8. (Optional) Back up the current software configuration to a second storage option.
See the [Junos OS Installation and Upgrade Guide](#) for instructions on performing this task.

Installing Software on the Backup Routing Engine

After you have prepared the switch for software installation, install the software on the backup Routing Engine. During the installation, the master Routing Engine continues operations, minimizing the disruption to network traffic.

1. Download the software.
2. Copy the software package to the switch. We recommend that you use FTP to copy the file to the `/var/tmp` directory.
3. Log in to the console of the backup Routing Engine.
4. Install the new software package:

```
user@switch> request system software add /var/tmp/package.tgz
```

where *package.tgz* is, for example, `jinstall-ex-8200-10.2R1.8-domestic-signed.tgz`.



NOTE: To abort the installation, do not reboot your device; instead, finish the installation and then issue the `request system software delete package.tgz` command, where *package.tgz* is, for example, `jinstall-ex-8200-10.2R1.8-domestic-signed.tgz`. This is your last chance to stop the installation.

5. Reboot to start the new software:

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



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

6. After the reboot has completed, log in and verify the new version of the software is properly installed:

```
user@switch> show version
```

Installing Software on the Default Master Routing Engine

To transfer control to the backup Routing Engine and then upgrade or downgrade the master Routing Engine software:

1. Log in to the master Routing Engine console port.
2. Transfer control to the backup Routing Engine:



CAUTION: Because graceful Routing Engine switchover is disabled, this switchover causes all line cards in the switch to reload. All network traffic passing through these line cards is lost during the line card reloads.

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

3. Verify that the default backup Routing Engine (shown as slot 1 in the command output) is now the master Routing Engine:

```
user@switch> show chassis routing-engine
```

You will see:

```
Routing Engine status:
Slot 0:
  Current state           Backup
  Election priority       Master (default)
Routing Engine status:
Slot 1:
  Current state           Master
  Election priority       Backup (default)
```

4. Install the new software package:

```
user@switch> request system software add package.tgz
```

5. Reboot the Routing Engine:

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

When the reboot completes, the prompt will reappear. Wait for this prompt to reappear before proceeding to the next step.

6. Log in to the default backup Routing Engine (slot 1) through the console port.
7. Re-enable graceful Routing Engine switchover:

```
[edit]
```

```
user@switch# activate chassis redundancy graceful-switchover
```

Re-enabling graceful Routing Engine switchover allows any future Routing Engine switchovers to occur without loss of any network traffic.

8. Re-enable nonstop active routing:

```
[edit]
user@switch# set routing-options nonstop-routing
```



NOTE: Automatic commit synchronization is a requirement for nonstop active routing. If you have not yet enabled it, do so with the `set system commit synchronize` command.

9. Save the configuration change:

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

10. To ensure that the resilient dual-root partitions feature operates correctly, execute the following command to copy the new Junos OS image into the alternate root partition on each Routing Engine:

```
user@switch> request system snapshot slice alternate routing-engine both
```

Resilient dual-root partitions allow the switch to boot transparently from the alternate root partition if the system fails to boot from the primary root partition.

If you want to return routing control to the Routing Engine that was the master Routing Engine at the beginning of the procedure (the default master Routing Engine), perform the next task.

Returning Routing Control to the Default Master Routing Engine (Optional)

The switch can maintain normal operations with the Routing Engine in slot 1 acting as the master Routing Engine after the software upgrade, so only perform this task if you want to return routing control to the default master Routing Engine in slot 0.

1. Transfer routing control back to the default master Routing Engine:

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

2. Verify that the default master Routing Engine (slot 0) is indeed the master Routing Engine:

```
user@switch> show chassis routing-engine
```

You will see:

```
Routing Engine status:
Slot 0:
  Current state      Master
  Election priority  Master (default)
```

```

Routing Engine status:
Slot 1:
  Current state      Backup
  Election priority  Backup (default)

```

Related Documentation

- [Installing Software on EX Series Switches \(J-Web Procedure\)](#)
- [Upgrading Software on an EX6200 or EX8200 Standalone Switch Using Nonstop Software Upgrade \(CLI Procedure\)](#)
- [Troubleshooting Software Installation on page 381](#)
- [Understanding Software Installation on EX Series Switches on page 59](#)

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 312](#).

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.

For specific installation procedures, see the following sections:

- [Determining Which Package or Packages to Install on page 103](#)
- [Installing Junos OS with Upgraded FreeBSD Over Plain Junos OS on page 106](#)
- [Installing Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Different Release on page 108](#)

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.

[Table 10 on page 105](#) indicates how to upgrade your device from a Junos OS release to a release that supports Junos OS with upgraded FreeBSD.

Before you begin:

1. Verify that the upgrade applies to your platform.

Different platforms support Junos OS with upgraded FreeBSD in different releases. To find which platforms support Junos OS with upgraded FreeBSD in which release, see [Feature Explorer](#) and enter one of the following:

- For non-virtualized, enter **freebsd** and select **Junos kernel upgrade to FreeBSD 10+**.
- For virtualized, enter **virtualization** and select **Virtualization of the Routing Engine**.

2. 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 10 on page 105](#). In using [Table 10 on page 105](#), be aware of the following:

- You can upgrade only to one of the three releases next to your current release. If you want to upgrade across more releases than this, you need to perform multiple upgrades.
- In determining what constitutes skipping no more than two releases, you must take into account which platforms you are upgrading. Whereas routing and switching platforms have released software in each main release, security platforms have had only the following releases: 17.4, 17.3, 15.1X49, and 12.3X48. Therefore, for a router to upgrade from Release 12.3 to the first release supporting Junos OS with upgraded FreeBSD (Release 15.1) would take multiple upgrades. But for a security device to upgrade from Release 12.3 to the first release supporting Junos OS with upgraded FreeBSD (Release 17.3) would take only one upgrade.

Table 10: Upgrade Path to Junos OS with the Upgraded FreeBSD

Release Sequence Type	Current Router's Junos OS Release	Release Supporting Junos OS with Upgraded FreeBSD	Upgrade Path	Example
Routing and Switching	earlier than Release 12.3	15.1	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	15.1	Upgrade in two steps.	To upgrade from Release 12.3, first upgrade to Release 13.3, then upgrade to Release 15.1. To upgrade from Release 13.2, first upgrade to Release 14.2, then upgrade to Release 15.1.
	13.3 to 14.2	15.1	Upgrade in a single step.	To upgrade from Release 13.3, upgrade directly to Release 15.1. To upgrade from Release 14.2, upgrade directly to either Release 15.1.
Security	12.3 to 17.2	17.3	Upgrade in a single step.	To upgrade from Release 12.3X48, upgrade directly to Release 17.3.
	15.1 to 17.2	17.3	Upgrade in a single step.	To upgrade from Release 15.1X49, upgrade directly to Release 17.3.
	15.1 to 17.3	17.4	Upgrade in a single step.	To upgrade from Release 15.1x49-D80, upgrade directly to Release 17.4.



NOTE: You can also downgrade from Junos OS Release with upgraded FreeBSD to a pre-upgraded-FreeBSD release of Junos OS, as long as the path complies with the Junos OS policy of skipping at most two earlier releases.

3. Download the Junos OS with upgraded FreeBSD package.

See [“Changes in Package Names for Junos OS with Upgraded FreeBSD”](#) on page 33.

Installing Junos OS with Upgraded FreeBSD Over Plain Junos OS

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 preserves only 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`

Before you begin, if you have important files in other directories that are not preserved, copy them from the router or switch to a secure location before upgrading the router or switch.



CAUTION: If you do a media install (either USB or network), the system is wiped and re-partitioned completely. Before you begin, if you have important files, copy them from the device to a secure location before upgrading the device.

To install Junos OS with upgraded FreeBSD over plain Junos OS:

1. Enter the **request system software add *install-package-name.tgz* 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 Junos OS with upgraded FreeBSD.



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/install-package-name.tgz
no-validate
```

The new Junos OS image is installed on the device.

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 *install-package-name.tgz*** command. 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.



NOTE: The output shows the OS kernel, OS runtime, and other packages installed on the device.

Installing Junos OS with Upgraded FreeBSD Over Junos OS with Upgraded FreeBSD of a Different Release



CAUTION: If you do a media install (either USB or network), the system is wiped and re-partitioned completely. Before you begin, if you have important files, copy them from the device to a secure location before upgrading the device.

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 device.

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 device.

Related Documentation

- [Downgrading Junos OS from Upgraded FreeBSD on page 109](#)
- [Release Information for Junos OS with Upgraded FreeBSD on page 32](#)
- [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 465](#)
- [request system reboot \(Junos OS with Upgraded FreeBSD\) on page 449](#)

- *Establishing an SSH Connection for a NETCONF Session*

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 103](#).
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 109](#)
- [Downgrading from Junos OS with Upgraded FreeBSD Release 17.4 or Later to an Earlier Release of Junos OS with Upgraded FreeBSD on page 110](#)
- [Downgrading from Junos OS with Upgraded FreeBSD Release 17.3 or Earlier to an Earlier Release of Junos OS with Upgraded FreeBSD on page 111](#)

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 a 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.

Downgrading from Junos OS with Upgraded FreeBSD Release 17.4 or Later to an Earlier Release of Junos OS with Upgraded FreeBSD

This procedure is applicable when downgrading from Junos OS with Upgraded FreeBSD Release 17.4 or later to an earlier release of Junos OS with Upgraded FreeBSD.



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 downgrade from Junos OS with upgraded FreeBSD Release 17.3 or earlier to an earlier release of Junos OS with upgraded FreeBSD:

1. Enter the **request system software add *package-name* no-validate reboot** command from the operational mode in the CLI:



NOTE: The **no-copy** option is enabled by default.

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 an earlier release of Junos OS with upgraded FreeBSD from Junos OS with upgraded FreeBSD Release 17.4 or later.



NOTE: To validate current configuration on an downgrade from Junos OS with upgraded FreeBSD Release 17.4 or later to an earlier release of 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 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.

Downgrading from Junos OS with Upgraded FreeBSD Release 17.3 or Earlier to an Earlier Release of Junos OS with Upgraded FreeBSD

This procedure is applicable when downgrading from Junos OS with Upgraded FreeBSD Releases 17.3 through 15.1 to an earlier release of Junos OS with Upgraded FreeBSD.



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 downgrade from Junos OS with upgraded FreeBSD Release 17.3 or earlier to an earlier release of Junos OS with upgraded FreeBSD:

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 Release 17.3 or earlier to Junos OS with upgraded FreeBSD Release 17.2 or earlier.

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 103](#)
- [Release Information for Junos OS with Upgraded FreeBSD on page 32](#)
- [request system snapshot \(Junos OS with Upgraded FreeBSD\) on page 465](#)
- [request system reboot \(Junos OS with Upgraded FreeBSD\) on page 449](#)

Installing Junos OS Software with Junos Automation Enhancements

Junos operating system (Junos OS) with Junos Automation Enhancements is a full-featured version of Junos OS with Veriexec disabled, which can only be installed on supported devices.



NOTE: You must install the `jinstall-qfx-5-flex-x.tgz` software bundle in order to use the automation enhancements.

Before you install software, download the Junos OS `jinstall-qfx-5-flex-x.tgz` software bundle. For information on downloading and accessing the files, see [“Upgrading Software” on page 79](#).



BEST PRACTICE: Before you install the software, back up any critical files in `/var/home`. For more information regarding how to back up critical files, contact Customer Support at <https://www.juniper.net/support>.

Install the software:

1. Execute the **request system software add** command with the **validate** option:

- If the installation package resides locally on the switch, execute the **request system software add validate *pathname source* reboot** command, using the following format:

```
user@switch> request system software add validate /var/tmp/jinstall-qfx-5-flex-x.tgz
reboot
```

- If the installation package resides remotely, execute the **request system software add validate *pathname source* reboot** command, using the following format:

```
user@switch> request system software add validate
ftp://ftpsrvr/directory/jinstall-qfx-5-flex-x.tgz reboot
```

2. After the reboot has finished, verify that the new version of software has been properly installed by executing the **show version** command.

```
user@switch> show version
root@qfx5100-24q-et013> show version
fpc0:
```

```
-----
Hostname: qfx5100-24q-et013
Model: qfx5100-24q-2p
JUNOS Base OS Software Suite [13.2X51-D20]
JUNOS Base OS boot [13.2X51-D20]
JUNOS Crypto Software Suite [13.2X51-D20]
JUNOS Online Documentation [13.2X51-D20]
JUNOS Kernel Software Suite [13.2X51-D20]
JUNOS Packet Forwarding Engine Support (qfx-x86-32) [13.2X51-D20]
JUNOS Routing Software Suite [13.2X51-D20]
JUNOS Enterprise Software Suite [13.2X51-D20]
JUNOS py-base-i386 [13.2X51-D20]
Puppet on Junos [2.7.19_1.junos.i386]
Ruby Interpreter [11.10.4_1.junos.i386]
Chef [11.10.4_1.junos.i386]
junos-ez-stdlib [11.10.4_1.junos.i386]
JUNOS Host Software [13.2X51-D20]
JUNOS for Automation Enhancement
```



NOTE: If you are upgrading a device from standard Junos OS to use Junos Automation Enhancements and you are *not* loading the new factory default configuration, you need to use the following procedure.

To upgrade an existing device from standard Junos to use Junos Automation Enhancements:

1. Edit your existing Junos OS configuration to include the following configuration statements:

```
[edit]
user@switch# set system extensions providers juniper license-type juniper
deployment-scope commercial
user@switch# set system extensions providers chef license-type juniper
deployment-scope commercial
```



NOTE: The factory default configuration of the QFX5100 switch `jinstall-qfx-5-flex-x.tgz` software bundle is a Layer 3 configuration, whereas the factory default configuration for QFX5100 switch software bundles is a Layer 2 configuration. Therefore, if you are running the `jinstall-qfx-5-flex-x.tgz` software bundle on a QFX5100 switch and you use the `load factory-default` command, the resulting factory default configuration is set up for Layer 3 interfaces.

This is the factory default configuration for QFX5100 switch `jinstall-qfx-5-flex-x.tgz` software bundle:

```
user@switch> show configuration
```

```
system syslog user * any emergency
system syslog file messages any notice
system syslog file messages authorization info
system syslog file interactive-commands interactive-commands any
system extensions providers juniper license-type juniper deployment-scope
commercial
system extensions providers chef license-type juniper deployment-scope commercial
system commit factory-settings reset-virtual-chassis-configuration
system commit factory-settings reset-chassis-lcd-menu
system processes app-engine-virtual-machine-management-service traceoptions level
notice
system processes app-engine-virtual-machine-management-service traceoptions flag
all
interfaces et-0/0/0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/0:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/0:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/0:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/0:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/1:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/1:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/1:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/1:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
```

Copyright © 2018, Juniper Networks, Inc. 115

Copyright © 2018, Juniper Networks, Inc.

```
protocols igmp-snooping vlan default
vlans default vlan-id 1
```

Related Documentation

- *Overview of Junos Automation Enhancements*

Upgrading Software by Using Automatic Software Download

Related Documentation

- *Verifying That Automatic Software Download Is Working Correctly*
- [Understanding Software Installation on EX Series Switches on page 59](#)

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 <https://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*

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 120](#)
- [Overview on page 121](#)
- [Configuration on page 121](#)
- [Verification on page 122](#)

Requirements

Before you begin:

- Verify the available space on the internal media.
- Download the software package.

- 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*.

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 122](#)

Verifying the Junos OS Upgrade Installation

Purpose Verify that the Junos OS upgrade was installed.

Action From operational mode, enter the **show system** command.

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 123](#)
- [Overview on page 123](#)
- [Configuration on page 123](#)
- [Verification on page 125](#)

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 125](#)

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 220](#)
- [Understanding Junos OS Upgrades for SRX Series Devices on page 61](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 120](#)
- [Restarting and Halting SRX Series Devices on page 250](#)

Upgrading Jloader Software on QFX Series Devices

Jloader software contains a boot loader (Uboot), which is used to bring up QFX Series devices and load the Junos OS from the flash memory of these devices. You can upgrade Jloader software on QFX3500 switches, QFX3500 and QFX3600 Node devices, and QFX3600-I and QFX3008-I Interconnect devices.



NOTE: Before you upgrade the Jloader software, see [Table 11 on page 125](#), [Table 12 on page 126](#), and [Table 13 on page 126](#) to make sure that you are upgrading to the right version of Jloader software for the Junos OS software release running on your QFX3500 switches, or Node devices and Interconnect devices in your QFabric system.

See [Table 14 on page 126](#) to see which Uboot software versions are available and the filenames of the Jloader software packages.

Table 11: Junos OS and Jloader Software Compatibility Matrix for the QFX3500 Switch and QFX3500 Node Device

Junos OS Software Version	11.2	11.4	11.5	11.8
11.3R1 and later (QFX3500 switch)	Supported	Supported	Not supported	Supported and recommended
11.3X30.6 and later (QFX3500 Node device)	Supported	Supported	Not supported	Supported and recommended

Table 11: Junos OS and Jloader Software Compatibility Matrix for the QFX3500 Switch and QFX3500 Node Device (continued)

Junos OS Software Version	1.1.2	1.1.4	1.1.5	1.1.8
12.1X49-D1 and later (QFX3500 switch)	Supported	Supported	Not supported	Supported and recommended
12.2X50-D1 and later (QFX3500 switch and QFX3500 Node device)	Supported	Supported	Not supported	Supported and recommended



NOTE: An en dash means that the item is not applicable.

Table 12: Junos OS and Jloader Software Compatibility Matrix for the QFX3008-I Interconnect Device

Junos OS Software Version	1.1.2	1.1.4	1.1.5	1.1.8
11.3X30.9 and later (QFX3008-I Interconnect device)	Supported	Supported	Not supported	Supported and recommended
11.3X30.6 and later (QFX3008-I Interconnect device)	Supported	Supported	Not supported	Supported and recommended
12.2X50-D10.3 and later (QFX3008-I Interconnect device)	Supported	Supported	Not supported	Supported and recommended



NOTE: An en dash means that the item is not applicable.

Table 13: Junos OS and Jloader Software Compatibility Matrix for the QFX3600-I Interconnect Device and QFX3600 Node Device

Junos OS Software Version	1.1.2	1.1.4	1.1.5	1.1.8
12.2X50-D10.3 and later (QFX3600-I Interconnect Device and QFX3600 Node Device)	-	-	Supported	Supported and recommended
12.2X50-D20 and later (QFX3600 switch)	-	-	Supported	Supported and recommended

Table 14: Uboot Software Release and Jloader Software Compatibility Matrix

Uboot Software Release Number	Jloader Software Package Name
1.1.2	jloader-qfx-11.3X30.9-signed.tgz

Table 14: Uboot Software Release and Jloader Software Compatibility Matrix (continued)

Uboot Software Release Number	Jloader Software Package Name
1.1.4 (11.3R3 and 11.3R2 releases only. Not supported on 11.3R1)	jloader-qfx-11.3I20120127_0733_dc-builder-signed.tgz
1.1.4 (12.1R1 release and later)	jloader-qfx-12.1-20120125_pr.0-signed.tgz
1.1.5 (12.2X50-D10.3 and later)	jloader-qfx-12.2X50.D10.3-signed.tgz
1.1.8 (13.1X50-D15.1 and later)	jloader-qfx-13.3-20130831_pr_branch_qfd.0.tgz

Jloader Software Version 1.1.4 Guidelines

Jloader Release 1.1.4 is compatible with Junos OS Release 11.3R3 and 11.3R2, and Junos OS Release 12.1R1 and later. Jloader Release 1.1.4 is not compatible with Junos OS Release 11.3R1. The Jloader software package names are different for versions 1.1.4 (Junos OS 11.3R3 and 11.3R2) and 1.1.4 (Junos OS 12.2R1 release and later), but the binaries are the same. Because the binaries are the same, you can upgrade or downgrade to any Junos OS release.

- If you have Junos OS Release 11.3 installed and want to upgrade the Jloader software from version 1.1.2 to version 1.1.4, you need to upgrade using the **jloader-qfx-11.3I20120127_0733_dc-builder-signed.tgz** software package.
- If you have Junos OS Release 11.3R2 installed and want to upgrade to Junos OS Release 12.1, you do not need to upgrade the Jloader Release and can continue to use Jloader Release 1.1.2.
- If you have Junos OS Release 12.1 installed and want to upgrade the Jloader software from version 1.1.2 to version 1.1.4, you need to upgrade using the **jloader-qfx-12.1-20120125_pr.0-signed.tgz** software package.
- If you upgrade to Junos OS Release 12.1, you can upgrade to Jloader Release 1.1.4 using the **jloader-qfx-12.1-20120125_pr.0-signed.tgz** software package.

Upgrading Jloader Software on a QFX3500 Switch

The Jloader software for a QFX3500 switch resides in two flash memory banks. At any time, one bank acts as the primary bank, and the QFX3500 switch boots from it. The other bank is the backup bank—if the QFX3500 switch cannot boot from the primary bank, it boots from the backup bank. When you upgrade the Jloader software, the upgraded software is installed in the backup bank, which then becomes the new primary bank. Thus the primary and backup banks alternate each time you upgrade the Jloader software, with the primary bank containing the most recently installed version of the software, and the backup bank containing the previous version. To upgrade the Jloader software on a QFX3500 switch, you must perform the upgrade twice: once for each bank. Each upgrade requires that you to reboot the QFX3500 switch.



NOTE: If you are running Junos OS Release 11.3R1 or Junos OS Release 11.3R2, you must use the `no-validate` option when you issue the `request system software add` command to upgrade the Jloader software. Otherwise, the installation will fail and you receive a configuration error. The `no-validate` option is not required for Junos OS Release 11.3R3 and later.



NOTE: After you upgrade the Jloader software on the first bank, the software package is deleted after you reboot. Make sure that you have either downloaded the Jloader software package to either a remote site or in a local directory on the switch, such as the `/var/tmp` directory on the QFX3500 device.

1. In a browser, go to <https://www.juniper.net/support/downloads/junos.html> .
The Junos Platforms Download Software page appears.
2. In the QFX Series section of the Junos Platforms Download Software download page, select the QFX Series platform software you want to download.
3. Select the number of the software version that you want to download in the Release: pull-down window to the right of the tabs on the Download Software page.
4. Select the Software tab and then select the install package you want to download in the Install Package section.
5. In the pop-up Alert box, click the link to the Product Support Notification (PSN) document.
6. Enter your name and password and press **Enter**.

7. Read the End User License Agreement, click the **I agree** radio button, and then click **Proceed**.
8. Open or save the **jloader-qfx-version-signed.tgz** file either to a local system or to a remote location. If you are saving the installation package to a remote system, make sure that you can access it using HTTP, TFTP, FTP, or scp.
9. Log in to the QFX3500 switch and enter the shell. We recommend using a console connection.
10. Determine the version of the Jloader software package installed on the switch.

For example:

```
root@switch% ls
gres-tp krt_gencfg_filter.txt
jloader-qfx-11.3-20110510.0-signed.tgz
```

11. Determine the version of the Uboot software that is running in the bank:

For example:

```
root@switch% kenv | grep boot.version
boot.version="1.0.7"
```

12. Enter the CLI and install the Jloader software package.

- To install a Jloader software package that is located in the **/var/tmp** directory, issue the **request system software add /var/tmp/jloader-qfx-version.tgz no-validate** command:

For example:

```
user@switch> request system software add
/var/tmp/jloader-qfx-11.3-20110510.0-signed.tgz no-validate
```

You see the following messages during the installation:

```
Verified jloader-qfx-11.3-20110510.0.tgz signed by PackageProduction_11_3_0
Adding jloader-qfx...
Installation in progress, please wait...
Mounted jloader-qfx package on /dev/md8...
Verified manifest signed by PackageProduction_11_3_0
Verified jloader-qfx-11.3-20110510.0 signed by PackageProduction_11_3_0
Registering jloader-qfx as unsupported

Installation finished successfully.
Please reboot to activate the package
Saving package file in /var/sw/pkg/jloader-qfx-11.3-20110510.0-signed.tgz
...
Saving state for rollback ...

juniper@qfx3500>
```

- To install a Jloader software package located on a remote server using FTP, issue the **request system software add /ftp://hostname/pathname/jloader-qfx-version-signed.tgz no-validate** command.

For example:

```
user@switch> request system software add  
/ftp://hostname/pathname/jloader-qfx-11.3-20110510.0-signed.tgz no-validate
```

- To install a Jloader software package located on a remote server using HTTP, issue the **request system software add /http://hostname/pathname/jloader-qfx-version-signed.tgz no-validate** command.

For example:

```
user@switch> request system software add  
/http://hostname/pathname/jloader-qfx-11.3-20110510.0-signed.tgz no-validate
```

13. When prompted, reboot the Control Board by issuing the **request system reboot** command.

For example:

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

14. Enter the shell and verify that the version of the Uboot software in the primary bank is the version you just installed.

For example:

```
root@switch% kenv | grep boot.version  
boot.version="1.1.1"
```

15. To install the Jloader software package on the current backup bank, repeat Step 10 through Step 14.

Upgrading Jloader Software on a QFabric System

This procedure explains how to upgrade the Jloader software on your Node devices and Interconnect devices. The example shows how to upgrade the Jloader Release 1.1.1 to 1.1.2 on a Node device with the serial number BBAK1186.



NOTE: Before you upgrade the Jloader software, make sure you have the serial numbers of the Node devices, Interconnect devices, and Control Boards in the Interconnect devices you want to upgrade.

1. Issue the **show chassis hardware node-device ?** command to view the serial numbers of the Node devices.

For example:

```
user@qfabric> show chassis hardware node-device ?
<node-device>      Node device identifier
BBAK1186            Node device
BBAK3149            Node device
BBAK3177            Node device
BBAK8063            Node device
BBAK8799            Node device
P2443-C             Node device
P2515-C             Node device
P3708-C             Node device
P3885-C             Node device
P3916-C             Node device
node0               Node device
node1               Node device
node2               Node device
node3               Node device
node4               Node device
node5               Node device
node6               Node device
node7               Node device
node8               Node device
```

An example of a Node device serial number is BBAK1186.

2. Issue the **show chassis hardware interconnect-device ?** command to view the serial numbers of the Interconnect devices.

For example:

```
user@qfabric> show chassis hardware interconnect-device ?
Possible completions:
interconnect-device  Interconnect device identifier
IC-F1052             Interconnect device
IC-F3947             Interconnect device
```

The Interconnect device serial numbers are IC-F1052 and IC-F3947.

3. Issue the **show chassis hardware interconnect-device *name*** command to view the serial numbers of the Control Boards in the Interconnect device.

For example:

```
user@qfabric> show chassis hardware interconnect-device IC-F3947
Hardware inventory:
Item          Version  Part number  Serial number  Description
Chassis       REV 10   F3947        F3947          QFXC08-3008
Midplane      REV 10   750-035835   F3947-C        QFX Midplane
CB 0          REV 14   750-035855   ZJ9432         QFX Chassis Control
Board
Routing Engine 0      BUILTIN    BUILTIN        QFX Routing Engine
CB 1          REV 14   750-035855   ZJ9404         QFX Chassis Control
Board
```

The Control Board serial numbers are ZJ9432 and ZJ9404.

4. Issue the **show chassis firmware node-device *name*** command to see which version of Uboot software you have installed on your Node device.

For example:

```
user@qfabric> show chassis firmware node-device BBAK1186

Part          Type      Version
node4         U-Boot   1.1.6 (May 10 2011 - 04:52:59) 1.1.1
loader        FreeBSD/MIPS U-Boot bootstrap loader 0.1
```

The Uboot software version is 1.1.1. The loader software version appears after the timestamp for U-Boot 1.1.6.

5. Issue the **show chassis firmware interconnect-device *name*** command to see which version of Uboot software you have installed on the Routing Engines located on the Control Boards of the Interconnect device.

For example:

```
user@qfabric> show chassis firmware interconnect-device IC-F3947

Part          Type      Version
Routing Engine 0      U-Boot   U-Boot 1.1.6 (Jan 27 2012 - 03:24:34) 1.1.4
loader        FreeBSD/MIPS U-Boot bootstrap loader 0.1
Routing Engine 1      U-Boot   U-Boot 1.1.6 (Jan 27 2012 - 03:24:34) 1.1.4
loader        FreeBSD/MIPS U-Boot bootstrap loader 0.1
```

The Uboot software version is 1.1.4. The loader software version appears after the timestamp for U-Boot 1.1.6.

6. In a browser, go to <https://www.juniper.net/support/downloads/junos.html>.

The Junos Platforms Download Software page appears.

7. In the QFX Series section of the Junos Platforms Download Software download page, select the QFX Series platform software you want to download.
8. Select the number of the software version that you want to download in the Release: pull-down window to the right of the tabs on the Download Software page.
9. Select the **Software** tab and then select the install package you want to download in the Install Package section.
10. In the pop-up Alert box, click the link to the Product Support Notification (PSN) document.
11. Enter your username and password, and press **Enter**.
12. Read the End User License Agreement, click the **I agree** radio button, and then click **Proceed**.
13. Open or save the **jloader-qfx-version-signed.tgz** file either to a local system or to a remote location. If you are saving the installation package to a remote system, make sure that you can access it using HTTP, TFTP, FTP, or scp.
14. Retrieve the software from the location in which you downloaded it. To do this, issue the **request system software download /path/package-name** command.

For example:

```
user@qfabric> request system software download  
ftp://server/files/jloader-qfx-11.3X30.9-signed.tgz
```

15. Log in to the Director device as root and enter the shell to verify that you have downloaded the Jloader software package. We recommend using a console connection. The software package is copied from where you downloaded it and is placed locally on the QFabric system in the **/pbdata/packages** directory.

For example:

```
[root@dg0] # pwd  
/pbdata/packages  
[root@dg0] # ls  
jloader-qfx-11.3X30.9-signed.tgz
```

16. Before you copy over the Jloader software package to the Node device or Interconnect device, determine the directory that matches the serial number of the Node device or Interconnect device that you want to upgrade. View the remote logs and the Node device and Interconnect device serial numbers by issuing the **ls /pdbata/export/rlogs** command at the command line of the Director device before you copy the software package over to the device.



NOTE: The `/pbdata/export/rlogs/node-device-serial-ID` and `/pbdata/export/rlogs/interconnect-device-serial-ID` directories on the Director device are NFS mounted as the `/tftpboot/logfiles` directories on the Node device and Interconnect device. These directories are created for all Node devices and Interconnect devices in a QFabric system. The Jloader files are stored in the `/tftpboot/logfiles` directories for each Node device and Interconnect device.

For example:

```
[root@dg0 tmp] # ls /pbdata/export/rlogs
02de4930-828b-11e1-a319-00e081c57938  c9898afe-828b-11e1-956c-00e081c57938
04103b2a-29d5-e011-bf8a-0e6bdf3aa1e6  eebe4aac-828b-11e1-85e2-00e081c57938
1e2739e0-828b-11e1-bf74-00e081c57938  F1052
8d8a978c-828b-11e1-a833-00e081c57938  F3947
ad55b89e-828b-11e1-b70e-00e081c57938  P2443-C
BBAK1186                               P2515-C
BBAK3149                               P3708-C
BBAK3177                               P3885-C
BBAK8063                               P3916-C
BBAK8799
```

BBAK1186 is the serial number of the Node device that needs to be upgraded.

17. Copy the Jloader software package from the `/var/tmp` directory to the `/pbdata/export/rlogs/BBAK1186` directory.

For example:

```
[root@dg0 tmp] # cp jloader-qfx-11.3X30.9-signed.tgz /pbdata/export/rlogs/BBAK1186
```

18. Confirm that the Jloader software package you copied over is in the `/pbdata/export/rlogs/BBAK1186` directory.

For example:

```
[root@dg0 tmp] # ls /pbdata/export/rlogs/BBAK1186
jloader-qfx-11.3X30.9-signed.tgz
```

19. Issue the `/root/dns.dump` command to find out the internal IP addresses of the Node device or Interconnect device.

```
[root@dg0 tmp] # /root/dns.dump
; <<>> DiG 9.3.6-P1-RedHat-9.3.6-4.P1.e15 <<>> -t axfr pkg.test.device.net
@169.254.0.1
;; global options: printcmd
pkg.test.device.net. 600 IN SOA ns.pkg.test.device.net.
mail.pkg.test.device.net. 152 3600 600 7200 3600
pkg.test.device.net. 600 IN NS ns.pkg.test.device.net.
pkg.test.device.net. 600 IN A 169.254.0.1
pkg.test.device.net. 600 IN MX 1 mail.pkg.test.device.net.
dcfnod---DCF-ROOT.pkg.test.device.net. 45 IN A 169.254.192.17
dcfnod---DRE-0.pkg.test.device.net. 45 IN A 169.254.3.3
dcfnod-8d8a978c-828b-11e1-a833-00e081c57938.pkg.test.device.net. 45 IN A
```

```

169.254.128.19
dcfnode-ad55b89e-828b-11e1-b70e-00e081c57938.pkg.test.device.net. 45 IN A
169.254.128.20
dcfnode-BBAK1186.pkg.test.device.net. 45 IN A 169.254.128.14

```

The internal IP address for BBAK1186 is 169.254.128.14.

20. Upgrade the Jloader software on the Node device or Interconnect device.

Before you can upgrade the Jloader software, you need to use SSH to log in to the Node device or Interconnect device and verify that the software is in the **/tftpboot/logfiles** directory.

- a. Use SSH to log in to the Node device or Interconnect device.

For example:

```

[root@dg0 tmp] # ssh 160.254.128.14
root@169.254.128.14's password:
--- JUNOS 11.3X30.10 built 2012-03-11 22:55:43 UTC
At least one package installed on this device has limited support.
Run 'file show /etc/notices/unsupported.txt' for details.
root@sng3%

```

- b. Verify that the Jloader software package is in the **tftpboot/logfiles** directory of the Node device or Interconnect device.

For example:

```

root@sng3% ls /tftpboot/logfiles
.index                               jloader-qfx-11.3X30.9-signed.tgz

```

- c. Copy the Jloader software package from the **/tftpboot/logfiles** directory to the **/var/tmp** directory of the Node device or Interconnect device.

For example:

```

root@sng3% cp /tftpboot/logfiles/jloader-qfx-11.3X30.9-signed.tgz /var/tmp

```

- d. Verify that the Jloader software package is in the **/var/tmp** directory of the Node device or Interconnect device.

For example:

```

root@sng3% ls /var/tmp
.snap                               jloader-qfx-11.3X30.9-signed.tgz
    tmp
gres-tp                             krt_gencfg_filter.txt
    vc-autoupgrade
if-rtbdb                             rtsdb

```

- e. Enter CLI mode and issue the **request system software add /var/tmp/jloader-qfx-version-signed.tgz** command.

For example:

```

root@sng3% cli

```

```

root@sng3> request system software add /var/tmp/jloader-qfx-11.3X30.9-signed.tgz
Validating on fpc0
Checking compatibility with configuration
Initializing...
Using jbase-11.3X30.10
Verified manifest signed by PackageProduction_11_3_0
Verified jbase-11.3X30.10 signed by PackageProduction_11_3_0
Using /var/tmp/jloader-qfx-11.3X30.9-signed.tgz
Verified jloader-qfx-11.3X30.9.tgz signed by PackageProduction_11_3_0
Using jloader-qfx-11.3X30.9.tgz
Checking jloader-qfx requirements on /
Verified manifest signed by PackageProduction_11_3_0
Verified jloader-qfx-11.3X30.9 signed by PackageProduction_11_3_0
Using jkernel-qfx-11.3X30.10
Verified manifest signed by PackageProduction_11_3_0
Verified jkernel-qfx-11.3X30.10 signed by PackageProduction_11_3_0
Using jroute-qfx-11.3X30.10
Verified manifest signed by PackageProduction_11_3_0
Verified jroute-qfx-11.3X30.10 signed by PackageProduction_11_3_0
Using jcrypto-qfx-11.3X30.10
Verified manifest signed by PackageProduction_11_3_0
Verified jcrypto-qfx-11.3X30.10 signed by PackageProduction_11_3_0
Using jweb-qfx-11.3X30.10
Verified manifest signed by PackageProduction_11_3_0
Verified jweb-qfx-11.3X30.10 signed by PackageProduction_11_3_0
Using jswitch-qfx-11.3X30.10
Verified manifest signed by PackageProduction_11_3_0
Verified jswitch-qfx-11.3X30.10 signed by PackageProduction_11_3_0
Hardware Database regeneration succeeded
Validating against /config/juniper.conf.gz
mgd: commit complete
Validation succeeded
Done with validate on all chassis

```

```

fpc0:
Verified jloader-qfx-11.3X30.9.tgz signed by PackageProduction_11_3_0
Adding jloader-qfx...
Installation in progress, please wait...
Mounted jloader-qfx package on /dev/md10...
Verified manifest signed by PackageProduction_11_3_0
Verified jloader-qfx-11.3X30.9 signed by PackageProduction_11_3_0
#####
#####
Installation finished successfully.
Please reboot to activate the package
Saving package file in /var/sw/pkg/jloader-qfx-11.3X30.9-signed.tgz ...
Saving state for rollback ...

```

Upgrade has completed successfully.
Reboot is now required.

- f. Reboot both the Node device and Interconnect device twice, because they each contain two partitions.

For example:

```

root@sng3> request system reboot
Reboot the system ? [yes,no] (no) yes
Shutdown NOW!
[pid 37663]

```



```
root@sng3>
```

```
*** FINAL System shutdown message from root@sng3 ***
```

```
System going down IMMEDIATELY
```

- g. Verify that the Uboot software on the Node device or Interconnect device has been upgraded to the new Uboot software by logging in to the QFabric CLI and issuing either the **show chassis firmware node-device *name*** command or the **show chassis firmware interconnect-device *name*** command.

For example:

```
user@qfabric> show chassis firmware node-device BBAK1186
```

Part	Type	Version
node4	U-Boot	1.1.6 (Nov 19 2011 - 11:42:07) 1.1.2
	loader	FreeBSD/MIPS U-Boot bootstrap loader
0.1		

The Uboot software version is now 1.1.2. The loader software version appears after the timestamp for U-Boot 1.1.6.

- See Also**
- *Performing a Nonstop Software Upgrade on the QFabric System*
 - [Upgrading Software on page 79](#)
 - *Upgrading Software on a QFabric System*
 -

CHAPTER 3

Using the USB Flash Drive and Boot Loader to Upgrade Software

- [Preparing the USB Flash Drive to Upgrade Junos OS on SRX Series Devices on page 139](#)
- [Installing Junos OS on SRX Series Devices Using a USB Flash Drive on page 141](#)
- [Booting an EX Series Switch Using a Software Package Stored on a USB Flash Drive on page 142](#)
- [Upgrading the Boot Loader on SRX Series Devices on page 143](#)
- [Installing Junos OS on SRX Series Devices from the Boot Loader Using a TFTP Server on page 144](#)
- [Installing Junos OS on SRX Series Devices from the Boot Loader Using a USB Storage Device on page 147](#)
- [Verifying Junos OS and Boot Loader Software Versions on an EX Series Switch on page 147](#)

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.

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.

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 215](#). 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.
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 \(CLI Procedure\) on page 90](#)
- [Installing Software on EX Series Switches \(J-Web Procedure\)](#)
- [Understanding Software Installation on EX Series Switches on page 59](#)
- 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.

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.

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 15 on page 145](#).

Table 15: 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.

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.

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 148](#)
- [Verifying the Loader Software Version on page 149](#)

- [Verifying Which Root Partition Is Active on page 149](#)
- [Verifying the Junos OS Version in Each Root Partition on page 150](#)

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 381](#)
 - [Troubleshooting a Switch That Has Booted from the Backup Junos OS Image on page 384](#)
 - [Understanding Resilient Dual-Root Partitions on Switches on page 187](#)

CHAPTER 4

Upgrading the BIOS and Firmware

- [Before You Begin Installing or Upgrading the Firmware on page 153](#)
- [Understanding BIOS Upgrades on SRX Series Devices on page 155](#)
- [Disabling Auto BIOS Upgrade on SRX Series Devices on page 157](#)
- [Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\) on page 158](#)
- [Upgrading Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\) on page 159](#)
- [Installing Firmware on the 100-Gigabit DWDM OTN MIC \(MIC3-100G-DWDM\) on page 161](#)
- [Upgrading Firmware on the 100-Gigabit DWDM OTN MIC \(MIC3-100G-DWDM\) on page 162](#)

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 161](#) or [“Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\)” on page 158](#).

```
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 161](#)
 - [Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\) on page 158](#)
 - *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 162](#)
 - [Upgrading Firmware on the 5-Port 100-Gigabit DWDM OTN PIC \(PTX-5-100G-WDM\) on page 159](#)

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 16 on page 155](#) Lists the CLI commands used for manual BIOS upgrade.

Table 16: 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:

1. **Install the jloader-srxsme package.**

1. 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.

2. 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 61](#)
- [Installing Junos OS on SRX Series Devices Using a USB Flash Drive on page 141](#)
- [Installing Junos OS on SRX Series Devices from the Boot Loader Using a TFTP Server on page 144](#)

- [Installing Junos OS on SRX Series Devices from the Boot Loader Using a USB Storage Device on page 147](#)
- [Disabling Auto BIOS Upgrade on SRX Series Devices on page 157](#)

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.

Starting in Junos OS Release 15.1X49-D120, the **set chassis routing-engine bios uninterrupt** command can be used on SRX300, SRX320, SRX340, and SRX345 devices to disable user inputs at U-boot, boot loader and Junos-Kernel boot stage.

To disable the user inputs at u-boot, boot loader and Junos Kernel boot stage, use the **chassis routing-engine bios** command as following:

```
user@host> set chassis routing-engine bios uninterrupt
```



NOTE: To disable user inputs at U-boot and boot loader stage using the `chassis routing-engine bios` command, SRX devices must have u-boot version of v3.2 or a higher version, and loader version of v2.9 or a higher version.

You can check the version number at console output when your device boots up as shown in the following sample:

```
U-Boot 2013.07-JNPR-3.4 (Build time: Aug 02 2017 - 18:57:37)
FreeBSD/MIPS U-Boot bootstrap loader, Revision 2.9
```

You can also check the u-boot and loader version at Junos shell prompt as shown the following sample:

```
root@% kenv
      LINES="24"
      boot.ver="3.5"
      loader.name="FreeBSD/MIPS U-Boot bootstrap loader"
      loader.version="2.9"
root@%
```



WARNING: On SRX Series devices, if both `set system ports console insecure` and `set chassis routing-engine bios uninterrupt` options are configured, there is no alternative recovery method available incase Junos OS fails to boot and the device might become unusable.

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.
15.1X49-D120	Starting in Junos OS Release 15.1X49-D120, the set chassis routing-engine bios uninterrupt command can be used on SRX300, SRX320, SRX340, and SRX345 devices to disable user inputs at U-boot, boot loader and and Junos-Kernel boot stage.

Related Documentation

- [Understanding Junos OS Upgrades for SRX Series Devices on page 61](#)
- [Understanding BIOS Upgrades on SRX Series Devices on page 155](#)

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 153](#).

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 92 or “Installing the Software Package on a Router with a Single Routing Engine” on page 88 for more information.
2. Download the firmware package from <https://www.juniper.net/support/>. For information about downloading software packages, see “Downloading Software” on page 76.



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 153](#)
- [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 159](#)

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 153.

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 153](#)
- [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 158](#)
- `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 153](#).

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 92](#) or [“Installing the Software Package on a Router with a Single Routing Engine” on page 88](#) for more information.
- Download the firmware package from <https://www.juniper.net/support/>. For information about downloading software packages, see [“Downloading Software” on page 76](#).



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 153](#)
- [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 162](#)

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 153](#).

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 153](#)
- [Installing Firmware on the 100-Gigabit DWDM OTN MIC \(MIC3-100G-DWDM\) on page 161](#)
- `show system firmware`
- [Understanding the MIC3-100G-DWDM MIC](#)

CHAPTER 5

Reinstalling Software

- Checklist for Reinstalling Junos OS on page 165
- Log the Software Version Information on page 167
- Log the Hardware Version Information on page 168
- Log the Chassis Environment Information on page 169
- Log the System Boot-Message Information on page 169
- Log the Active Configuration on page 171
- Log the Interfaces on the Router on page 171
- Log the BGP, IS-IS, and OSPF Adjacency Information on page 172
- Log the System Storage Information on page 173
- Back Up the Currently Running and Active File System on page 174
- Reinstall Junos OS on page 175
- Reconfigure Junos OS on page 175
- Configure Host Names, Domain Names, and IP Addresses on page 179
- Protecting Network Security by Configuring the Root Password on page 181
- Check Network Connectivity on page 183
- Copy Backup Configurations to the Router on page 183
- After You Reinstall Junos OS on page 183
- Compare Information Logged Before and After the Reinstall on page 184
- Back Up the New Software on page 185

Checklist for Reinstalling Junos OS

Table 17 on page 165 provides links and commands for reinstalling Junos OS.

Table 17: Checklist for Reinstalling Junos OS

Tasks	Command or Action
Before You Reinstall Junos OS	
1. Log the Software Version Information on page 167	<code>show version</code> <i>save filename</i>

Table 17: Checklist for Reinstalling Junos OS (continued)

Tasks	Command or Action
2. Log the Hardware Version Information on page 168	<code>show chassis hardware save filename</code>
3. Log the Chassis Environment Information on page 169	<code>show chassis environment save filename</code>
4. Log the System Boot-Message Information on page 169	<code>show system boot-messages save filename</code>
5. Log the Active Configuration on page 171	<code>show configuration save filename</code>
6. Log the Interfaces on the Router on page 171	<code>show interface terse save filename</code>
7. Log the BGP, IS-IS, and OSPF Adjacency Information on page 172	<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 173	<code>show system storage save filename</code>
9. Back Up the Currently Running and Active File System on page 174	<code>request system snapshot</code>
10.	https://www.juniper.net/support
<hr/>	
“Reinstall Junos OS” on page 175	Insert the floppy and reboot the system.
<hr/>	
“Reconfigure Junos OS” on page 175	
1. Configure Host Names, Domain Names, and IP Addresses on page 175	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 177	<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 179	<code>ping address</code>
4. Copy Backup Configurations to the Router on page 179	<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 17: Checklist for Reinstalling Junos OS (continued)

Tasks	Command or Action
“After You Reinstall Junos OS” on page 183	
1. Compare Information Logged Before and After the Reinstall on page 183	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 184	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
ALARMD release 5.0R5 built by builder on 2002-03-02 04:43:24 UTC
PFED release 5.0R5 built by builder on 2002-03-02 04:46:06 UTC
CRAFTD release 5.0R5 built by builder on 2002-03-02 04:44:30 UTC
SAMPLED release 5.0R5 built by builder on 2002-03-02 04:52:20 UTC
ILMID release 5.0R5 built by builder on 2002-03-02 04:45:21 UTC
BPRELAYD release 5.0R5 built by builder on 2002-03-02 04:42:41 UTC
```

```
RMOPD release 5.0R5 built by builder on 2002-03-02 04:46:11 UTC
jkernel-dd release 5.0R5 built by builder on 2002-03-02 04:41:07 UTC
jroute-dd release 5.0R5 built by builder on 2002-03-02 04:41:21 UTC
jdocs-dd release 5.0R5 built by builder on 2002-03-02 04:39:11 UTC
```

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

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 175](#)
2. [Protecting Network Security by Configuring the Root Password on page 177](#)
3. [Check Network Connectivity on page 179](#)
4. [Copy Backup Configurations to the Router on page 179](#)

Configure Host Names, Domain Names, and IP Addresses

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
```

- See Also**
- *Accessing a Junos OS Device the First Time*
 - *Junos OS User Accounts Overview*
 - *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

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

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*
- *Junos OS User Accounts Overview*
- *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

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 183](#)
2. [Back Up the New Software on page 184](#)

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.

CHAPTER 6

Configuring Dual-Root Partitions on Switches

- [Configuring Dual-Root Partitions on page 187](#)
- [Dual-Root Partitioning Scheme on SRX Series Devices on page 190](#)
- [Example: Installing Junos OS on SRX Series Devices Using the Partition Option on page 196](#)
- [Reinstalling the Single-Root Partition on SRX Series Devices on page 200](#)

Configuring Dual-Root Partitions

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 188](#)
- [Automatic Fixing of Corrupted Primary Root Partition with the Automatic Snapshot Feature on page 188](#)

- [Earlier Partition Scheme \(Junos OS Release 10.4R2 and Earlier\)](#) on page 189
- [Understanding Upgrading or Downgrading Between Resilient Dual-Root Partition Releases and Earlier Releases](#) on page 190

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 18 on page 188](#).

Table 18: 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 19 on page 190](#).

Table 19: Earlier Partition Scheme

Slice 1		Slice 2		Slice 3	
s1a	s1f	s2a	s2f	s3d	s3e
/ (root Junos OS)		/var	(empty until initial software upgrade)	(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 18 on page 188](#).

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 394](#)

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 191](#)
- [Important Features of the Dual-Root Partitioning Scheme on page 192](#)
- [Understanding Automatic Recovery of the Primary Junos OS Image with Dual-Root Partitioning on page 193](#)
- [Understanding How the Primary Junos OS Image with Dual-Root Partitioning Recovers Devices on page 194](#)
- [Understanding How Junos OS Release 10.0 or Later Upgrades with Dual-Root Partitioning on page 196](#)

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 20 on page 191](#) provides information on the storage media available on SRX Series devices.

Table 20: 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 20: 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 144](#)
- 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 147](#)
- 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 61](#)
- [Example: Installing Junos OS on SRX Series Devices Using the Partition Option on page 196](#)

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 197](#)
- [Overview on page 197](#)
- [Configuration on page 198](#)
- [Verification on page 199](#)

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*.

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 139.

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 200](#)

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	<ul style="list-style-type: none">• Dual-Root Partitioning Scheme on SRX Series Devices on page 190• Reinstalling the Single-Root Partition on SRX Series Devices on page 200

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 190](#)
- [Example: Installing Junos OS on SRX Series Devices Using the Partition Option on page 196](#)

CHAPTER 7

Storage Media

- [Routing Engines and Storage Media on page 203](#)
- [System Memory and Storage Media on Routers on page 204](#)
- [System Memory and Storage Media for SRX Series Services Gateways on page 207](#)
- [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 210](#)
- [Storage Media Names for SRX Series Devices on page 212](#)
- [Repartitioning Routing Engine System Storage to Increase the Swap Partition on page 212](#)
- [Accessing USB Storage on PTX1000 Routers on page 213](#)

Routing Engines and Storage Media

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](#).

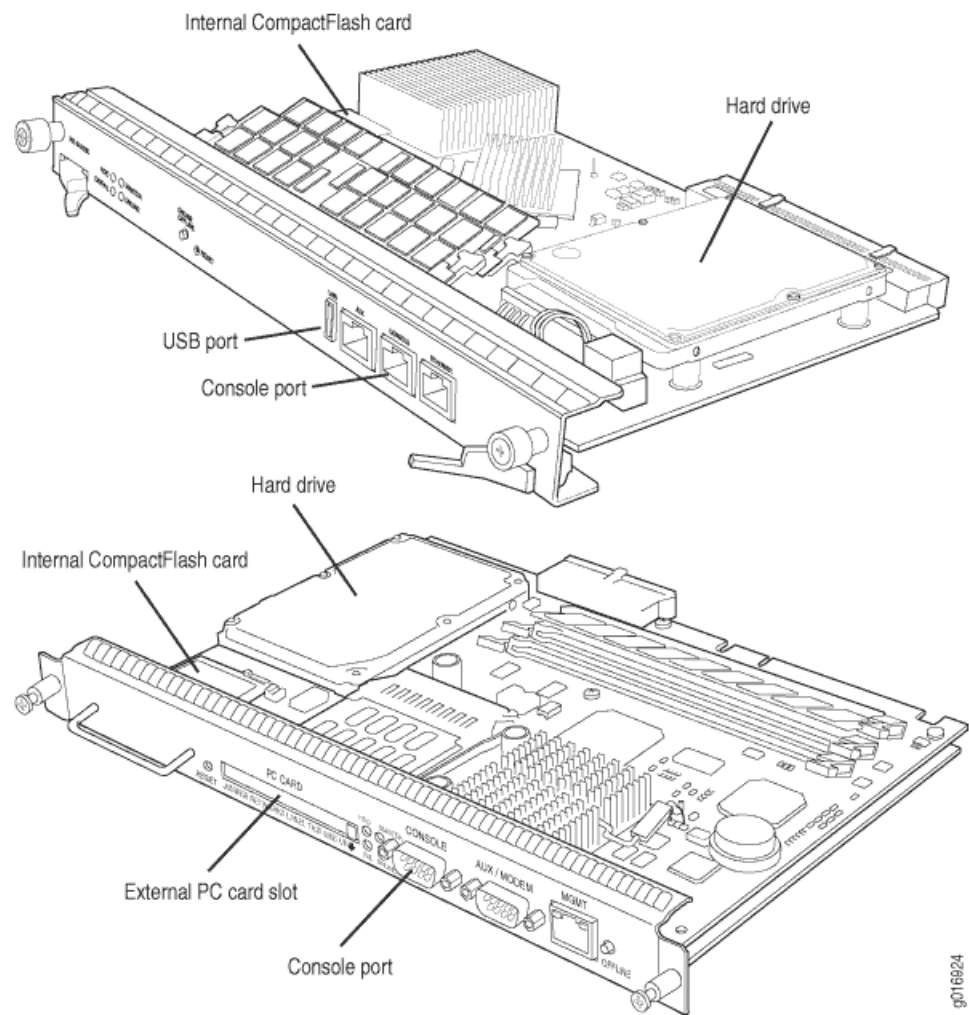
Related Documentation

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

System Memory and Storage Media on Routers

Figure 5 on page 205 shows examples of Routing Engines.

Figure 5: 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 205](#)
- [Storage Media on page 206](#)

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 Routing Engines with VM Host Support” on page 305 section.

System Memory and Storage Media for SRX Series Services Gateways

SRX Series Device Overview

Figure 6 on page 207 shows an example of SRX240 device.

Figure 6: SRX240 Device Front Panel

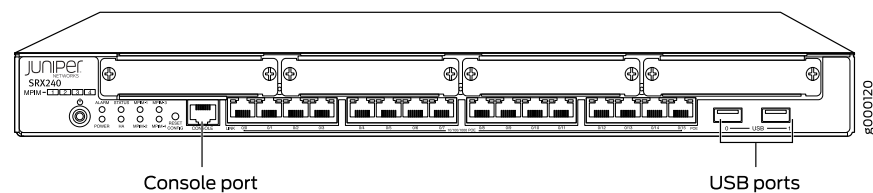


Figure 7 on page 207 shows an example of SRX650 device.

Figure 7: SRX650 Device System Routing Engine

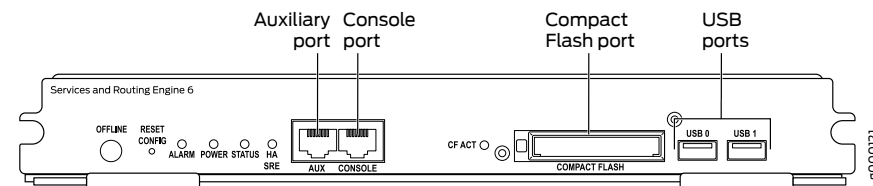


Figure 8 on page 207 shows the front panel of an SRX345 device.

Figure 8: SRX345 Device Front Panel

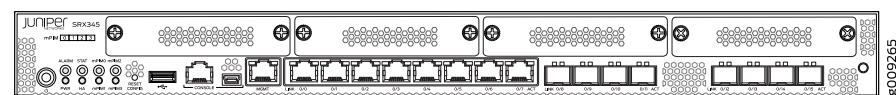


Figure 9 on page 207 shows an example of an SRX1500 device.

Figure 9: SRX1500 Device Front Panel

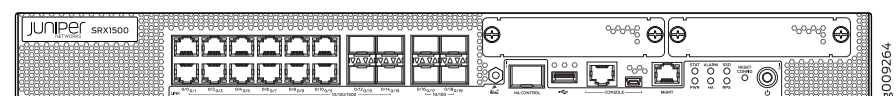


Figure 10 on page 208 shows an example of an SRX4200 device.

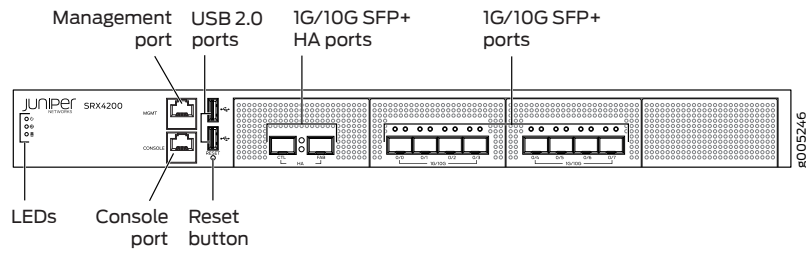
Figure 10: SRX4200 Services Gateway Front Panel

Figure 11 on page 208 shows an example of an SRX4600 device.

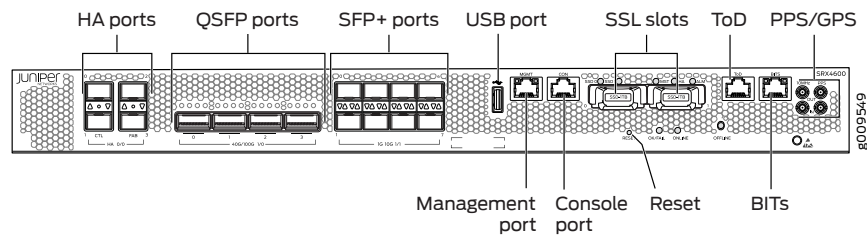
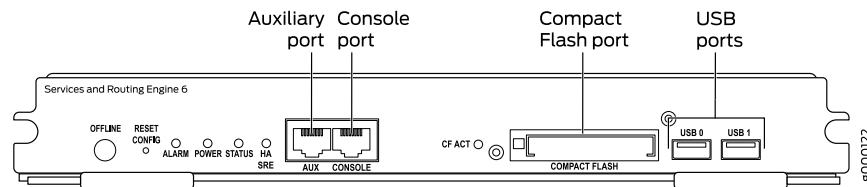
Figure 11: SRX4600 Services Gateway Front Panel

Figure 12 on page 208 shows an example of an SRX5800 device Routing Engine.

Figure 12: SRX5800 Device Routing Engine

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

The SRX4600 devices include the following storage media:

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

Related Documentation

- [Verifying PIC Combinations on page 386](#)

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 21 on page 210 specifies the storage media names by Routing Engine. The storage media device names are displayed when the router boots.

Table 21: 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
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

Table 21: 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-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
RE-PTX-X8-64G	No	No	Yes SSD1: sda SSD2: sdb	da0
RE-S-X6-64G	No	No	Yes SSD1: sda SSD2: sdb	da0
REMX2K-X8-64G	No	No	Yes SSD1: sda SSD2: sdb	da0



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.

- 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 22 on page 212 specifies the storage media names used by the SRX Series devices. The storage media device names are displayed as the device boots.

Table 22: 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 207](#)

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 88](#)
- [Installing the Software Package on a Router with Redundant Routing Engines on page 92](#)

Accessing USB Storage on PTX1000 Routers

On PTX1000 routers, you can only view the USB storage information from Junos OS by using the CLI command `show vmhost hardware`, but cannot access it. However, you can access the USB storage information from the Linux host. From the Linux host, you can also send the USB storage device information with images across different sites where PTX1000 routers are deployed.

To access the USB storage device information on PTX1000 routers:

1. In Junos OS, ensure that the PTX1000 USB image to be copied to the USB storage device is present on the `var/tmp` folder of Junos OS. To copy the image from the `/var/tmp` directory of Junos OS to the `/var/tmp` directory of a Linux host, execute the following command on Junos OS:

```
vhclient rcp /var/tmp image-name
vhclient -s
```

2. On the Linux host shell, execute the following command:

```
dd if=/var/tmp/ copied-image-name of=/dev/sdc bs=4M
sync
sync
```

In the command above, `/dev/sdc` is the USB storage device detected by the Linux host. You can determine the name of the USB storage device from host logs as shown in the sample below:

```
user@host # dmesg
...
[645888.884431] usb 1-1.2: new high-speed USB device number 5 using ehci-pci
[645889.131217] usb-storage 1-1.2:1.0: USB Mass Storage device detected
[645889.131275] scsi8 : usb-storage 1-1.2:1.0
[645890.134290] scsi 8:0:0:0: Direct-Access      JetFlash Transcend 8GB
```

```
8.07 PQ: 0 ANSI: 2
[645890.134456] sd 8:0:0:0: Attached scsi generic sg2 type 0

[645890.135908] sd 8:0:0:0: [sdc] 15687680 512-byte logical blocks: (8.03
GB/7.48 GiB)
```

In this example, **sdc** is the name of the USB storage device.



NOTE: The `/var/tmp` directory of a Linux host is mounted on the RAM (at the `ramfs` location), which is volatile storage, and is thus lost when you perform power cycling of or reboot the device. However, the Junos OS `/var/tmp` directory resides on the physical (nonvolatile) hard disk and thus exists even after rebooting or power cycling.

- Related Documentation**
- [Creating an Emergency Boot Device on page 223](#)
 - [Release Information for Junos OS with Upgraded FreeBSD on page 32](#)

CHAPTER 8

Performing a Recovery Installation

- [Creating a Snapshot and Using It to Boot an EX Series Switch on page 215](#)
- [Creating a Snapshot and Using It to Boot a QFX3500 and QFX3600 Series Switch on page 216](#)
- [Creating a Snapshot and Using It to Boot a Device on page 218](#)
- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 220](#)
- [Creating an Emergency Boot Device for Routers on page 223](#)
- [Creating an Emergency Boot Device for QFX Series Switches on page 224](#)
- [Performing a Recovery Installation Using an Emergency Boot Device on page 226](#)
- [Performing a Recovery Installation on page 228](#)
- [Recovering from a Failed Software Installation on page 229](#)
- [Installing and Recovering Software Using the Open Network Install Environment \(ONIE\) on page 231](#)
- [Understanding Integrity Check and Autorecovery of Configuration, Licenses, and Disk Information on SRX Series Devices on page 244](#)
- [Saving a Rescue Configuration File on page 246](#)
- [Restoring a Saved Configuration on page 247](#)
- [Reverting to the Default Factory Configuration by Using the request system zeroize Command on page 249](#)
- [Reverting to the Rescue Configuration on page 249](#)
- [Restarting and Halting SRX Series Devices on page 250](#)

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 216](#)

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
```

Creating a Snapshot and Using It to Boot a QFX3500 and QFX3600 Series Switch

The system snapshot feature takes a “snapshot” of the files currently used to run the QFX Series switch—the complete contents of the `/config` and `/var` directories, which include the running Juniper Networks Junos OS, the active configuration, and the rescue configuration—and copies all of these files into an alternate (internal, meaning internal flash, or an external, meaning USB flash) memory source. You can then use these snapshots to boot the switch at the next bootup or as a backup boot option.

The system snapshot feature is especially effective as a bootup option after a partition corruption, as it is the only recovery option that allows you to completely restore the Junos OS and configuration in the event of a corrupted partition.

This topic includes the following tasks:

- [Creating a Snapshot on a USB Flash Drive and Using It to Boot the Switch on page 216](#)
- [Creating a Snapshot on an Internal Flash Drive and Using it to Boot the Switch on page 217](#)
- [Creating a Snapshot on the Alternate Slice of the Boot Media on page 218](#)

Creating a Snapshot on a USB Flash Drive and Using It to Boot the Switch



NOTE: Creating a snapshot is not supported on QFX10000 switches.

A snapshot can be created on USB flash memory after a switch is booted 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 QFX Series switch USB port specifications. See *USB Port Specifications for the QFX Series*.

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
```



NOTE: This example uses the **partition** option. If you have already created a partition for the snapshot, you don't need to use the **partition** option.

2. (Optional) Perform this step if you want to boot the switch now using the snapshot stored on the external USB flash drive. If you created the snapshot as a backup, do not perform this step.

- To reboot the switch using the most recently created snapshot:

```
user@switch> request system reboot
```

- To reboot the switch using a snapshot in a specific partition on the USB flash drive:

```
user@switch> request system reboot slice 1
```

Creating a Snapshot on an Internal Flash Drive and Using it to Boot the Switch

A snapshot can be created on internal memory after a switch is booted using files stored in external memory.

To create a snapshot in internal memory and use it to boot the switch:

1. Place the snapshot files in internal memory:

```
user@switch> request system snapshot partition
```



NOTE: This example uses the **partition** option. If you have already created a partition for the snapshot, you don't need to use the **partition** option.

2. (Optional) Perform this step if you want to boot the switch now using the newly created snapshot. If you created the snapshot as a backup, do not perform this step.

- To reboot the switch using the most recently created snapshot:

```
user@switch> request system reboot
```

- To reboot the switch using a snapshot in a specific partition in internal memory:

```
user@switch> request system reboot slice 1
```

Creating a Snapshot on the Alternate Slice of the Boot Media

The alternate slice of the boot media contains a backup software image that the switch can boot from if it is unable to boot from the primary slice. When you upgrade software, the new software image gets copied only to the primary slice of the boot media.

To create a snapshot of the currently booted software image on the backup slice of the boot media:

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

After the system boots up, you will see the following message before the login prompt:

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 up from the backup copy.

Please re-install JUNOS to recover the primary copy in case it has been corrupted.

The system will generate an alarm indicating that the switch has booted from the backup slice.

Creating a Snapshot and Using It to Boot a Device

The system snapshot feature takes a “snapshot” of the files currently used to run the device—the complete contents of the **/config** directories, which include the running Juniper Networks Junos OS, the active configuration, and the rescue configuration, as well as the host OS—and copies all of these files into an external USB flash drive.

You can use the snapshot to boot the device at the next bootup or as a backup boot option.

The system snapshot feature is especially effective as a bootup option after a partition corruption, as it is the only recovery option that allows you to completely restore the Junos OS and configuration in the event of a corrupted partition on a switch.



NOTE: System snapshot is not supported on QFX5200 and QFX10000 switches.

This topic includes the following tasks:

- [Creating a Snapshot on an External USB Flash Drive and Using It to Boot the Device on page 218](#)

Creating a Snapshot on an External USB Flash Drive and Using It to Boot the Device

A snapshot can be created on an external USB flash drive after a device is booted using files stored in internal memory.

Ensure that you have the following tools and parts available before creating a snapshot on an external USB flash drive:

- An external USB flash drive that meets the device USB port specifications. See *USB Port Specifications for the QFX Series*.

To create a snapshot on the external USB flash drive and use it to boot the device:

1. Insert the external USB flash drive.
2. Issue the **request system snapshot** command.

```
user@device> request system snapshot
fpc0:
```

```
-----
Starting snapshot to usb (/dev/da0)
Creating snapshot on the host ..
Copying bootable disk image from host ..
Writing to usb (/dev/da0) ..
Copying 'Host OS' to '/dev/da0s1' .. (this may take a few minutes)
Copying 'JUNOS' to '/dev/da0s1' .. (this may take a few minutes)
The following filesystems were archived: / /config Host-OS
```

3. (Optional) Perform this step if you want to boot the device now using the snapshot stored on the external USB flash drive. If you created the snapshot as a backup, do not perform this step.

- Insert the external USB flash drive.

- Power cycle the device.

The external USB flash drive is detected.

- The software prompts you with the following options:

```
Junos Snapshot Installer - (c) Juniper Networks 2013
Reboot
Install Junos Snapshot [13.2-20131115_x_132_x51_vjunos.0
Boot to host shell [debug]
```

- Select **Install Junos Snapshot** to install the snapshot located on the external USB flash drive to the device.

The device copies the software from the external USB flash drive, occasionally displaying status messages. When the software is finished being copied from the external USB flash drive to the device, the device then reboots from the internal flash storage on which the software was just installed. When the reboot is complete, the device displays the Junos OS login prompt:

```
root@device#
```

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 220](#)
- [Overview on page 220](#)
- [Configuration on page 220](#)
- [Creating a Snapshot on a USB Flash Drive and Using It to Boot the SRX Series Device on page 222](#)
- [Verification on page 222](#)

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 74](#).

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 222](#)

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.

Creating an Emergency Boot Device for Routers

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 [“Release Information for Junos OS with Upgraded FreeBSD” on page 32.](#))

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 210.
- **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 “Release Information for Junos OS with Upgraded FreeBSD” on page 32.)

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 210
- [Release Information for Junos OS with Upgraded FreeBSD](#) on page 32
- [Accessing USB Storage on PTX1000 Routers](#) on page 213

Creating an Emergency Boot Device for QFX Series Switches

If Junos OS on the device is damaged in some way that prevents the software from loading properly, you can use an emergency boot device to repartition the primary disk and load a fresh installation of Junos OS. Use the following procedure to create an emergency boot device.

Before you begin, you need to download the installation media image for your device and Junos OS release from <https://www.juniper.net/customers/support/>.



NOTE: You can create the emergency boot device on another Juniper Networks switch or router, or any PC or laptop that supports Linux. The steps you take to create the emergency boot device vary, depending on the device.

To create an emergency boot device:

1. Use FTP to copy the installation media image into the `/var/tmp` directory on the device.
2. Insert a USB device into the USB port.

3. From the Junos OS command-line interface (CLI), start the shell:

```
user@device> start shell
%
```

4. Switch to the root account using the **su** command:

```
% su
Password: password
```



NOTE: The password is the root password for the device. If you logged in to the device as root, you do not need to perform this step.

5. Enter the following command on the device:

```
root@device% dd if=/var/tmp/filename of=/dev/dal bs=1m
```

The device writes the installation media image to the USB device:

```
root@device% dd if=install-media-qfx-5e-15.1X53-D30.5-domestic.img of=/dev/da0
bs=1m
1399+0 records in
1399+0 records out
1466957824 bytes transferred in 394.081902 secs (3722469 bytes/sec)
```

6. Log out of the shell:

```
root@device% exit
% exit
user@device>
```

Related Documentation

- *USB Port Specifications for the QFX Series*
- *Performing a Recovery Installation*
- [Performing a Recovery Installation Using an Emergency Boot Device on page 226](#)

Performing a Recovery Installation Using an Emergency Boot Device

If Junos OS on your device is damaged in some way that prevents the software from loading correctly, you may need to perform a recovery installation using an emergency boot device (for example, a USB flash drive) to restore the default factory installation. Once you have recovered the software, you need to restore the device configuration. You can either create a new configuration as you did when the device was shipped from the factory, or if you saved the previous configuration, you can simply restore that file to the device.

If at all possible, you should try to perform the following steps before you perform the recovery installation:

1. Ensure that you have an emergency boot device to use during the installation. See [“Creating an Emergency Boot Device” on page 224](#) for information on how to create an emergency boot device.
2. Copy the existing configuration in the file `/config/juniper.conf.gz` from the device to a remote system, such as a server, or to an emergency boot device. 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) to a remote system or to an emergency boot device.

You can use the system snapshot feature to complete this step. The system snapshot feature takes a “snapshot” of the files currently used to run the QFX Series switch—the complete contents of the `/config` and `/var` directories, which include the running Juniper Networks Junos OS, the active configuration, and the rescue configuration—and copies all of these files into a memory source. See [“Creating a Snapshot and Using It to Boot a Device” on page 218](#).



NOTE: System snapshot is not supported on QFX10000 and QFX5200 switches.



WARNING: The recovery installation process completely overwrites the entire contents of the internal flash storage.

3. Copy any other stored files to a remote system as desired.

To reinstall Junos OS:

1. Insert the emergency boot device into the device.
2. Power cycle the device.

The emergency boot device is detected. At this time, you can load the Junos OS from the emergency boot device onto the internal flash storage.

3. The software prompts you with the following option if you have a snapshot saved on the emergency boot device:

```
Junos Snapshot Installer - (c) Juniper Networks 2013
Reboot
Install Junos Snapshot [14.1X53-D11_vjunos.61]
Boot to host shell [debug]
```

Select **Install Junos Snapshot** to install the snapshot.

The software prompts you with the following option if you have Junos OS software from the factory installed on the emergency boot device.

```
Juniper Linux Installer - (c) Juniper Networks 2014
Reboot
Install Juniper Linux Platform
Boot to host shell [debug]
```

Select **Install Juniper Linux Platform** to install the Junos OS software from the emergency boot device.

4. The device copies the software from the emergency boot device, occasionally displaying status messages. Copying the software can take up to 12 minutes.

When the software is finished being copied from the emergency device to the device, the device reboots from the internal flash storage on which the software was just installed. When the reboot is complete, the device displays the Junos OS login prompt:

```
root@switch#
```

5. Create a new configuration as you did when the device was shipped from the factory, or restore the previously saved configuration file to the device.
6. Remove the emergency boot device.

Related Documentation

- [Creating an Emergency Boot Device on page 224](#)

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 223](#)
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 ["Release Information for Junos OS with Upgraded FreeBSD" on page 32](#).

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*, *Creating a New Configuration with Redundant Routing Engines*, and ["Restoring a Saved Configuration" on page 247](#).

Recovering from a Failed Software Installation

Problem **Description:** If the Junos OS appears to have been installed but the CLI does not work, or if the switch has no software installed, you can use this recovery installation procedure to install the Junos OS.

Solution If a Junos OS image already exists on the switch, you can either install the new Junos OS package in a separate partition, in which case both Junos OS images remain on the switch, or you can remove the existing Junos OS image before you start the new installation process.



NOTE: QFX5100, QFX5200, EX4600, QFX10000, and OCX Series switches do not have a separate partition to reinstall a Junos OS image.

A recovery image is created automatically on these switches. If a previously-running switch is powered on and unable to boot using a Junos OS image, you can boot the switch using the recovery Junos OS image by selecting an option in the “Select a recovery image” menu.

We suggest creating a system snapshot on your switch onto the external USB flash drive, and using the snapshot for recovery purposes. The system snapshot feature takes a “snapshot” of the files currently used to run the device—the complete contents of the /config directories, which include the running Juniper Networks Junos OS, the active configuration, and the rescue configuration, as well as the host OS—and copies all of these files into an external USB flash drive. See [“Creating a Snapshot and Using It to Boot a QFX3500 and QFX3600 Switch” on page 216](#) or [“Creating a Snapshot and Using It to Boot a Device” on page 218](#).

System snapshot is not supported on QFX5200 and QFX10000 switches.

To perform a recovery installation:

1. Power on the switch. The loader script starts.
2. After the message **Loading /boot/defaults/loader.conf** appears, you are prompted with the following message:

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

Press the Spacebar to enter the manual loader. The **loader>** prompt appears.



NOTE: The loader prompt does not appear on QFX5100, QFX5200, EX4600, QFX10000, and OCX Series switches.

On QFX5100, QFX5200, EX4600, QFX10000, and OCX Series switches only, a recovery image is automatically saved if a previously-running switch is powered on and unable to boot using a Junos OS image.

The “Select a recovery image” menu appears on the console when one of these switches is booted and unable to load a version of Junos OS. Follow the instructions in the “Select a recovery image” menu to load the recovery version of Junos OS for one of these switches.

You can ignore the remainder of this procedure if you are using a QFX5100, QFX5200, EX4600, QFX10000, or OCX Series switch.

3. Enter the following command:

```
loader> install [- --format] [- --external] source
```

where:

- **format**—Enables you to erase the installation media before installing the installation package. If you do not include this option, the system installs the new Junos OS in a different partition from that of the most recently installed Junos OS.
- **external**—Installs the installation package onto external media (a USB stick, for example).
- **source**—Represents the name and location of the Junos OS package, either on a server on the network or as a file on an external media, as shown in the following two examples:
 - Network address of the server and the path on the server; for example, `ftp://192.0.2.0/junos/jinstall-qfx-5e-flex-15.1X53-D30.5-domestic-signed.tgz`
 - Junos OS package on a USB device (commonly stored in the root drive as the only file), for example, `file:///jinstall-qfx-5e-flex-15.1X53-D30.5-domestic-signed.tgz`.

The installation now proceeds normally and ends with a login prompt.

Related Documentation

- [Creating a Snapshot and Using It to Boot a QFX3500 and QFX3600 Switch on page 216](#)
- [Creating a Snapshot and Using It to Boot a Device on page 218](#)

Installing and Recovering Software Using the Open Network Install Environment (ONIE)

ONIE, the open network install environment from Cumulus Networks, is a network OS installer that installs Junos OS and third party applications on a switch. Juniper Network switches come preinstalled with ONIE. When you turn on a switch, the ONIE discovery and execution (ODE) application locates the management Ethernet interface and the Junos OS software package, which can be found either locally on the switch or on the network using HTTP, FTP, or TFTP. After the switch discovers and downloads the Junos OS software package, the switch installs the Junos OS software, reboots, and then boots from Junos OS. Junos OS then becomes the default software image.



NOTE: If you want to use the Junos OS CLI to install software, see [“Upgrading Software” on page 79](#).

Upgrading involves these tasks:

- [Understanding the Open Network Install Environment on page 233](#)
- [Downloading Software Files with a Browser on page 233](#)

- [Connecting to the Console Port on page 234](#)
- [Backing Up the Current Configuration Files on page 234](#)
- [Uninstalling the Existing Version of Junos OS on page 234](#)
- [Installing a Junos OS Software Package That Resides on a Webserver or DHCP Server with DHCP Options Configured on page 235](#)
- [Installing Junos OS Software Using Secure Copy Protocol \(SCP\) on page 236](#)
- [Installing Junos OS Software Using FTP or TFTP Without a Webserver on page 237](#)
- [Installing Junos OS Software Using DHCP Server with No DHCP Options Configured on page 238](#)
- [Installing Junos OS Software Using Webserver Without DHCP Configured on page 239](#)
- [Installing Junos OS Software Using USB Media on page 240](#)
- [Verifying Software Installation on page 241](#)
- [Troubleshooting Boot Problems on page 241](#)
- [Creating an Emergency Boot Device on page 242](#)
- [Performing a Recovery Installation on page 243](#)

Understanding the Open Network Install Environment

When you log into the switch with ONIE, you see the install boot menu:

- Juniper Linux (This is a default menu option.)
- Juniper Linux Debug
- Juniper Linux Recovery
- Go to ONIE Loader
 - ONIE: Install OS (This is a default menu option.)
 - ONIE: Rescue
 - ONIE: Uninstall OS
 - ONIE: Update ONIE
 - ONIE: Embed ONIE

You can use the following commands to install and uninstall Junos OS and start and stop the ONIE ODE application:

- **onie-nos-install**

Installs Junos OS from any URL, such as `http://`, `ftp://`, and `file://`.

- **onie-uninstaller**

Uninstalls Junos OS.

- **onie-discovery-start**

The discovery process starts automatically. However, if you stop the discovery process by issuing the **onie-discovery-stop** command, you can restart the discovery process by issuing the **onie-discovery-start** command.

- **onie-discovery-stop**

Stops the discovery process. To restart the discovery process, issue the **onie-discovery-start** command.

Downloading Software Files with a Browser

To download the software package from the Juniper Networks Support website at <https://www.juniper.net/support/>.



NOTE: To access the download site, you must have a service contract with Juniper Networks 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 a software package:

1. Using a Web browser, navigate to the <https://www.juniper.net/support>.
2. Click **Download Software**.
3. In the By Technology box, click **Switching > QFX Series > QFX10002**.
4. Click the **Software** tab and select the install package from the Install Package box.
A login screen appears.
5. Enter your name and password and press Enter.
6. Read the End User License Agreement, click the **I agree** radio button, and then click **Proceed**.
7. Save the **jinstall-qfx-10-f-15.1X53-D10.7-domestic-signed** file to your computer.
8. Open or save the installation package either to the local system in the **var/tmp** directory or to a remote location. If you are copying the installation package to a remote system, make sure that you can access it using HTTP, TFTP, FTP, or SCP.

Connecting to the Console Port

We recommend that you connect to the console port while installing the installation package so you can respond to any required user input and detect any errors that might occur.

Backing Up the Current Configuration Files

Before you install the new installation package, we strongly recommend that you back up your current configuration files because the upgrade process removes all of the stored files on the switch.

To back up your current configuration files, enter the **save** command:

```
user@switch# save filename
```

Executing this command saves a copy of your configuration files to a remote location such as an external USB device.

Uninstalling the Existing Version of Junos OS

The switch comes preinstalled with a version of Junos OS that is to be used with the Junos OS CLI. However, if you want to use ONIE to install Junos OS, you need to uninstall the existing Junos OS and reinstall the Junos OS image that has a .bin extension—for example, **jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin** file.

To uninstall your existing Junos OS version:

1. Select **Go to ONIE Loader** from the GNU GRUB menu.
2. Select **ONIE: Uninstall OS** from the GNU GRUB menu.

The Junos OS is uninstalled, and the switch reboots.

By default, the ONIE discovery and execution (ODE) application attempts to discover and fetch an image from a configured DHCP or webserver and the management IP address of the switch and the IP address of the default gateway. If you want to manually configure static addressing for the management IP address of the switch, issue **onie-discovery-stop** command at the ONIE prompt, and then manually configure the management IP address and IP address of the default gateway.

For example:

```
ONIE:/ # onie-discovery-stop
ONIE:/ # ifconfig eth0 10.204.32.96 netmask 255.255.254.0
ONIE:/ # route add default gw 10.204.47.254
```

To restart the ONIE discovery and execution (ODE) application, issue the **onie-discovery-start** command.

For example:

```
ONIE:/ # onie-discovery-start
```

Installing a Junos OS Software Package That Resides on a Webserver or DHCP Server with DHCP Options Configured

To install a Junos OS software package residing on a webserver or DHCP server:

1. Copy the software image with the filename **onie-installer** to the **var/www/html** directory of the webserver or DHCP server.
2. Configure the DHCP option 114 in the DHCP server to redirect to the webserver to fetch the Junos OS software image.
3. Uninstall the preinstalled Junos OS version.
 - Select **Go to ONIE Loader** from the GNU GRUB menu.
 - Select **ONIE: Uninstall OS** from the GNU GRUB menu.

The Junos OS is uninstalled, and the switch reboots.

4. Configure DHCP option 114 and other DHCP options as necessary.

Here is a sample Windows Open DHCP server configuration with DHCP option 114 configured.

```
#Following are range-specific DHCP options.  
#You can copy more option names from [GLOBAL_OPTIONS]  
IP=10.204.42.250  
SubnetMask=255.255.240.0  
Router=10.204.47.254  
114="http://10.207.66.147/onie-installer"
```

Here is a sample boot initialization log, showing the options you just configured:

```
Info: Trying DHCPv4 on interface: eth0  
ONIE: Using DHCPv4 addr: eth0: 10.204.42.250 / 255.255.240.0  
ONIE: Starting ONIE Service Discovery  
Info: Fetching http://10.207.66.147/onie-installer ...  
ONIE: Executing installer: http://10.207.66.147/onie-installer <-----  
automatically redirects to web sever to fetch Junos OS image.  
Verifying image checksum ... OK.  
Preparing image archive ... OK.  
Installing Juniper NOS...
```

The log shows that the installation process has fetched the Junos OS software image from the DHCP server and is installing the Junos OS software.

The switch reboots and the GNU GRUB menu is displayed.

Installing Junos OS Software Using Secure Copy Protocol (SCP)

To install Junos OS software using SCP:

1. Uninstall the preinstalled Junos OS version.
 - Select **Go to ONIE Loader** from the GNU GRUB menu.
 - Select **ONIE: Uninstall OS** from the GNU GRUB menu.

The Junos OS is uninstalled, and the switch reboots.

By default, the ONIE discovery and execution (ODE) application attempts to discover and fetch an image from a configured webserver. If you do not have DHCP configured, you will need to stop the ONIE discovery and execution (ODE) application and manually configure static addressing for the management IP address of the switch,

For example:

```
ONIE:/ # onie-discovery-stop  
ONIE:/ # ifconfig eth0 10.204.32.96 netmask 255.255.254.0  
ONIE:/ # route add default gw 10.204.47.254
```

2. Use SCP to copy the Junos OS image from a server or other location to the `/var/tmp` directory on the switch.

For example:

```
user@server scp jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin  
root@10.204.32.196:/var/tmp/
```

3. Issue the **onie-nos-install** command in the `/var/tmp` directory to install Junos OS software.

```
ONIE:/var/tmp # onie-nos-install
file:///var/tmp/jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin
```

The switch reboots and displays the GNU GRUB menu.

Installing Junos OS Software Using FTP or TFTP Without a Webserver

To install Junos OS software using FTP or TFTP:

1. Uninstall the preinstalled Junos OS version.
 - Select **Go to ONIE Loader** from the GNU GRUB menu.
 - Select **ONIE: Uninstall OS** from the GNU GRUB menu.

The Junos OS is uninstalled, and the switch reboots.

By default, the ONIE discovery and execution (ODE) application attempts to discover and fetch an image from a configured webserver. If you do not have DHCP configured, you will need to stop the ONIE discovery and execution (ODE) application and manually configure static addressing for the management IP address of the switch,

For example:

```
ONIE:/ # onie-discovery-stop
ONIE:/ # ifconfig eth0 10.204.32.96 netmask 255.255.254.0
ONIE:/ # route add default gw 10.204.47.254
```

2. Copy the Junos OS image to an FTP or TFTP directory.
3. Issue the **onie-nos-install** command at the ONIE prompt to install the Junos OS software.

If you are using FTP:

```
ONIE:/ # onie-nos-install
ftp://<username>:<password>@10.209.152.22/jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin
```

If you are using TFTP:



NOTE: The software image should be located in the `/tftp/boot` directory.

```
ONIE:/ # onie-nos-install
tftp://10.207.66.147/jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin
```

The switch reboots and displays the GNU GRUB menu.

Installing Junos OS Software Using DHCP Server with No DHCP Options Configured

Use this installation method if you cannot modify or set the DHCP options on your DHCP server.

To install the Junos OS software using a DHCP server with no DHCP options configured:

1. Copy the software image with the filename **jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin** to the **var/www/html** directory of the webserver or DHCP server.
2. Uninstall the preinstalled Junos OS version.
 - Select **Go to ONIE Loader** from the GNU GRUB menu.
 - Select **ONIE: Uninstall OS** from the GNU GRUB menu.

The Junos OS is uninstalled, and the switch reboots.

3. Issue the **onie-nos-install** command at the ONIE prompt to install the Junos OS software.

For example:

```
ONIE:/ # onie-nos-install
http://10.207.66.147/jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin
```

Here is sample log with the options you just configured:

```
ONIE:/ # ifconfig
eth0      Link encap:Ethernet  HWaddr 94:DE:80:AA:F2:E1
          inet addr:10.204.42.250  Bcast:10.204.47.255  Mask:255.255.240.0
<<<---- --> Received IP address from DHCP server, but auto redirected to web
server. Installation will not happen because DHCP option (114) is not
configured.

          inet6 addr: fe80::96de:80ff:feaa:f2e1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:444 errors:0 dropped:0 overruns:0 frame:0
          TX packets:17 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:48170 (47.0 KiB)  TX bytes:2678 (2.6 KiB)
          Memory:80180000-801a0000
```

This log shows that the installation process has fetched the Junos OS software image from the webserver and is installing the Junos OS software.

```
Stopping: discover... done.
Info: Fetching
http://10.207.66.147/jinstall-qfx-10-f-15.1X53-D10.7-domestic-signed.tgz ...
Connecting to 10.207.66.147 (10.207.66.147:80)
installer          100% |*****| 464M 0:00:00
ETA
ONIE: Executing installer:
http://10.207.66.147/jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin
```

```
Verifying image checksum ... OK.
Preparing image archive ... OK.
Installing Juniper NOS...
```

The switch reboots and the GNU GRUB menu is displayed.

Installing Junos OS Software Using Webserver Without DHCP Configured

Use this installation method if you do not have a DHCP server.

To install the Junos OS software using a webserver without DHCP configured:

1. Because the switch comes preinstalled with the Junos OS to be used with the Junos OS CLI, you need to uninstall this version of software before you can install the Junos OS image to be used with ONIE.

- Select **Go to ONIE Loader** from the GNU GRUB menu.
- Select **ONIE: Uninstall OS** from the GNU GRUB menu.

The Junos OS is uninstalled, and the switch reboots.

2. Select **ONIE: Uninstall OS** from the GNU GRUB menu.

The Junos OS is uninstalled, and the switch reboots.

By default, the ONIE discovery and execution (ODE) application attempts to discover and fetch an image from a configured webserver. Because you do not have DHCP configured, you will need to stop the ONIE discovery and execution (ODE) application and manually configure static addressing for the management IP address of the switch.

For example:

```
ONIE:/ # onie-discovery-stop
ONIE:/ # ifconfig eth0 10.204.32.96 netmask 255.255.254.0
ONIE:/ # route add default gw 10.204.47.254
```

3. Copy the software image to the **var/www/html** directory of the webserver.
4. Issue the **onie-nos-install** command at the ONIE prompt to install the Junos OS software.

For example:

```
ONIE:/ # onie-nos-install
http://10.204.35.100/jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin
```

Here is sample log:

```
Stopping: discover... done.
Info: Fetching
http://10.204.35.100/jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin ...
Connecting to 10.204.35.100 (10.204.35.100:80)
installer          100% | ***** | 464M  0:00:00
ETA
```

```
ONIE: Executing installer:
http://10.204.35.100/jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin
Verifying image checksum ... OK.
Preparing image archive ... OK.
Installing Juniper NOS...
```

The log shows that the installation process has fetched the Junos OS software image from the webserver and is installing the Junos OS software.

The switch reboots and the GNU GRUB menu is displayed.

Installing Junos OS Software Using USB Media

To install the Junos OS software using USB media:

1. Make sure the USB stick is formatted as VFAT or ext2.
2. Copy the software image with the filename **onie-installer** to the USB stick and insert the USB stick into the front USB port of the switch.
3. Uninstall the preinstalled Junos OS version.
 - Select **Go to ONIE Loader** from the GNU GRUB menu.
 - Select **ONIE: Uninstall OS** from the GNU GRUB menu.

The Junos OS is uninstalled, the switch reboots, and the software is installed from the USB stick.

Here is a sample log showing the USB installation process:

```
Version : master-201412161452.0.1
Build Date: 2014-12-16T14:56+0800
Info: Mounting kernel filesystems... done.
Info: Mounting LABEL=ONIE-BOOT on /mnt/onie-boot ...
Info: Using eth0 MAC address: 94:de:80:aa:f2:e1
Info: eth0: Checking link... scsi 6:0:0:0: Direct-Access SanDisk Ultra
2.01 P6
sd 6:0:0:0: [sdb] 31266816 512-byte logical blocks: (16.0 GB/14.9 GiB)
sd 6:0:0:0: [sdb] Write Protect is off
sd 6:0:0:0: [sdb] Write cache: disabled, read cache: enabled, doesn't support
DPO or FUA
sd 6:0:0:0: [sdb] Attached SCSI disk
up.
Info: Trying DHCPv4 on interface: eth0
ONIE: Using DHCPv4 addr: eth0: 10.204.42.250 / 255.255.240.0
Starting: dropbear ssh daemon... done.
Starting: telnetd... done.
discover: installer mode detected. Running installer.
Starting: discover... done.

Please press Enter to activate this console. Info: eth0: Checking link... up.
Info: Trying DHCPv4 on interface: eth0
ONIE: Using DHCPv4 addr: eth0: 10.204.42.250 / 255.255.240.0
ONIE: Starting ONIE Service Discovery
ONIE: Executing installer: file:///dev/sdb/onie-installer <<<<--- Installing
from external(dev/sdb) attached USB.
Verifying image checksum ... OK.
```


Verifying Software Installation

Action To verify that the software was properly installed, issue the **show version** command.

Troubleshooting Boot Problems

241

From the output, you can see that the vendor-specific information confirms that it is for Juniper Networks.

Creating an Emergency Boot Device

If the Open Network Install Environment (ONIE) software is damaged or corrupted in some way, or the switch went into rescue mode, you can use an emergency boot device to repartition the primary disk and load a fresh installation of ONIE. Use the following procedure to create an emergency boot device.

Before you begin, you need to have the `jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin` version of ONIE software.



NOTE: In the following procedure, we assume that you are creating the emergency boot device on a switch. You can create the emergency boot device on any PC or laptop that supports Linux.

To create an emergency boot device:

1. Insert the USB device into the front USB port of the switch.

Make sure the USB device is at least 1GB.

2. Issue the following command from the directory on the switch in which the ISO file is located:

```
ONIE:/ # jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin of=<usb-detected-drive>  
bs=1M
```

You can also issue the **dd** command using the full path to where the ISO file is located.

For example, if the ISO file is located in the `/var/tmp/` directory:

```
ONIE:/ # dd if=/var/tmp/jnpr-qfx-5e-jdm-onie-updater-15.1-20150819_ups.4.bin  
of=<usb-detected-drive> bs=1M
```

The switch writes the installation media image to the USB device:

3. Remove the USB device from the USB port of the switch.

Performing a Recovery Installation

In the event that the Open Network Install Environment (ONIE) is corrupted, the switch goes into rescue mode, or you need to reinstall ONIE software for any reason, you need to perform a recovery installation.



NOTE: All Junos OS partitions are destroyed during a recovery installation.



NOTE: Before you can perform a recovery installation, make sure you have an emergency boot device loaded with ONIE software.

1. Insert the emergency boot device into the device.
2. Power cycle the device.
3. Press the **ESC** button to go into the Boot Manager menu.
4. Select **Boot Manager**, and then press **Enter**.
5. Select **Unigen PQS1000** under **Legacy USB**, and then press **Enter**.
6. Select **ONIE: Embed ONIE** from the **ONIE Installer** menu, and then press **Enter**.

The recovery installation proceeds using the emergency boot device.

7. Remove the emergency boot device.
8. Verify that the ONIE software was installed by looking at the installation log file.

For example:

```
Info: Found static url: file:///lib/onie/onie-updater
ONIE: Executing installer: file:///lib/onie/onie-updater
Verifying image checksum ... OK.
Preparing image archive ... OK.
ONIE: Version          : master-201412161452.0.1
```

Installation log files are displayed automatically during the installation process, but if you want to verify installation log files at a different time, you can find them in the `/var/log/` directory. To view an installation log file, issue the `tail -f /var/log/onie.log` command.

9. Issue the `parted /dev/sda print` command to verify that the ONIE partitions have been created.

For example:

```
ONIE:/ # parted /dev/sda print
Model: ATA TS8GHSD630 (scsi)
Disk /dev/sda: 8012MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:
```

Number	Start	End	Size	File system	Name	Flags
1	1049kB	3146kB	2097kB		GRUB-B00T	hidden, bios_grub
2	3146kB	137MB	134MB	ext4	ONIE-B00T	hidden

- Related Documentation**
- [Upgrading Software on page 79](#)
 - [Upgrading Software by Using Automatic Software Download](#)
 - [Configuring a DHCP Server on Switches \(CLI Procedure\)](#)

Understanding Integrity Check and Autorecovery of Configuration, Licenses, and Disk Information on SRX Series Devices

This topic includes the following sections:

- [Overview on page 244](#)
- [How Autorecovery Works on page 244](#)
- [How to Use Autorecovery on page 245](#)
- [Data That Is Backed Up in an Autorecovery on page 245](#)
- [Troubleshooting Alarms on page 245](#)
- [Considerations on page 246](#)

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 23 on page 245 lists types of autorecovery alarms, descriptions, and required actions.

Table 23: 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.

Table 23: Autorecovery Alarms (continued)

Alarm	Alarm Type	Description	Action Required
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.
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 220](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 120](#)
- [Reverting the Junos OS Software Image Back to the Previous Version on page 123](#)

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 247](#)
2. [Loading and Committing the Configuration File on page 248](#)

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 525](#)

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](#)

Restarting and Halting SRX Series Devices

This topic includes the following sections:

- [Rebooting SRX Series Devices on page 250](#)
- [Halting SRX Series Devices on page 252](#)
- [Bringing Chassis Components Online and Offline on SRX Series Devices on page 254](#)
- [Restarting the Chassis on SRX Series Devices on page 255](#)

Rebooting SRX Series Devices

This example shows how to reboot a SRX Series device.

- [Requirements on page 250](#)
- [Overview on page 250](#)
- [Configuration on page 250](#)
- [Verification on page 251](#)

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*.

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 252](#)

Verifying the Device Reboot

Purpose Verify that the device rebooted.

Action From operational mode, enter the **show system** command.

See Also

- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 220](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 120](#)
- [Reverting the Junos OS Software Image Back to the Previous Version on page 123](#)

Halting SRX Series Devices

This example shows how to halt a device.

- [Requirements on page 252](#)
- [Overview on page 252](#)
- [Configuration on page 252](#)
- [Verification on page 253](#)

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*.

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 254](#)

Verifying the Device Halt

Purpose Verify that the device halted.

Action From operational mode, enter the **show system** command.

See Also

- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 220](#)
- [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 120](#)
- [Reverting the Junos OS Software Image Back to the Previous Version on page 123](#)

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
```

- See Also**
- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 220](#)
 - [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 120](#)
 - [Reverting the Junos OS Software Image Back to the Previous Version on page 123](#)

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`

- See Also**
- [Example: Creating a Snapshot and Using It to Boot an SRX Series Device on page 220](#)
 - [Example: Installing Junos OS Upgrade Packages on SRX Series Devices on page 120](#)
 - [Reverting the Junos OS Software Image Back to the Previous Version on page 123](#)

CHAPTER 9

Zero Touch Provisioning

- [Understanding Zero Touch Provisioning on page 257](#)
- [Configuring Zero Touch Provisioning on page 261](#)
- [Configuring Zero Touch Provisioning on SRX Series Devices on page 266](#)
- [Monitoring Zero Touch Provisioning on page 274](#)

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.



NOTE: The `request system zeroize` command is not supported on PTX1000 routers. You must issue the `request vmhost zeroize` command (instead of `request system zeroize`) for factory default configuration on PTX1000 routers.

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.



NOTE: In Junos OS Release 18.1R1, if you are upgrading the software, you must perform a full software upgrade. A full upgrade includes upgrading both the Junos OS software and the host software packages.

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 187](#).



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 downloads the software image and then checks the software version. If the software version is same, the software is not upgraded.



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 01 (the name of the configuration file or the script file on the FTP, TFTP, or HTTP server) is configured, the switch or router downloads the file from the FTP, TFTP or HTTP server. If the file is a Junos OS configuration file, it is loaded with an override option in the system after the software image is installed. If the file is a script, the interpreter path mentioned in the file is used to execute the script after the software image is installed. You can view the script output in `/var/log/op-script.log`.

Additionally, the ZTP process is logged in `/var/log/image_load_log`. View this file to know the status of the ZTP process, such as file copy, software image installation, and configuration and script execution.

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

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.

3. If DHCP option 43, suboption 02 (a symbolic link (for example, `symlink_vmhost` or `vmhost_symlink`) to the software image file on the FTP, HTTP, or TFTP server) is configured, the switch or router downloads the software image and then checks the software version. If the software version is same, the software is not upgraded.
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 router or switch downloads the software image and then checks the software version. If the software version is same, the software is not upgraded.



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 261](#)

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.



NOTE: If you are upgrading the software, you must perform a full software upgrade. A full upgrade includes upgrading both the Junos OS software and the host software packages.

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.



NOTE: The **request system zeroize** command is not supported on PTX1000 routers. You must issue the **request vmhost zeroize** command (instead of **request system zeroize**) for factory default configuration on PTX1000 routers.

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-name code 0 = 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.http-port code 5 = 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 software image file name 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. If you select FTP as the transfer mode, Junos OS uses the anonymous FTP login to download files from the FTP 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 257](#)
- [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 249](#)

Configuring Zero Touch Provisioning on SRX Series Devices

- [Understanding Zero Touch Provisioning on SRX Series Devices on page 267](#)
- [Configuring Zero-Touch Provisioning on an SRX Series Device on page 270](#)
- [Understanding Factory-Default Configuration on SRX Series Device for Zero Touch Provisioning on page 273](#)

Understanding Zero Touch Provisioning on SRX Series Devices

This topic includes following sections:

- [Understanding ZTP on SRX Series Devices on page 267](#)
- [Network Activator Overview on page 267](#)
- [Limitations on page 270](#)

Understanding ZTP on SRX Series Devices

Zero Touch Provisioning (ZTP) enables you to provision and configure devices automatically, minimizing most of the manual intervention required for adding devices to a network. ZTP is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices.

ZTP on SRX Series devices is responsible for the initial bootup and configuration of the device when the device is powered on. This functionality includes:

- Providing the bare-minimum bootstrapping of the device. The SRX Series device is shipped with a factory-default configuration. The factory-default configuration includes the URL of the redirect server, that is used to connect to the central server by using a secure encrypted connection.
- Automatically connecting to the server over the Internet, and downloading the configuration and Junos OS image as specified by the customer or user from the server when the SRX Series device boots up with the factory-default configuration. The new image is installed first and then the initial configuration is applied and committed on the SRX Series device.

ZTP offers the following advantages:

- Simplified and faster deployment
- Increased configuration accuracy
- Support for scaling of network without additional resources

The ZTP process uses Network Activator to initially provision SRX Series devices.

Network Activator Overview

Network Service Activator enables fast device discovery and provisioning for automated configuration to eliminate complex device setup.

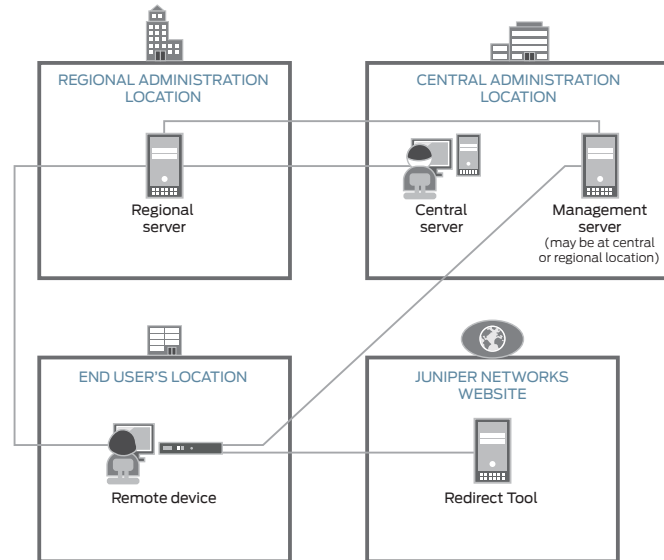
Network Activator initially provisions SRX Series devices (henceforth referred to as *remote devices* in this documentation), which reside at end users' sites. The remote devices download a boot image and initial configuration files from servers hosting Network Activator, using a process that provides full authorization and authentication for all interactions. When initial provisioning is complete, the remote device communicates with a management server, which then starts to manage and monitor the remote device.

Network Activator uses a distributed architecture to support remote devices. Network Activator is installed on one central administration server (central server) and multiple

regional administration servers (regional servers). A device communicates directly with its assigned regional server. The distributed architecture optimizes the efficiency of the initial provisioning process, contributing to high performance and scaling of the network.

Figure 13 on page 268 illustrates the distributed architecture and the components involved in the initial provisioning process.

Figure 13: Components Involved in Initial Provisioning of Remote Device

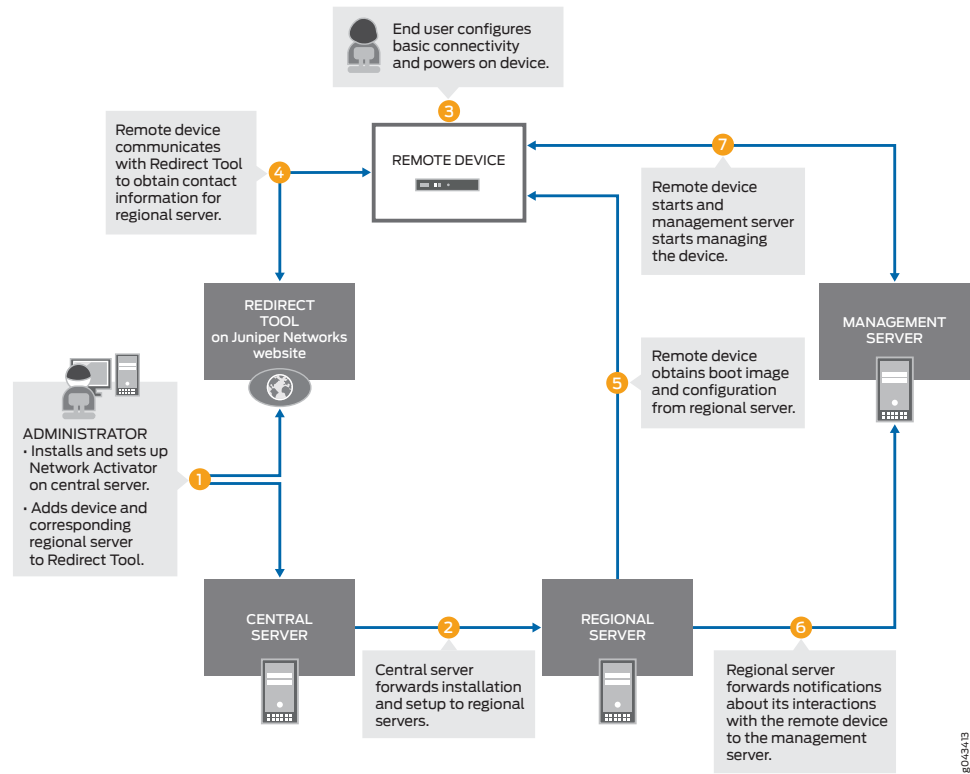


The roles of the components in the initial provisioning process are as follows:

- The remote device sends requests for initial provisioning. The remote device resides at the end user's location.
- The Redirect Tool provides authentication and authorization for remote devices to access their assigned regional servers through use of ITU-T X.509 private key infrastructure (PKI) digital certificates. Redirect service is hosted on Amazon Web Services (AWS), operated and maintained by Juniper Networks.
- The central server hosts Network Activator and communicates with the regional activator servers. Administrators at a service provider or central enterprise location interact with this server to install and set up Network Activator. The central server is located at a central geographic location for the service provider.
- The regional server also hosts Network Activator. This server stores information about its assigned remote devices and communicates directly with those devices. This server typically resides at a regional administrative location the provider designates for the end user.

Figure 14 on page 269 illustrates the initial provisioning workflow.

Figure 14: Workflow for Initial Provisioning



In detail, the provisioning workflow proceeds as follows:

1. The administrator at the service provider:
 - Installs and sets up Network Activator on the central server.
 - Adds remote devices and regional servers in the Redirect Tool.
2. The central server forwards the installation to the regional servers.
3. The end user powers on the remote device, connects it to a computer, and enters the authentication code in the webpage to send a request for initial provisioning.
4. The device transmits its X.509 certificate and fully qualified domain name (FQDN) as a provisioning request to the Redirect Tool.
5. The Redirect Tool searches its data store for the regional server that the administrator specified for this device, and confirms that the device's request corresponds to the X.509 certificate specified for the server.
6. The Redirect Tool sends contact information for the regional server to the device.

7. The device sends a request to the regional server for the URL of the boot image and the location of the initial configuration.
8. The regional server sends the information to the device.
9. The device obtains the boot image and configuration from the regional server.
10. The device uses the boot image and configuration to start and become operational.

For more details on Network Activator, see [Network Activator User Guide](#).

Limitations

- There are no restrictions on the number of attempts for entering the correct activation code.
- If the remote device is not able to reach the server (because the configured address in the factory-default configuration is not correct or the server is down, and so on), the remote device attempts to connect to an alternative server (if configured in the factory-default configuration). If there is only one server configured, then you can reattempt to connect. In such scenarios, we recommend that you configure the device manually through the console.
- Captive portal redirection, required for automatically redirecting users to the authentication webpage for entering the activation code, is not supported. You must manually navigate to the activation page after connecting to the device.

Configuring Zero-Touch Provisioning on an SRX Series Device

This section provides step-by-step instructions on how to use ZTP on an SRX Series device for initial provisioning of the device.

Before you begin:

- Unpack the device, install it, complete the necessary cabling, connect a laptop or any other terminal device, and power on the device. See the *Hardware Installation Guide* for your device more information.
- For SRX300, SRX320, SRX340, SRX345, and SRX550M devices, connect the management device and access the J-Web interface.

For more information, see Quick Start guides of respective devices at [SRX300](#), [SRX320](#), [SRX340](#), [SRX345](#), and [SRX550M](#).

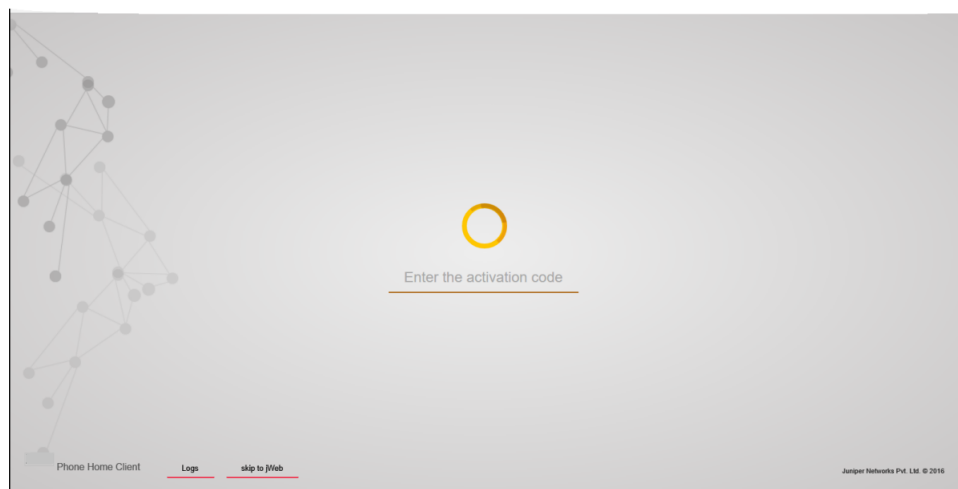
You are provided with an option to use ZTP; you can use this option or skip it and continue with J-Web wizards.

- For SRX1500 devices, before you can use J-Web to configure your device, you must access the CLI to configure the root authentication and the management interface. For more information, see [How to Set Up Your SRX1500 Services Gateway](#).

To provision an SRX Series device by using ZTP:

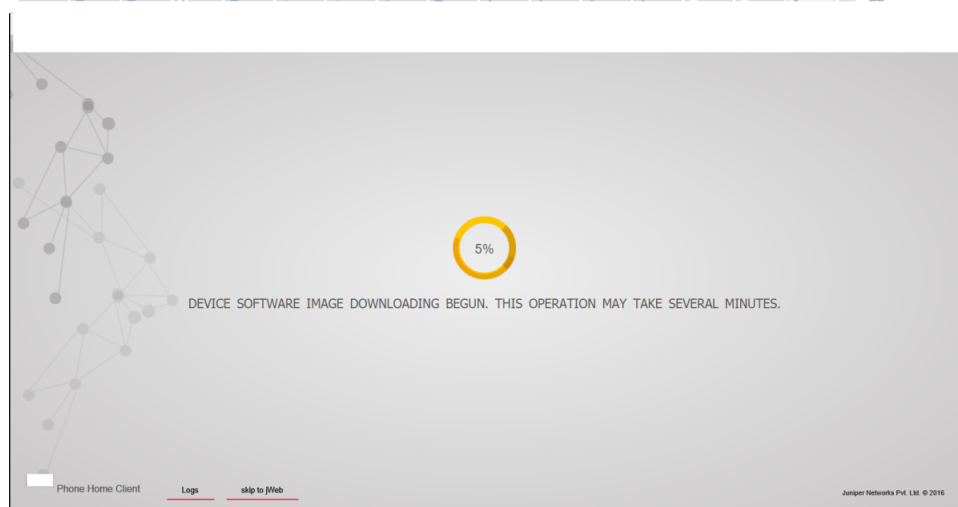
1. Connect a management device (PC or laptop) to any front panel Ethernet port (WAN port) of the SRX Series device.
2. Launch a Web browser from the management device and enter the authentication code in the webpage as shown in [Figure 15 on page 271](#).

Figure 15: Entering Activation Code for ZTP



After the device is successfully authenticated, it starts downloading the software image and initial configuration from the server as shown in [Figure 16 on page 271](#).

Figure 16: Initiating ZTP Process (Software Image Downloading)

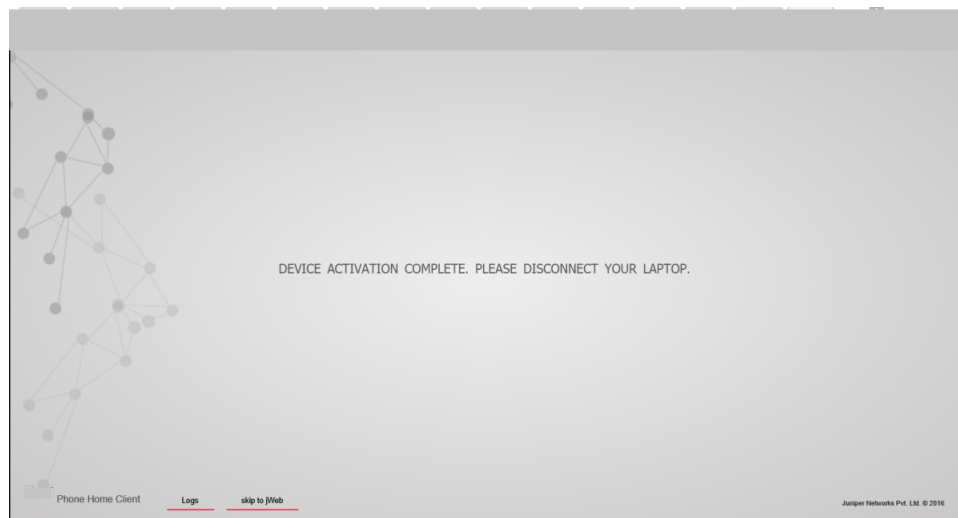


At this step:

- The activation code is sent to the server, and if the authentication is successful, the server pushes the initial configuration to the device. If the authentication is unsuccessful, you are asked to provide the correct code.
- The server can optionally push a new software image on the SRX Series device. In that case, the new image is installed first and then the initial configuration is applied and committed on the device.

The new image is installed and then the initial configuration is applied and committed on the device. When the process is complete, a confirmation message is displayed, as shown in [Figure 17 on page 272](#).

Figure 17: Completing ZTP Process



3. Click **Logs** to display details of the bootstrapping process.

After successfully installing the new software image and configuration on the system, the client sends the **bootstrap-complete** notification to the server that provided the image and the configuration. After the notification is sent, the configuration that includes the names of servers is deleted from the system. When you use ZTP the next time, you must explicitly configure the URL of the redirect server.



NOTE: In case of failure at any stage, the procedure is started all over again.

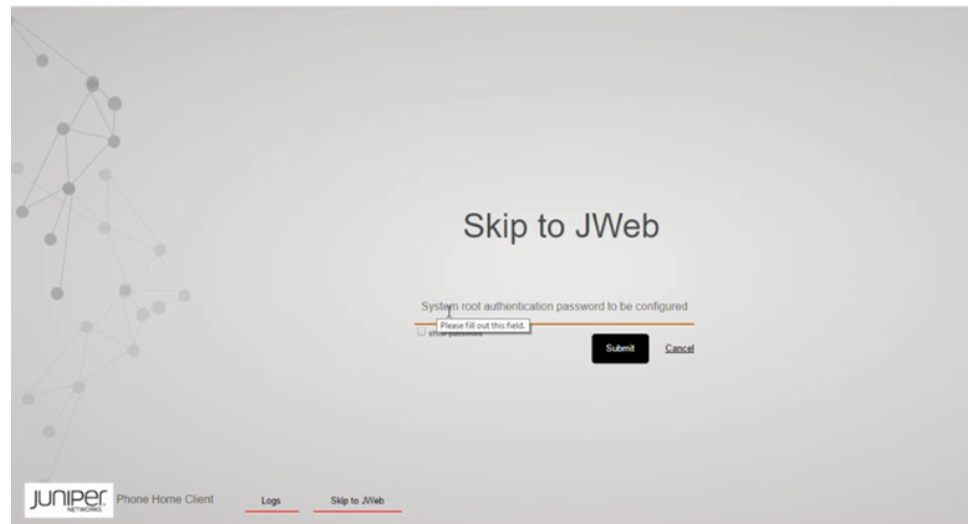


NOTE: The ZTP process either upgrades or downgrades the Junos OS version. During a downgrade on an SRX Series device, if you downgrade to a software version earlier than Junos OS Release 15.1X49-D100, in which ZTP is not supported, the autoinstallation phase of the ZTP process does not happen.

For SRX300, SRX320, SRX340, SRX345, and SRX550M devices, ZTP is the default method for provisioning the devices. However, if you want to use J-Web-based provisioning (J-Web setup wizards supported for the SRX300 line of devices and SRX550M devices), then instead of ZTP, you can use the option provided in the client portal to skip to J-Web setup wizards for performing the initial software configuration of your device.

If you select the **Skip to JWeb** option, you must configure the system root authentication password as shown in [Figure 18 on page 273](#).

Figure 18: Configuring System Root-Authentication Password



NOTE: For SRX1500 devices, the **Skip to JWeb** option is not supported. To access J-Web, the ZTP client configuration must be deleted during the initial setup of SRX1500 through CLI.

Understanding Factory-Default Configuration on SRX Series Device for Zero Touch Provisioning

Your services gateway is shipped with a factory-default configuration. Following is a sample of the default configuration that includes configuration for ZTP:

```
system {
  phone-home {
    rfc-compliant;
    server https://redirect.juniper.net;
  }
}
```

Note that, in this configuration:

- **server** indicates the name or IP address of the server. The factory-default configuration on an SRX Series device might include IP addresses of more than one servers.
- **rfc-compliant** indicates that after an upgrade, the server enforces certain behaviors that are compliant with RFC standards.



NOTE: By default, the system autoinstallation configuration is part of the factory-default configuration of the device. So, the administrator must ensure that the configuration file sent from the regional server to the remote device (SRX series device) must include the delete system autoinstallation option in the factory-default configuration.

Related Documentation

- [Juniper Networks Network Activator](#)

Monitoring Zero Touch Provisioning

Starting in Junos OS Release 12.2, you can use the console and operational commands to monitor Zero Touch Provisioning.

1. [Using the Console to Monitor Zero Touch Provisioning on page 274](#)
2. [Using System Log Alerts to Monitor Zero Touch Provisioning on page 275](#)
3. [Using Error Messages to Monitor Zero Touch Provisioning on page 275](#)
4. [Using System Log Files to Monitor Zero Touch Provisioning on page 275](#)
5. [Using the show dhcp client binding Command on page 276](#)
6. [Using the show dhcp client statistics Command on page 276](#)

Using the Console to Monitor Zero Touch Provisioning

The following Zero Touch Provisioning (ZTP) activities are displayed on the console during the ZTP process:

- Starting and ending times of ZTP process.
- Lists of bound and unbound DHCP client interfaces.
- DHCP options that DHCP servers send to DHCP clients.
- Logs indicating which interfaces are used for ZTP.
- ZTP parameters that DHCP clients obtain from DHCP servers.
- File names of configuration and image files, names of file servers, protocols used to fetch files, and times when DHCP servers fetch configuration and image files.
- Failure states caused by files not being on servers, or unreachable servers, and time outs.
- Number of attempts made, and number of attempts remaining, for retry in current ZTP cycle.
- Completion of file transfers.
- Installation, reboot, and state of ZTP process.
- Internal state errors and termination of ZTP process.
- Logs for when default routes were added or deleted.

Using System Log Alerts to Monitor Zero Touch Provisioning

Purpose In this example, the system log alert alerts you that the auto-image upgrade will start.

Action Use the following system log alert to monitor the auto-image upgrade process.

```
"ALERT:Auto-image upgrade will start. This can terminate config CLI session(s).  
Modified configuration will be lost. To stop Auto-image, in CLI do the  
following: 'edit; delete chassis auto-image-upgrade; commit'."
```

```
"Checking whether image upgrade is already invoked"
```

Meaning This system log alert indicates that the auto-image upgrade will start, and provides information on how to stop the auto-image upgrade process.

Using Error Messages to Monitor Zero Touch Provisioning

Purpose Error messages provide information on which DHCP options are not configured.

Action Use the information in the following error message to find out which DHCP options are not configured.

```
"DHCP Log Server Option"  
"DHCP Host Name Option"  
"DHCP NTP Server Option"
```

Meaning The error message indicates that the DHCP log server, hostname, and NTP server options are not configured.

Using System Log Files to Monitor Zero Touch Provisioning

Purpose System log files provide information on the state of the auto-upgrade process, lists of bound and unbound DHCP client interfaces, IP addresses of file servers, names and locations of image and configuration files, and successful and failed attempts at fetching configuration and image files.

Action Use the information in the following system log files to monitor the auto-upgrade process.

```
Auto Image Upgrade: Start fetching config-file file from server 10.1.1.1 through  
irb using ftp
```

```
Auto Image Upgrade: Tried [2] attempts to fetch config-file file from server  
10.1.1.1 through irb. Summary: "Retrieving /config-file  
:: Failed to open file.". To retry [4] times.
```

```
Auto Image Upgrade: Tried [4] attempts to fetch config-file file from server
10.1.1.1 through irb. Summary: "Retrieving /config-fileconfig-file
:: Failed to open file.". To retry [2] times.
```

```
Auto Image Upgrade: Tried [6] attempts to fetch config-file file from server
10.1.1.1 through irb. Summary: "Retrieving /config-file
:: Failed to open file.". To retry [0] times.
```

```
Auto Image Upgrade: All [6] attempts to fetch config-file file from server 10.1.1.1
through irb FAILED. Start retry again in few minutes.
```

Meaning These system log files indicate that there were six failed attempts to fetch the configuration file from the file server, the IP address of the file server, the DHCP client interface name, and the number of times the retry process occurred.

Using the show dhcp client binding Command

Purpose Issue the **show dhcp client binding** command to display DHCP client binding information

Action Issue the **show dhcp client binding** command to display the IP address of the DHCP client, the hardware address of the DHCP client, number of seconds in which the DHCP client's IP address lease expires, state of the DHCP client IP address in the binding table, and the name of the interface that has active client bindings.

show dhcp client binding

```
user@switch# show dhcp client binding
IP address      Hardware address Expires State      Interface
10.0.0.0        00:22:83:2a:db:dc 0       SELECTING irb.0
10.6.6.13       00:22:83:2a:db:dd 49201   BOUND      vme.0
10.0.0.0        00:22:83:2a:db:df 0       SELECTING xe-0/0/0.0
10.0.0.0        00:22:83:2a:db:e0 0       SELECTING xe-0/0/1.0
```

Meaning The output of this command shows that there is one client interface that is bound, and that there are three interfaces that are receiving DHCP offers from the DHCP server.

Using the show dhcp client statistics Command

Purpose Issue the **show dhcp client statistics** command to display DHCP client statistics.

Action Issue the **show dhcp client statistics** command to display DHCP client statistics, such as the number of packets dropped, and the number DHCP and BOOTP messages sent and received.

show dhcp client statistics

```
user@switch# show dhcp client statistics
Packets dropped:
Total              14
```

```

Send error                14
Messages received:
  BOOTREPLY               5
  DHCPOFFER               1
  DHCPACK                 4
  DHCPNAK                 0
  DHCPFORCERENEW         0
Messages sent:
  BOOTREQUEST             6751
  DHCPDECLINE             0
  DHCPDISCOVER            6747
  DHCPREQUEST             4
  DHCPINFORM              0
  DHCPRELEASE             0
  DHCPRENEW               0
  DHCPREBIND              0

```

Meaning The output of this command displays how many packets were dropped with errors, the number of BOOTREPLY and DHCPOFFER messages that were received, and the number of BOOTREQUEST and DHCPREQUEST messages that were sent.

Release History Table

Release	Description
12.2	Starting in Junos OS Release 12.2, you can use the console and operational commands to monitor Zero Touch Provisioning.

- Related Documentation**
- *Understanding Zero Touch Provisioning*
 - *Configuring Zero Touch Provisioning*

CHAPTER 10

Automatic Installation of Configuration Files

- [Autoinstallation Overview on page 279](#)
- [Understanding Autoinstallation of Configuration Files on page 285](#)
- [Configuring Autoinstallation of Configuration Files \(CLI Procedure\) on page 287](#)
- [Verifying Autoinstallation Status on page 289](#)
- [Example: Configuring Autoinstallation on SRX Series Devices on page 290](#)
- [Autoinstalling a Configuration File from a Disk-on-Key USB Memory Stick onto an EX2200 or EX3300 Switch on page 293](#)
- [Configuring Autoinstallation on JNU Satellite Devices on page 295](#)
- [Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 298](#)
- [Autoinstallation of Satellite Devices in a Junos Node Unifier Group on page 300](#)
- [Verifying Autoinstallation on JNU Satellite Devices on page 301](#)

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 280](#)
- [Supported Autoinstallation Interfaces and Protocols on page 280](#)
- [Typical Autoinstallation Process on a New Device on page 281](#)

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 24 on page 280](#) lists the protocols that the device can use on these interfaces for IP address acquisition.

Table 24: 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)

Table 24: Interfaces and Protocols for IP Address Acquisition During Autoinstallation (continued)

Interface and Encapsulation Type	Protocol for Autoinstallation
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: Starting in Junos OS Release 17.4R1, on SRX300, SRX320, SRX340, SRX345, and SRX550M devices, telnet and xnm-clear-text are not part system services in factory-default configurations.



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 290](#)

Understanding Autoinstallation of Configuration Files

Autoinstallation is the automatic configuration of a device over the network from a preexisting configuration file that you create and store on a configuration server—typically a Trivial File Transfer Protocol (TFTP) server. You can use autoinstallation to configure new devices automatically and to deploy multiple devices from a central location in the network.

You enable autoinstallation so that the switches in your network implement autoinstallation when they are powered on. To configure autoinstallation, you specify a configuration server, an autoinstallation interface, and a protocol for IP address acquisition.



NOTE: The QFX5200 switches only work with HTTP for autoinstallation. TFTP and FTP protocols are not supported.

This topic describes:

- [Typical Uses for Autoinstallation on page 285](#)
- [Autoinstallation Configuration Files and IP Addresses on page 285](#)
- [Typical Autoinstallation Process on a New Switch on page 286](#)

Typical Uses for Autoinstallation

Typical uses for autoinstallation of the software include:

- To deploy and update multiple devices from a central location in the network.
- To update a device—Autoinstallation occurs when a device that has been manually configured for autoinstallation is powered on.

Autoinstallation Configuration Files and IP Addresses

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 switch.

You can set up the following configuration files for autoinstallation on the switch:

- **network.conf**—Default configuration file for autoinstallation, in which you specify IP addresses and associated hostnames for devices on the network.
- **switch.conf**—Default configuration file for autoinstallation with a minimum configuration sufficient for you to telnet to the device and configure it manually.
- **hostname.conf**—Host-specific configuration file for autoinstallation on a device that contains all the configuration information necessary for the switch. In the filename, **hostname** is replaced with the hostname assigned to the switch.

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 switch, through which the new switch can send TFTP, Boot Protocol (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 Switch

When the switch configured for autoinstallation is powered on, it performs the following autoinstallation tasks:

1. The switch sends out DHCP or BOOTP requests on each connected interface simultaneously to obtain an IP address.

If a DHCP server responds to these requests, it provides the switch with some or all of the following information:

- An IP address and subnet mask for the autoinstallation interface.
- The location of the (typically) TFTP server, Hypertext Transfer Protocol (HTTP) server, 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 the server's 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 switch.
2. After the switch acquires an IP address, the autoinstallation process on the switch attempts to download a configuration file in the following ways:
 - a. If the DHCP server specifies the host-specific configuration file **hostname.conf**, the switch uses that filename in the TFTP server request. The autoinstallation process on the new switch makes three unicast TFTP requests for **hostname.conf**. If these attempts fail, the switch broadcasts three requests to any available TFTP server for the file.
 - b. If the switch does not locate a **hostname.conf** file, the autoinstallation process sends three unicast TFTP requests for a **network.conf** file that contains the switch's hostname-to-IP-address mapping information. If these attempts fail, the switch broadcasts three requests to any available TFTP server for the file.
 - c. If the switch fails to find a **network.conf** file that contains a hostname entry for the switch, the autoinstallation process sends out a DNS request and attempts to resolve the switch's IP address to a hostname.

- d. If the switch determines its hostname, it sends a TFTP request for the **hostname.conf** file.
 - e. If the switch is unable to map its IP address to a hostname, it sends TFTP requests for the default configuration file **switch.conf**. The TFTP request procedure is the same as for the **network.conf** file.
3. After the switch locates a configuration file on a TFTP server, the autoinstallation process downloads the file, installs the file on the switch, and commits the configuration.

**Related
Documentation**

- [Configuring Autoinstallation of Configuration Files \(CLI Procedure\) on page 287](#)
- [Connecting and Configuring an EX Series Switch \(CLI Procedure\)](#)
- [Connecting and Configuring an EX Series Switch \(J-Web Procedure\)](#)
- [Configuration Files Terms](#)

Configuring Autoinstallation of Configuration Files (CLI Procedure)

Autoinstallation is the automatic configuration of a device over the network from a pre-existing configuration file that you create and store on a configuration server—typically a Trivial File Transfer Protocol (TFTP) server. You can use autoinstallation to automatically deploy multiple devices from a central location in the network.

To specify autoinstallation to run when you power on a switch already installed in your network, you can enable it by specifying one or more interfaces, protocols, and configuration servers to be used for autoinstallation.

Before you explicitly enable and configure autoinstallation on the switch, perform these tasks as needed for your network's configuration:

- Have a service available—typically Dynamic Host Configuration Protocol (DHCP)—to assign an IP address to the switch
- Configure a DHCP server on your network to meet your network requirements. You can configure a switch to operate as a DHCP server. For more information, see *Configuring a DHCP Server on Switches (CLI Procedure)*.
- Create one of the following configuration files, and store it on a TFTP server (or HTTP server or FTP server) in the network:
 - A host-specific file with the name **hostname.conf** for each switch undergoing autoinstallation. Replace **hostname** with the name of a switch. The **hostname.conf** file typically contains all the configuration information necessary for the switch with this hostname.
 - A default configuration file named **switch.conf** with the minimum configuration necessary to enable you to telnet into the new switch for further configuration.

- Physically attach the switch to the network using a Gigabit Ethernet port.
- 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 Domain Name System (DNS) server in the network.
- If the switch 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 device 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 services. Connect this interface to the switch.
- If you are using **hostname.conf** files for autoinstallation, you must also complete the following tasks:
 - Configure the DHCP server to provide a **hostname.conf** filename to each switch. Each switch uses its **hostname.conf** filename to request a configuration file from the TFTP server. Copy the necessary **hostname.conf** configuration files to the TFTP server.
 - Create a default configuration file named **network.conf**, and copy it to the TFTP server. This file contains IP-address-to-hostname mapping entries. If the DHCP server does not send a **hostname.conf** filename to a new switch, the switch uses **network.conf** to resolve its hostname based on its IP address.

Alternatively, you can add the IP-address-to-hostname mapping entry for the switch to a DNS database file.

The switch uses the hostname to request a **hostname.conf** file from the TFTP server.

To configure autoinstallation:

1. Specify the URL address of one or more servers from which to obtain configuration files.

```
[edit system]
user@switch# set autoinstallation configuration-servers tftp://tftpconfig.example.com
```



NOTE: You can also use an FTP address, for example, **ftp://user:password@sftpconfig.example.com**.

2. Configure one or more Ethernet interfaces to perform autoinstallation and one or two procurement protocols for each interface. The switch uses the protocols to send a request for an IP address for the interface:

```
[edit system]
user@switch# set autoinstallation interfaces ge-0/0/0 bootp
```


To verify autoinstallation:

1. From the CLI, enter the **show system autoinstallation status** command.

```
user@switch> 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: switch-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-0/0/1
  State: None
  Address acquisition:
    Protocol: DHCP Client
    Acquired address: None
    Protocol: RARP Client
    Acquired address: None
```

- Related Documentation**
- [Understanding Autoinstallation of Configuration Files on page 285](#)
 - [Understanding DHCP Services for Switches](#)

Verifying Autoinstallation Status

Purpose Display the status of the autoinstallation feature.

Action From the CLI, enter the **show system autoinstallation status** command.

Sample Output

```
user@switch> 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: switch-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-0/0/1
  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 switch when it is deployed on the network.

Related Documentation

- [Configuring Autoinstallation of Configuration Files \(CLI Procedure\) on page 287](#)

Example: Configuring Autoinstallation on SRX Series Devices

This example shows how to configure a device for autoinstallation.

- [Requirements on page 290](#)
- [Overview on page 291](#)
- [Configuration on page 291](#)
- [Verification on page 292](#)

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 50](#)):
 - 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.



NOTE: Starting with Junos OS Release 15.1X49, you need to additionally configure the family inet under the interface using the `set interfaces ge-0/0/X unit 0 family inet` command for the SRX Series device to send dhcp requests out.

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 293](#)

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 279](#)

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

If you have a new EX2200 or EX3300 switch, you can use a Disk-on-Key USB memory stick (“USB key”) to configure the switch, using either a text configuration file or an XML configuration file.

Before you begin this task, ensure you have the following items:

- A management device (PC or laptop).
- A Junos Space platform to generate a valid XML file (if you will be installing the XML configuration file).
- A Disk-on-Key device with one of the following 16-bit or 32-bit FAT file systems:
 - DOS 3.0+ 16-bit FAT (up to 32 MB)
 - DOS 3.31+ 16-bit FAT (more than 32 MB)
 - FAT32
 - FAT32, LBA-mapped
 - 16-bit FAT, LBA-mapped
- An EX2200 or EX3300 switch with the factory configuration. If other Junos OS configuration files exist on the switch, the switch cannot read the **juniper-config.txt** or **juniper-config.xml** file from the Disk-on-Key device.



NOTE: The USB-based autoinstallation process overrides the network-based autoinstallation process. If the switch detects a Disk-on-Key device containing a valid configuration file during autoinstallation, it configures the switch by using the configuration file on the Disk-on-Key device instead of fetching the configuration from the network.

If both **juniper-config.txt** and **juniper-config.xml** files are on the Disk-on-Key device, the switch uses the text (txt) file.

To configure the switch by using a Disk-on-Key device that contains the configuration file in *text format*:

1. Using a text editor on the PC or laptop, create the configuration file, named **juniper-config.txt**, as a sequence of configuration commands (**set** commands). To reuse the configuration from another switch, save the configuration in configuration mode as a sequence of configuration commands on the switch using the **show | display set | save filename** command and then copying the file to the PC or switch as **juniper-config.txt**.



NOTE: Ensure that the first line in the **juniper-config.txt** is **[edit]** and that the last line in the file is **commit and-quit**.

2. Copy the **juniper-config.txt** file to the Disk-on-Key device.
3. Plug the Disk-on-Key device into the USB port on the switch.
4. Power on the switch.
5. Observe the LEDs on the Disk-on-Key device, and wait as the switch starts and then accesses the Disk-on-Key device.

The switch reads the **juniper-config.txt** file from the Disk-on-Key device and commits the configuration.



NOTE: Before you remove the Disk-on-Key device from the switch, ensure that the configuration has been applied to the switch. You can issue the **show configuration operational mode** command on the switch to see the configuration.

Then remove the Disk-on-Key device from the switch.

The configuration of the switch is complete.

To configure the switch by using a Disk-on-Key device that contains the configuration file in *XML format*:

1. Power on the switch.
2. Configure the switch to use autoinstallation:
 - a. Load the factory default configuration:

```
[edit]
user@switch# load factory-default
```
 - b. Set the switch for autoinstallation:

```
[edit]
user@switch# set system autoinstallation delete-upon-commit
```

- c. Set the root authentication password:

```
[edit]
user@switch# set system root-authentication plain-text-password
```

- d. Commit the changes:

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

3. Power off the switch.
4. Using the Junos Space platform, create a valid configuration file in XML format, and name it **juniper-config.xml**.
5. Copy the **juniper-config.xml** file to the Disk-on-Key device.
6. Plug the Disk-on-Key device into the USB port on the switch.
7. Power on the switch.
8. Observe the LEDs on the Disk-on-Key device, and wait as the switch starts and then accesses the Disk-on-Key device.

The switch reads the **juniper-config.xml** file from the Disk-on-Key device and commits the configuration.



NOTE: Before you remove the Disk-on-Key device from the switch, ensure that the configuration has been applied to the switch. You can issue the **show configuration operational mode** command on the switch to see the configuration.

Then remove the Disk-on-Key device from the switch.

The configuration of the switch is complete.

- Related Documentation**
- [show system autoinstallation status on page 532](#)
 - [Understanding Software Installation on EX Series Switches on page 59](#)

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 300](#)
- [Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 298](#)
- [Verifying Autoinstallation on JNU Satellite Devices on page 301](#)
- [autoinstallation on page 396](#)
- [delete-after-commit \(JNU Satellites\) on page 400](#)
- [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 298](#)
- [Typical Autoinstallation Process on a New Router on page 298](#)

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 300](#)
 - [Configuring Autoinstallation on JNU Satellite Devices on page 295](#)
 - [Verifying Autoinstallation on JNU Satellite Devices on page 301](#)
 - [autoinstallation on page 396](#)
 - [delete-after-commit \(JNU Satellites\) on page 400](#)
 - [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 298](#)
 - [Configuring Autoinstallation on JNU Satellite Devices on page 295](#)
 - [Verifying Autoinstallation on JNU Satellite Devices on page 301](#)

- [autoinstallation on page 396](#)
- [delete-after-commit \(JNU Satellites\) on page 400](#)
- [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 300](#)
- [Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 298](#)
- [Configuring Autoinstallation on JNU Satellite Devices on page 295](#)
- [autoinstallation on page 396](#)
- [delete-after-commit \(JNU Satellites\) on page 400](#)
- [*configuration-servers*](#)
- [show system autoinstallation status on page 532](#)

CHAPTER 11

Installation, Upgrade, and Recovery of VM Host Support on Devices with Routing Engines

- [Routing Engines with VM Host Support on page 303](#)
- [Architecture of Routing Engines with VM Host Support on page 304](#)
- [Salient Features of the Routing Engines with VM Host Support on page 305](#)
- [Routers with VM Host Support-Boot Process on page 310](#)
- [VM Host Installation on page 312](#)
- [Copying VM Host Installation Package to the PXE Boot Server on page 316](#)
- [Creating an Emergency Boot Device for Routing Engines with VM Host Support on page 318](#)
- [Upgrading the SSD Firmware on Routing Engines with VM Host Support on page 320](#)
- [Disabling Autorecovery on Routing Engines with VM Host Support on page 322](#)
- [VM Host Operations and Management on page 322](#)

Routing Engines 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 25 on page 304 lists the hardware specifications of the Routing Engines.

Table 25: 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 Routing Engines with VM Host Support

Figure 19 on page 305 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.

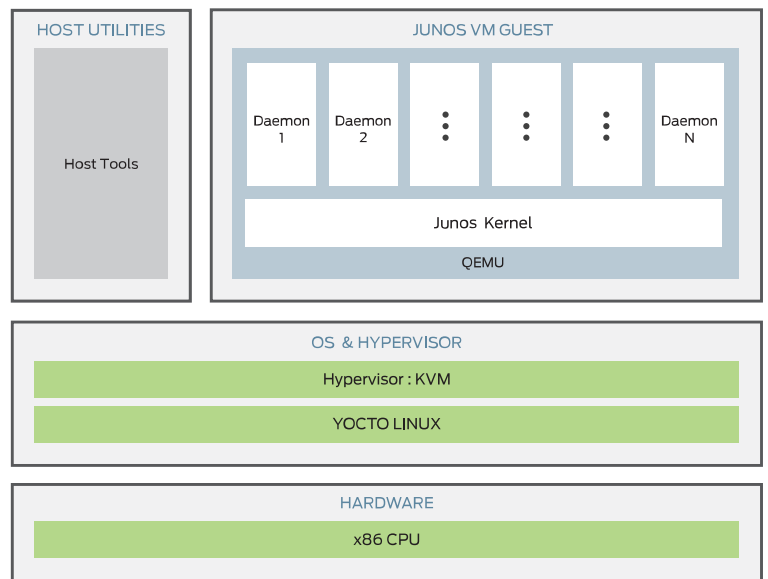
The server at the hardware layer contains the physical network interface cards (NICs), CPUs, memory, and Ethernet management port. The NICs support hardware virtualization based on single root I/O virtualization (SR-IOV). With SR-IOV, the physical NICs (known as a physical functions) are managed by the 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 19: Architecture of RE-MX-X6, RE-MX-X8 and RE-PTX-X8 Routing Engines



Related Documentation • [Routing Engines with VM Host Support on page 303](#)

Salient Features of the Routing Engines with VM Host Support

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

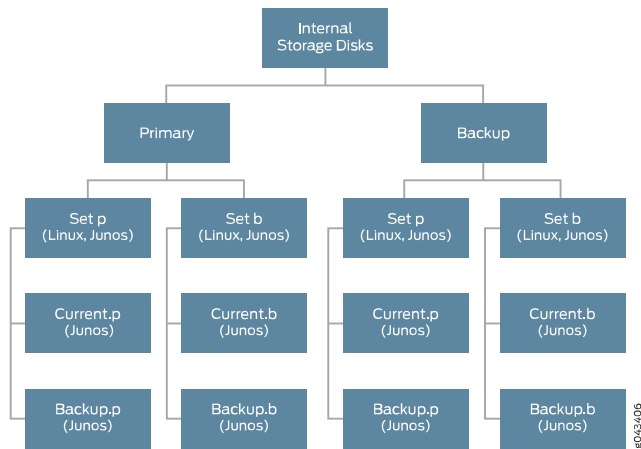
An Internal solid-state drive (SSD) is used as boot media for operating the Routing Engine. Additional options such as USB storage and network boot are available for installation and recovery purposes. A set of two 50-GB SSDs is available for normal functioning of the Routing Engine. The Routing Engine 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 20 on page 307](#) shows the partitioning of SSDs.

Figure 20: 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.

Starting in Junos OS Release 18.1R1, the Routing Engines on the MX240, MX480, MX960, MX2010, MX2020, and PTX5000 support Secure Boot.

The Routing Engines with Secure Boot support have both RAM and SSD upgraded to 128GB and 2x200GB respectively. The increased SSD size facilitates increased storage of core and log files.

The following table provides information on the SSD size for different Routing Engines:

Routers	Routing Engine model number	SSD size
MX240, MX480, and MX960	RE-S-2200X6-64G-S	2x50GB
	RE-S-X6-64G-LT	2x50GB
	RE-S-X6-128G-S	2x200GB
PTX5000	RE-P-2200-64G-S	2x50GB
	RE-PTX-X8-128G-S	2x200GB
MX2010 and MX2020	RE-MX2K-X8-64G	2x100GB
	RE-MX2K-X8-64G-LT	2x100GB
	RE-MX2K-X8-128G-S	2x200GB

You can use the **show vmhost hardware** command to display the increased RAM size, SSD size, and other hardware information.

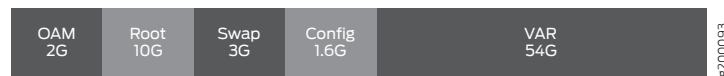
The following illustrations explain the partition of the host to facilitate the increased storage of core files and log files. [Figure 21 on page 308](#) illustrates the partition of the host on MX240, MX480, MX960 and PTX5000 routers with the 200-GB SSDs. A virtual disk of size 56-GB will be allocated from VM partition to the guest as var-config.disk. The current size of this disk is 15-GB.

Figure 21: Host partition table for Routing Engines with 200-GB SSDs



[Figure 22 on page 308](#) illustrates the storage allocation of the guest VM.

Figure 22: Partitioning of the guest VM



NOTE: For Routing Engines with 50GB SSD, the host partition remains as-is.

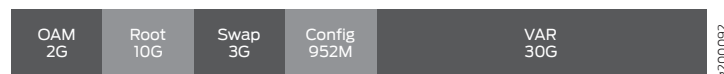
[Figure 23 on page 308](#) and [Figure 24 on page 308](#) illustrate the host partition table and the storage allocation of the guest VM for the MX2010 and MX2020 routers respectively.

Figure 23: Host partition table for Routing Engines on MX2010 and MX2020 routers with 100GB SSD



A virtual disk of size 32-GB is allocated from VM partition to the guest Junos OS as var-config.disk.

Figure 24: Guest VM partition on MX2010 and MX2020 Routers



A reformatting of the SSD is required to implement the enhancement of the /var size. The upgrade can be implemented by any of the following methods:

- Installation from SSD Disk2-Boot the host OS from the backup disk (SSD Disk2) and install the junos-vmhost-install-x.tgz image.
- Installation from USB

Related Documentation

- [request vmhost snapshot on page 590](#)
- [request vmhost reboot on page 588](#)

- [request vmhost power-off on page 585](#)
- [Upgrading the SSD Firmware on Routing Engines with VM Host Support on page 320](#)

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 Routing Engines with VM Host Support on page 322](#)

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 590](#)
- [request vmhost reboot on page 588](#)
- [request vmhost power-off on page 585](#)
- [Disabling Autorecovery on Routing Engines with VM Host Support on page 322](#)

Routers with VM Host Support-Boot Process

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 Routing Engines with VM Host Support on page 318](#)

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*.

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 411](#)
 - [request vmhost reboot on page 588](#)
 - [request vmhost power-off on page 585](#)
 - [Creating an Emergency Boot Device for Routing Engines with VM Host Support on page 318](#)

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 316](#)

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 Routing Engines with VM Host Support” on page 318](#).

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.



NOTE: The RE-S-X6-64G-LT and RE-MX2K-X8-64G-LT Routing Engines are restricted to boot only the Junos OS with upgraded FreeBSD Limited image. They fail to boot if you try to install or upgrade the router with an image other than the Limited image, which begins with the `junos-vmhost-install` prefix.

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.

The upgrade applies to one partition of the SSD (primary). In case of a boot failure on the primary SSD, the router can boot by using the snapshot available on the alternate SSD.

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 593](#)
- [request vmhost software rollback on page 599](#)
- [request vmhost snapshot on page 590](#)
- [show vmhost snapshot on page 625](#)

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/vecdd-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 312](#)
- [Creating an Emergency Boot Device for Routing Engines with VM Host Support on page 318](#)

Creating an Emergency Boot Device for Routing Engines with VM Host Support

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 210.
- **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

- [Routers with VM Host Support-Boot Process on page 310](#)

Upgrading the SSD Firmware on Routing Engines with VM Host Support

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 Routing Engines with VM Host Support

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 411](#)

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**

- [Routing Engines with VM Host Support on page 303](#)

CHAPTER 12

Installing and Managing Software Licenses

- [Junos OS Feature Licenses on page 325](#)
- [License Key Components for the EX Series Switch on page 326](#)
- [Understanding Software Licenses for EX Series Switches on page 327](#)
- [License Enforcement on page 336](#)
- [Software Feature Licenses on page 337](#)
- [Junos OS Feature License Keys on page 354](#)
- [Managing Licenses for the EX Series Switch \(CLI Procedure\) on page 357](#)
- [Monitoring Licenses for the EX Series Switch on page 358](#)
- [Generating License Keys on page 360](#)
- [Adding New Licenses \(CLI Procedure\) on page 362](#)
- [Deleting License Keys \(CLI\) on page 367](#)
- [Saving License Keys \(CLI\) on page 369](#)
- [Verifying Junos OS License Installation \(CLI\) on page 369](#)
- [License Modes for Enhanced MPCs Overview on page 372](#)
- [Configuring the License Mode for Specific Enhanced MPCs on MX Series Routers on page 373](#)
- [Example: Configuring the License Mode for MPC5E on page 374](#)

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**
- [Verifying Junos OS License Installation \(CLI\) on page 369](#)
 - [show system license on page 547](#)

License Key Components for the EX Series Switch

When you purchase a license for a Junos OS feature that requires a separate license, you receive a license key.

A license key consists of two parts:

- License ID—Alphanumeric string that uniquely identifies the license key. When a license is generated, it is given a license ID.
- License data—Block of binary data that defines and stores all license key objects.

For example, in the following typical license key, the string **Junos204558** is the license ID, and the trailing block of data is the license data:

```
Junos204558 aeaqea qmijhd amrqha ztfmbu gqzama uqceds
ra32zr lsevik ftvjed o4jy5u fynzzj mgviyl
kgioyf ardb5g sj7wnt rsfked wbjf5a sg
```

The license data defines the device ID for which the license is valid and the version of the license.

- Related Documentation**
- [Managing Licenses for the EX Series Switch \(CLI Procedure\) on page 357](#)
 - [Managing Licenses for the EX Series Switch \(J-Web Procedure\)](#)
 - [Understanding Software Licenses for EX Series Switches on page 327](#)

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 327](#)
- [Features Requiring a License on EX2200 Switches on page 328](#)
- [Features Requiring a License on EX2300 Switches on page 329](#)
- [Features Requiring a License on EX3300 Switches on page 329](#)
- [Features Requiring a License on EX3400 Switches on page 331](#)
- [Features Requiring a License on EX4300 Switches on page 332](#)
- [Features Requiring a License on EX4600 Switches on page 333](#)
- [Features Requiring a License on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches on page 334](#)
- [License Warning Messages on page 335](#)

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 (<https://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 26 on page 328 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 26: 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, VRRPv3
- 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 27 on page 329 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 27: 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
EX2300-48T EX2300-48P	EX-48-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, VRRPv3, 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 28 on page 330](#) 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 28: 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 29 on page 331](#) 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 29: 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, VRRPv3, 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 30 on page 331](#) 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 30: 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)
- RIPng (RIPng is for RIP IPv6)
- Unicast reverse-path forwarding (RPF)
- Virtual Router
- Virtual Router Redundancy Protocol (VRRP)

[Table 31 on page 332](#) 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 31: 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 32 on page 333 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 32: Junos OS AFL Part Number on EX4300 Switches

Switch Model	AFL Part Number
EX4300-24T EX4300-24P	EX4300-24-AFL
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 (<https://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 33 on page 333 lists the AFLs that you can purchase for EX4600 switch models.

Table 33: 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 (<https://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 34 on page 334 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 34: Junos OS AFL Part Number on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches

Switch Model	AFL Part Number
EX3200-24P	EX-24-AFL
EX3200-24T	
EX4200-24F	
EX4200-24P	
EX4200-24PX	
EX4200-24T	

Table 34: Junos OS AFL Part Number on EX3200, EX4200, EX4500, EX4550, EX6200, EX8200, and EX9200 Switches (continued)

Switch Model	AFL Part Number
EX3200-48P EX3200-48T EX4200-48F EX4200-48P EX4200-48PX EX4200-48T	EX-48-AFL
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 (<https://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 \(CLI Procedure\) on page 357](#)
- [Monitoring Licenses for the EX Series Switch on page 358](#)
- [License Key Components for the EX Series Switch on page 326](#)

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

- [Adding New Licenses \(CLI Procedure\) on page 362](#)
- [Deleting License Keys \(CLI\) on page 367](#)
- [Saving License Keys \(CLI\) on page 369](#)
- [Verifying Junos OS License Installation \(CLI\) on page 369](#)

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 <https://www.juniper.net/in/en/contact-us/>.

- [Software Features That Require Licenses on M Series, MX Series, and T Series Routers on page 338](#)
- [Software Features That Require Licenses on M Series Routers Only on page 341](#)
- [Software Features That Require Licenses on MX Series Routers Only on page 342](#)

- [Software Feature Licenses for SRX Series Devices on page 348](#)
- [Software Features That Require Licenses on EX Series Switches on page 348](#)
- [Software Features That Require Licenses on the QFX Series on page 349](#)
- [Disaggregated Software Features That Require Licenses on the QFX Series on page 352](#)

Software Features That Require Licenses on M Series, MX Series, and T Series Routers

Table 35 on page 338 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 <https://www.juniper.net/in/en/contact-us/>.

Table 35: 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

Table 35: 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 Passive Monitoring DFC Application, supporting 100Kpps throughput; Chassis based license for Multiservices PIC	M320, T640, T320, and T1600	S-DFC-100K
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

Table 35: Junos OS Feature License Model Number for M Series, MX Series, and T Series Routers (continued)

Licensed Software Feature	Supported Devices	Model Number
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
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 Pack 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

Table 35: 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 MX10 routers	MX10	mx10-to-mx40-upgrade
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 36 on page 341 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 <https://www.juniper.net/in/en/contact-us/>.

Table 36: 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

Table 36: Junos OS Feature License Model Number for M Series Routers (continued)

Licensed Software Feature	Supported Devices	Model Number
NAT, FW license on Integrated ASM and Integrated Multi Services Module: Single instance	M7i	S-NAT-FW-SINGLE-BB
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 37 on page 342 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 <https://www.juniper.net/in/en/contact-us/>.

Table 37: 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

Table 37: Junos OS Feature License Model Number for MX Series Routers (continued)

Licensed Software Feature	Supported Devices	Model Number
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
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 37: 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 37: 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 37: 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 37: 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 37: 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 <https://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 327.

For information about how to purchase a software license, contact your Juniper Networks sales representative at <https://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 38 on page 350](#) 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](#)” on page 352.

For information about how to purchase a software license, contact your Juniper Networks sales representative.

Table 38: 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
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
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)	QFX5110-48S switch	One per switch, two per Virtual Chassis, and two per Virtual Chassis Fabric	QFX5K-C1-PFL
	QFX5110-32Q switch		
	QFX5200-48Y		
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)			QFX5K-C1-AFL

Table 38: 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 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)	QFX5210-64C switch	One per switch, two per Virtual Chassis, and two per Virtual Chassis Fabric	QFX5K-C1-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)			QFX5K-C1-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 38: 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) feature license	Any member device in a Virtual Chassis Fabric (VCF)	Two per Virtual Chassis Fabric (VCF)	QFX-VCF-LIC

Disaggregated Software Features That Require Licenses on the QFX Series

- [Disaggregated Software Feature Licenses on QFX5200 Switches on page 353](#)

Disaggregated Software Feature Licenses on QFX5200 Switches



NOTE: For information on standard Junos OS feature licenses, see “[Software Features That Require Licenses on the QFX Series](#)” on page 349.

The Junos OS software is disaggregated from the hardware. With disaggregated Junos OS, you can purchase the following feature licenses, which are available on a perpetual basis:

- Junos Base Software (JBS) license:

Includes basic layer 2 switching, basic layer 3 routing, multicast, automation, programmability, Zero Touch Provisioning (ZTP) and basic monitoring.



NOTE: You must purchase the JBS license to use basic functions, but you do not need to install the license key in Junos OS Release 15.1X53-D30. JBS basic functions work with this release without installing the license key. However, you will need to install the license key in a future release of Junos OS to be determined, so make sure to retain the authorization code you received from the License Management System to generate a license key for the JBS license.

- Junos Advanced Software (JAS) license:

Includes features supported in JBS license and Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (IS-IS), and Virtual Extensible Local Area Network (VXLAN). You need to install the license key to use these features.

- Junos Premium Software (JPS) license:

Includes features supported in JAS license and Multi-protocol Label Switching (MPLS) feature set. You need to install the license key to use these features.

For information about how to purchase a software feature license, contact your Juniper Networks sales representative.

Table 39: Disaggregated Junos OS Feature Licenses and Associated SKU's

Licensed Software Features	SKU's
Junos base software (JBS) license	QFX5000-35-JBS
Junos advanced software (JAS) license	QFX5000-35-JAS
Junos premium software (JPS) license	QFX5000-35-JPS

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 40 on page 356](#) 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 40: 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 • 1/MIC0	f1—MX5 to MX10 upgrade	Slot 1 and 2 • 1/MIC0 • 1/MIC1
MX10	40G	0x555	Slot 1 and 2 • 1/MIC0 • 1/MIC1	f2—MX10 to MX40 upgrade	Slot 2 and first 2 ports on Slot 0 • 1/MIC1 • First 2 ports on 0/MIC0
MX40	60G	0x554	Slot 1, Slot 2 and first 2 ports on Slot 0 • 1/MIC0 • 1/MIC1 • First 2 ports on 0/MIC0	f3—MX40 to MX80 upgrade	Slot 2 and all ports on Slot 0 • 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 41 on page 357](#).

Table 41: 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 41 on page 357](#) 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

- [License Enforcement on page 336](#)
- [Software Feature Licenses on page 337](#)
- [Verifying Junos OS License Installation \(CLI\) on page 369](#)
- [show system license on page 547](#)

Managing Licenses for the EX Series Switch (CLI Procedure)

To enable and use some Junos OS features on an EX Series switch, you must purchase, install, and manage separate software licenses. Each switch requires one license. For a Virtual Chassis deployment, two licenses are recommended for redundancy. After you have configured the features, you see a warning message if the switch does not have a license for the feature.

Before you begin managing licenses, be sure that you have:

- Obtained the needed licenses. For information about how to purchase software licenses, contact your Juniper Networks sales representative.
- Understand what makes up a license key. For more information, see [“License Key Components for the EX Series Switch” on page 326](#).

This topic includes the following tasks:

- [Adding New Licenses on page 358](#)
- [Deleting Licenses on page 358](#)
- [Saving License Keys on page 358](#)

Adding New Licenses

To add one or more new license keys on the switch, with the CLI:

1. Add the license key or keys:

- To add one or more license keys from a file or URL, specify the filename of the file or the URL where the key is located:

```
user@switch> request system license add filename | url
```

- To add a license key from the terminal:

```
user@switch> 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 the license entry mode.

Deleting Licenses

To delete one or more license keys from the switch with the CLI, specify the license ID:

```
user@switch> request system license delete license-id
```

You can delete only one license at a time.

Saving License Keys

To save the installed license keys to a file (which can be a URL) or to the terminal:

```
user@switch> request system license save filename | url
```

For example, the following command saves the installed license keys to a file named **license.conf**:

```
user@switch> request system license save ftp://user@switch/license.conf
```

Related Documentation

- [Managing Licenses for the EX Series Switch \(J-Web Procedure\)](#)
- [Monitoring Licenses for the EX Series Switch on page 358](#)
- [Understanding Software Licenses for EX Series Switches on page 327](#)

Monitoring Licenses for the EX Series Switch

To enable and use some Junos OS features on the EX Series switch, you must purchase, install, and manage the appropriate software licenses. Each switch requires one license. For a Virtual Chassis deployment, two licenses are recommended for redundancy.

To monitor your installed licenses, perform the following tasks:

- [Displaying Installed Licenses and License Usage Details on page 359](#)
- [Displaying Installed License Keys on page 360](#)

Displaying Installed Licenses and License Usage Details

Purpose Verify that the expected license is installed and active on the switch and fully covers the switch configuration.

Action From the CLI, enter the **show system license** command. (To display only the **License usage** list, enter the **show system license usage** command. To display only the **Licenses installed** output, enter **show system license installed**.)

```
user@switch> show system license
License usage:
```

Feature name	Licenses	Licenses	Licenses	Expiry
	used	installed	needed	
bgp	1	1	0	permanent
isis	0	1	0	permanent
ospf3	0	1	0	permanent
ripng	0	1	0	permanent
mpls	0	1	0	permanent

```
Licenses installed:
```

```
License identifier: JUNOS204558
```

```
License version: 2
```

```
Valid for device: BN0208380000
```

```
Features:
```

```
ex-series - Licensed routing protocols in ex-series
```

```
permanent
```

Meaning The output shows the license or licenses (for Virtual Chassis deployments) installed on the switch and license usage. Verify the following information:

- If a feature that requires a license is configured (used), a license is installed on the switch. The **Licenses needed** column must show that no licenses are required.

- The appropriate number of licenses is installed. Each switch requires one license. For a Virtual Chassis deployment, two licenses are recommended for redundancy.
- The expected license is installed.

Displaying Installed License Keys

Purpose Verify that the expected license keys are installed on the switch.

Action From the CLI, enter the **show system license keys** command.

```
user@switch> show system license keys
JUNOS204558 abcdef qhijkl mnopqr stuvwxyz yzabcd efghij
klmnop qrstuv wxyzab cdefgh ijklmn opqrst
uvwxyz 61abcd efgh21 31efgh yzabcd
```

Meaning The output shows the license key or keys (for Virtual Chassis deployments) installed on the switch. Verify that each expected license key is present.

- Related Documentation**
- [Managing Licenses for the EX Series Switch \(CLI Procedure\) on page 357](#)
 - [Managing Licenses for the EX Series Switch \(J-Web Procedure\)](#)
 - [Understanding Software Licenses for EX Series Switches on page 327](#)

Generating License Keys

When you purchase a Junos OS software feature license for a device, you receive an e-mail containing an authorization code for the feature license from Juniper Networks. You can use the authorization code to generate a unique license key (a combination of the authorization code and the device's serial number) for the device, and then add the license key on the device.

Before generating the license keys for a device:

- Purchase the required licenses for the device. See [“Software Features That Require Licenses on the QFX Series” on page 349](#) and [“Disaggregated Software Features That Require Licenses on the QFX Series” on page 352](#).
- Note down the authorization code in the e-mail you received from Juniper Networks when you purchased the license.
- Determine the serial number of the device. For instructions, see *Locating the Serial Number on a QFX3500 Device or Component*.

To generate the license keys for a device:



NOTE: This procedure shows you how to generate license keys on a QFX Series device, but you can follow the same procedure for any device.

1. In a browser, log in to the Juniper Networks License Management System at <https://www.juniper.net/lcrs/license.do>.

The Manage Product Licenses page appears.



NOTE: To access the licensing site, you must have a service contract with Juniper Networks 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>.

2. On the Generate Licenses tab, select **QFX Series Product** from the drop-down list, and click **Go**.

The Generate Licenses - QFX Series Product page appears.

3. Select the **QFX Series Product Device** option button, and click **Continue**.

The Generate Licenses - QFX Series Product Devices page appears.

4. In the **Device Serial Number** field, enter the serial number for the device.

5. In the **Authorization Code** field, enter the authorization code in the e-mail you received from Juniper Networks when you purchased the license.

6. (Optional) If you want to enter another authorization code for the same device, click **Enter More Authorization Codes** to display a new authorization code field. Enter the authorization code in this field.

7. Click **Confirm**.

The Confirm License Information page appears, displaying a summary of the information you submitted to the License Management System.

8. Review the information to ensure everything is correct and then click **Generate License**.

The Generate Licenses - QFX Series Product Devices page appears, displaying a summary of your license keys, including a link that displays the details of your new license keys.

9. Select the file format in which you want to obtain your new license keys.
10. Select the delivery method you want to use to obtain your new license keys.

To download the license keys:

- Select the **Download to this computer** option button, and click **OK**.

To e-mail the license keys:

- Select the **Send e-mail to e-mail ID** option button, and click **OK**.

Related Documentation

- [Software Features That Require Licenses on the QFX Series on page 349](#)
- [Adding New Licenses \(CLI Procedure\) on page 362](#)
- [Locating the Serial Number on a QFX3500 Device or Component](#)

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 362](#)
- [Installing a License Using an Operational Command on page 366](#)

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 363](#)
- [Installing Licenses Using a Configuration File on page 364](#)

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 366](#)
- [Adding a License to a Device with Dual Routing Engines on page 366](#)

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 369](#).

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 367](#)
- [Junos OS Feature Licenses on page 325](#)
- [Verifying Junos OS License Installation \(CLI\) on page 369](#)
- [request system license add on page 432](#)

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 367](#)
- [Using a Configuration Command to Delete Licenses on page 368](#)

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.

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

- [Adding New Licenses \(CLI Procedure\) on page 362](#)

Verifying Junos OS License Installation (CLI)

To verify Junos OS license management, perform the following tasks:

- [Displaying Installed Licenses on page 369](#)
- [Displaying License Usage on page 371](#)

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.

See Also • [Adding New Licenses \(CLI Procedure\) on page 362](#)

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-12tp	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.

See Also • [Adding New Licenses \(CLI Procedure\) on page 362](#)

License Modes for Enhanced MPCs Overview

Enhanced MPCs are available in three license variants. Before Junos OS Release 16.1, there were two variants: infrastructure routing (IR) and routing (R). Starting in Junos OS Release 16.1, there is also a base variant, making a total of three license variants. All variants support an identical feature set, but with a few scale differences. [Table 42 on page 372](#) describes the three license variants.

Table 42: License Variants for MPCs

License	How to Identify	Description
base	No special suffix in the license name.	<ul style="list-style-type: none"> All Layer 2, Layer 2.5, and Layer 3 features. Up to 32 Layer 3 routing instances of the virtual routing and forwarding (VRF) instance type. Up to 2 million routes in the forwarding information base (FIB), provided there is hardware support. (FIB is also known as forwarding table.) Up to 6 million routes in the routing information base (RIB), also known as routing table.
IR	-IR suffix in the license name.	<ul style="list-style-type: none"> All Layer 2, Layer 2.5, and Layer 3 features. Up to 32 Layer 3 routing instances of the virtual routing and forwarding (VRF) instance type.
R	-R suffix in the license name.	Full-scale Layer 2, Layer 2.5, and Layer 3 features. Scale is determined by the hardware capabilities.

Suppose you have purchased two MPC4Es: one with IR license and one with R license. After the MPCs are installed on a router, both MPCs appear identical. To distinguish between an MPC with an IR license and an MPC with an R license after the MPC is installed on the router, you must configure the license mode based on the license purchased. For instance, if you have purchased an MPC with the IR license, you must configure the license mode for that MPC as IR. The license mode settings are set specific to each MPC slot. If the MPC is installed in a different slot, or moved to another device, the license mode settings must be reconfigured on the new slot or device. Also, the license mode settings previously configured must be deleted.



NOTE: The license mode settings are used only to provide information. You cannot set or alter the license of the MPC by configuring the license mode.

To view the current license mode settings on an MPC, from the configuration mode, use the **show chassis fpc** command. To view the current license mode settings on an MPC, from the operational mode, use the **show chassis hardware extensive** command. To delete the existing license mode settings on an MPC, use the **delete chassis fpc** command.

Release History Table

Release	Description
16.1	Starting in Junos OS Release 16.1, there is also a base variant, making a total of three licence variants.

Related Documentation

- [Configuring the License Mode for Specific Enhanced MPCs on MX Series Routers on page 373](#)
- [Example: Configuring the License Mode for MPC5E on page 374](#)

Configuring the License Mode for Specific Enhanced MPCs on MX Series Routers

Starting with Junos OS Release 14.2, you can set the license mode for enhanced MPCs such as MPC4E, MPC5E, and MPC6. Configuring the license mode enables you to distinguish between an MPC with an IR license and an MPC with an R license after the MPC is installed on the router. An MPC with an R license supports all the Layer 2, Layer 2.5, and Layer 3 features. An MPC with an IR license offers partial support for these features. For more information about the license variants, see [“License Modes for Enhanced MPCs Overview” on page 372](#)



NOTE: The license mode settings are used only to provide information. You cannot set or alter the license of the MPC when you configure the license mode.

Before you configure the license mode of the MPC, verify the license of the MPC. You will need this information to configure the license mode.

Do not try to set the license mode while the card is rebooting or the following error message will appear: **Card not online or TRIO/DPC based.**

To configure the license mode for MPCs on MX Series routers:

1. Configure the license mode for the MPC in a specified MPC slot.

If the MPC has an IR license, configure the license mode as IR. If the MPC has an R license, configure the license mode of the MPC as R.

[edit]

```
user@host# set chassis fpc slot-number ir-mode ir-mode
```

2. In configuration mode, verify the configuration, for example:

[edit]

```
user@host# show chassis
```

```
fpc 1 {
  ir-mode IR;
}
```

3. After verifying the license mode, commit the changes by using the **commit** statement.

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

Release History Table

Release	Description
14.2	Starting with Junos OS Release 14.2, you can set the license mode for enhanced MPCs such as MPC4E, MPC5E, and MPC6.

Related Documentation

- [Example: Configuring the License Mode for MPC5E on page 374](#)
- [License Modes for Enhanced MPCs Overview on page 372](#)
- *ir-mode*

Example: Configuring the License Mode for MPC5E

This example describes how to configure the license mode for MPC5E on the MX480 router. It also describes how to remove the license mode settings and reconfigure the license mode settings on a new slot.

- [Requirements on page 374](#)
- [Overview on page 374](#)
- [Configuration on page 375](#)
- [Verification on page 377](#)

Requirements

This example uses the following hardware and software components:

- Junos OS Release 14.2 or later for MX Series routers
- A single MX480 router with MPC5E with R license

Overview

Configuring the license mode for an MPC enables you to distinguish between an MPC with an IR license and an MPC with an R license after the MPC is installed on the router.



NOTE: The license mode settings are used only to provide information. You cannot set or alter the license of the MPC when you configure the license mode.

The license mode settings are set specific to each MPC slot. If the MPC is installed in a different slot, or moved to another device, the license mode settings must be reconfigured on the new slot or device. Also, the license mode settings configured previously must be

removed. You can view the license mode settings from both configuration mode and operational mode.

Topology

In this example, an MPC5E is installed in slot 4 of an MX480 router and has an R license. The R license indicates that all Layer 2, Layer 2.5, and Layer 3 features are supported on the MPC. You first configure the license mode of the MPC5E in slot 4 to R. After configuring the license mode, you can verify the license mode settings. You then install the MPC5E in slot 2 of the same router. License mode settings are set specific to each MPC slot. Therefore, the license mode setting must be reconfigured. After you move the MPC5E, delete the license mode setting on slot 4 and then reconfigure the license mode setting on slot 2.

Configuration

To configure the license mode for the MPC5E according to the topology specified in the overview section, perform these tasks:

- [Configuring the License Mode for MPC5E in Slot 4 on page 375](#)
- [Deleting the License Mode for MPC5E in Slot 4 on page 376](#)
- [Configuring the License Mode for MPC5E in Slot 2 on page 376](#)

Configuring the License Mode for MPC5E in Slot 4

Step-by-Step Procedure

To configure the license mode for the MPC5E in slot 4:

1. Configure the license mode R for the MPC5E in slot 4:

```
[edit]
user@host# set chassis fpc 4 ir-mode R
```
2. In configuration mode, verify the configuration.

```
user@host# show chassis fpc 4
pic 0 {
  power off;
}
pic 1 {
  power off;
}
ir-mode R;
```
3. After verifying the license mode, commit the changes by using the **commit** statement.

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

Deleting the License Mode for MPC5E in Slot 4

Step-by-Step Procedure

To delete the license mode R for the MPC5E in slot 4:

1. Delete the license mode for the MPC5E.

```
[edit]
user@host# delete chassis fpc 4 ir-mode R
```
2. In configuration mode, verify the configuration.

```
user@host# show chassis fpc 4
pic 0 {
  power off;
}
pic 1 {
  power off;
}
```
3. After verifying the license mode, commit the changes by using the **commit** statement.

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

Configuring the License Mode for MPC5E in Slot 2

Step-by-Step Procedure

To configure the license mode for the MPC5E in slot 2:

1. Configure the license mode R for the MPC5E.

```
[edit]
user@host# set chassis fpc 2 ir-mode R
```
2. In configuration mode, verify the configuration.

```
user@host# show chassis fpc 2
pic 0 {
  power off;
}
pic 1 {
  power off;
}
ir-mode R;
```
3. After verifying the license mode, commit the changes by using the **commit** statement.

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


Verification

To confirm that you have accurately configured the license mode settings on MPC5E, perform these tasks:

- [Verifying That License Mode Is Configured for MPC5E in Slot 4 on page 377](#)
- [Verifying That the Configured License Mode Is Deleted on page 377](#)
- [Verifying That the License Mode Is Configured for MPC5E in Slot 2 on page 378](#)

Verifying That License Mode Is Configured for MPC5E in Slot 4

Purpose To verify that license mode R is configured for the MPC5E in slot 4.

Action From operational mode, enter the **show chassis hardware extensive** command.

```
user@host> show chassis hardware extensive
...
FPC 4          REV 30    750-045715  CABM2612          MPC5E 3D Q 24XGE+6XLGE
Jedec Code:    0x7fb0    EEPROM Version:  0x02
P/N:           750-045715 S/N:             CABM2612
Assembly ID:   0x0b8a    Assembly Version: 01.30
Date:          08-27-2013 Assembly Flags:   0x00
Version:       REV 30    CLEI Code:       PROTOXCLEI
ID: MPC5E 3D Q 24XGE+6XLGE FRU Model Number: PROTO-ASSEMBLY
Board Information Record:
Address 0x00: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
I2C Hex Data:
Address 0x00: 7f b0 02 fe 0b 8a 01 1e 52 45 56 20 33 30 00 00
Address 0x10: 00 00 00 00 37 35 30 2d 30 34 35 37 31 35 00 00
Address 0x20: 53 2f 4e 20 43 41 42 4d 32 36 31 32 00 1b 08 07
Address 0x30: dd ff ff ff ff ff ff ff ff ff ff ff ff ff ff
Address 0x40: ff ff ff ff 01 50 52 4f 54 4f 58 43 4c 45 49 50
Address 0x50: 52 4f 54 4f 2d 41 53 53 45 4d 42 4c 59 00 00 00
Address 0x60: 00 00 00 00 00 00 41 30 30 ff ff ff ff ff ff ff
Address 0x70: ff ff ff c2 ff ff ff ff ff ff ff ff ff ff ff ff
R/IR Mode: R
...
```

Meaning License mode **R** is configured for the MPC5E in slot 4.

Verifying That the Configured License Mode Is Deleted

Purpose To verify that the configured license mode is deleted.

Action From operational mode, enter the **show chassis hardware extensive** command.

```
user@host> show chassis hardware extensive
...
FPC 4          REV 30    750-045715  CABM2612          MPC5E 3D Q 24XGE+6XLGE
Jedec Code:    0x7fb0    EEPROM Version:  0x02
P/N:           750-045715 S/N:             CABM2612
```

```

Assembly ID: 0x0b8a      Assembly Version: 01.30
Date:          08-27-2013  Assembly Flags: 0x00
Version:       REV 30      CLEI Code:      PROTOXCLEI
ID: MPC5E 3D Q 24XGE+6XLGE  FRU Model Number: PROTO-ASSEMBLY
Board Information Record:
Address 0x00: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
I2C Hex Data:
Address 0x00: 7f b0 02 fe 0b 8a 01 1e 52 45 56 20 33 30 00 00
Address 0x10: 00 00 00 00 37 35 30 2d 30 34 35 37 31 35 00 00
Address 0x20: 53 2f 4e 20 43 41 42 4d 32 36 31 32 00 1b 08 07
Address 0x30: dd ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
Address 0x40: ff ff ff ff 01 50 52 4f 54 4f 58 43 4c 45 49 50
Address 0x50: 52 4f 54 4f 2d 41 53 53 45 4d 42 4c 59 00 00 00
Address 0x60: 00 00 00 00 00 00 41 30 30 ff ff ff ff ff ff ff
Address 0x70: ff ff ff c2 ff ff ff ff ff ff ff ff ff ff ff ff
...

```

Meaning The license mode setting has been removed for the MPC5E in slot 4.

Verifying That the License Mode Is Configured for MPC5E in Slot 2

Purpose To verify that license mode R is configured for the MPC5E in slot 2.

Action From operational mode, enter the **show chassis hardware extensive** command.

```

user@host> show chassis hardware extensive
...
FPC 2          REV 30    750-045715  CABM2612          MPC5E 3D Q 24XGE+6XLGE
Jedec Code:    0x7fb0    EEPROM Version: 0x02
P/N:           750-045715 S/N:             CABM2612
Assembly ID:   0x0b8a    Assembly Version: 01.30
Date:          08-31-2013 Assembly Flags: 0x00
Version:       REV 30    CLEI Code:      PROTOXCLEI
ID: MPC5E 3D Q 24XGE+6XLGE  FRU Model Number: PROTO-ASSEMBLY
Board Information Record:
Address 0x00: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
I2C Hex Data:
Address 0x00: 7f b0 02 fe 0b 8a 01 1e 52 45 56 20 33 30 00 00
Address 0x10: 00 00 00 00 37 35 30 2d 30 34 35 37 31 35 00 00
Address 0x20: 53 2f 4e 20 43 41 42 4d 32 36 31 32 00 1b 08 07
Address 0x30: dd ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
Address 0x40: ff ff ff ff 01 50 52 4f 54 4f 58 43 4c 45 49 50
Address 0x50: 52 4f 54 4f 2d 41 53 53 45 4d 42 4c 59 00 00 00
Address 0x60: 00 00 00 00 00 00 41 30 30 ff ff ff ff ff ff ff
Address 0x70: ff ff ff c2 ff ff ff ff ff ff ff ff ff ff ff ff
R/IR Mode: R
...

```

Meaning License mode **R** is configured for the MPC5E in slot 2.

Related Documentation

- [License Modes for Enhanced MPCs Overview on page 372](#)

- [Configuring the License Mode for Specific Enhanced MPCs on MX Series Routers on page 373](#)

Troubleshooting Software Installation

- [Troubleshooting Software Installation on page 381](#)
- [Troubleshooting a Switch That Has Booted from the Backup Junos OS Image on page 384](#)
- [Disk Space Management for Junos OS Installation on page 385](#)
- [Verifying PIC Combinations on page 386](#)

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 381](#)
- [Rebooting from the Inactive Partition on page 382](#)
- [Freeing Disk Space for Software Installation on page 383](#)
- [Installation from the Boot Loader Generates 'cannot open package' Error on page 383](#)

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 142](#) 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=diskxsys:
```

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 \(CLI Procedure\) on page 90](#)
- [Upgrading Software on an EX6200 or EX8200 Standalone Switch Using Nonstop Software Upgrade \(CLI Procedure\)](#)
- [Installing Software on EX Series Switches \(J-Web Procedure\)](#)
- [Understanding Software Installation on EX Series Switches on page 59](#)
- [show system storage partitions \(EX Series Switches Only\) on page 571](#)

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 147](#)
 - [Troubleshooting Software Installation on page 381](#)
 - [show system storage partitions \(EX Series Switches Only\) on page 571](#)

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 <https://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 <https://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 207](#)
- [Storage Media Names for SRX Series Devices on page 212](#)

CHAPTER 14

Configuration Statements

- [auto-configuration on page 390](#)
- [auto-configuration \(System\) on page 391](#)
- [auto-image-upgrade on page 393](#)
- [auto-snapshot on page 394](#)
- [autoinstallation on page 395](#)
- [autoinstallation \(JNU Satellite Devices\) on page 396](#)
- [bootp on page 397](#)
- [commit on page 398](#)
- [configuration-servers on page 399](#)
- [delete-after-commit \(JNU Satellites\) on page 400](#)
- [file \(App Engine Virtual Machine Management Service\) on page 401](#)
- [flag \(App Engine Virtual Machine Management Service\) on page 403](#)
- [interfaces \(Autoinstallation\) on page 404](#)
- [level \(App Engine Virtual Machine Management Service\) on page 405](#)
- [license on page 406](#)
- [traceoptions \(App Engine Virtual Machine Management Service\) on page 408](#)
- [traceoptions \(System License\) on page 409](#)
- [vmhost on page 411](#)
- [vmhost management-if disable on page 412](#)
- [vmhost management-if link-mode on page 413](#)
- [vmhost management-if speed on page 414](#)
- [usb on page 415](#)
- [vmhost on page 416](#)

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 279• Example: Configuring Autoinstallation on SRX Series Devices on page 290

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 279• Example: Configuring Autoinstallation on SRX Series Devices on page 290
------------------------------	---

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"> • <i>Upgrading Software by Using Automatic Software Download</i> • Understanding Software Installation on EX Series Switches on page 59 • <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 187• show system auto-snapshot on page 543

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 290](#)

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 300• Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 298• Configuring Autoinstallation on JNU Satellite Devices on page 295• Verifying Autoinstallation on JNU Satellite Devices on page 301• delete-after-commit (JNU Satellites) on page 400• 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 290](#)

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 300• Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group on page 298• Configuring Autoinstallation on JNU Satellite Devices on page 295• Verifying Autoinstallation on JNU Satellite Devices on page 301• autoinstallation on page 396• <i>configuration-servers</i>

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 <i>number</i>—(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 408](#)
- [level \(App Engine Virtual Machine Management Service\) on page 405](#)
- [flag \(App Engine Virtual Machine Management Service\) on page 403](#)

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 408 • file (App Engine Virtual Machine Management Service) on page 401

- [level \(App Engine Virtual Machine Management Service\) on page 405](#)

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 279](#)
 - [Example: Configuring Autoinstallation on SRX Series Devices on page 290](#)

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 408 • flag (App Engine Virtual Machine Management Service) on page 403 • file (App Engine Virtual Machine Management Service) on page 401

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 405](#)
- [flag \(App Engine Virtual Machine Management Service\) on page 403](#)
- [file \(App Engine Virtual Machine Management Service\) on page 401](#)

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

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"> • Routing Engines with VM Host Support on page 303 • Disabling Autorecovery on Routing Engines with VM Host Support on page 322

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 411](#)
- [vmhost management-if speed on page 414](#)
- [vmhost management-if link-mode on page 413](#)
- [show vmhost management-if on page 617](#)

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>  NOTE: The command is supported on the routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines only. </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 411 • vmhost management-if disable on page 412 • vmhost management-if speed on page 414 • show vmhost management-if on page 617

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 411](#)
- [vmhost management-if disable on page 412](#)
- [vmhost management-if link-mode on page 413](#)
- [show vmhost management-if on page 617](#)

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 290

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">• Routing Engines with VM Host Support on page 303• Disabling Autorecovery on Routing Engines with VM Host Support on page 322

CHAPTER 15

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 download
- request system software recover-from-restore-point
- request system software restore-point
- 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
- 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 rollback
- show system software restore-point-status
- show system software usb-software-version
- show system snapshot
- show system snapshot (Junos OS with Upgraded FreeBSD)
- show system snapshot media
- show system storage partitions
- 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 559
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 534](#)

List of Sample Output [request system autorecovery state save on page 421](#)
[request system autorecovery state recover on page 421](#)
[request system autorecovery state clear on page 421](#)

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 426](#)
- [request system download pause on page 424](#)
- [request system download resume on page 425](#)
- [request system download clear on page 423](#)

List of Sample Output [request system download abort on page 422](#)

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 426 • request system download pause on page 424 • request system download resume on page 425 • request system download abort on page 422
List of Sample Output	request system download clear on page 423
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 426](#)
- [request system download resume on page 425](#)
- [request system download abort on page 422](#)
- [request system download clear on page 423](#)

List of Sample Output [request system download pause on page 424](#)

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 426](#)
- [request system download pause on page 424](#)
- [request system download abort on page 422](#)
- [request system download clear on page 423](#)

List of Sample Output [request system download resume on page 425](#)

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 424• request system download resume on page 425• request system download abort on page 422• request system download clear on page 423
List of Sample Output	request system download start on page 427
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><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>
	<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 436
List of Sample Output	request system firmware upgrade on page 428
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 438
List of Sample Output	request system halt on page 430
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	<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 9.5 for SRX Series devices.</p> <p>Command introduced in Junos OS Release 11.1 for the QFX Series.</p> <p>Added additional information section on XML RPC in Junos OS Release 17.4.</p>
Description	Add a license key.
Options	<p><i>filename</i>—License key from a file or URL. Specify the filename or the URL where the key is located.</p> <p><i>terminal</i>—License key from the terminal.</p>
Additional Information	<p>Although the <code> display xml rpc</code> filter returns “xml rpc equivalent of this command is not available,” the following RPC is supported for license installation:</p> <pre><rpc> <request-license-add> <key-data> key </key-data> </request-license-add> </rpc></pre> <p>Where <i>key-data</i> is the license key data.</p> <pre><rpc> <request-license-add> <source> key-file </source> </request-license-add> </rpc></pre> <p>Where <i>source</i> is the URL of the source license key file.</p>
Required Privilege Level	maintenance
List of Sample Output	request system license add on page 432
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 qcsbja okbuqe rcmxnq vjocwf uxfsta
          z5ufjb kdrmt6 57bimv 2f3ddp qttdcn 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	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. Option license-identifier-list introduced in Junos OS Release 13.1.
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	<i>license-identifier</i> —Text string that uniquely identifies a license key. license-identifier-list [<i>licenseid001</i> <i>licenseid002</i> <i>licenseid003</i>....] —Delete multiple license identifiers as a list enclosed in brackets. all —Delete all licenses on the device.
Required Privilege Level	maintenance
Related Documentation	

request system license save


Syntax	<code>request system license save (<i>filename</i> terminal)</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>Command introduced in Junos OS Release 9.5 for SRX Series devices.</p> <p>Added additional information section on XML RPC in Junos OS Release 17.4.</p>
Description	Save installed license keys to a file or URL.
Options	<p><i>filename</i>—License key from a file or URL. Specify the filename or the URL where the key is located.</p> <p><i>terminal</i>—License key from the terminal.</p>
Additional Information	<p>Although the <code> display xml rpc</code> filter returns “xml rpc equivalent of this command is not available,” the following RPC is supported for saving installed license keys to a file or URL:</p> <pre><rpc> <request-license-save> <destination>destination</destination> </request-license-save> </rpc></pre> <p>Where <i>destination</i> is the URL of the destination license key file.</p>
Required Privilege Level	maintenance
List of Sample Output	request system license save on page 435
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:</div> <ul style="list-style-type: none">• For Juniper Agile Licensing (JAL) keys: https://license.juniper.net/• For non-JAL keys: https://ae1.juniper.net
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 436 request system license update trial on page 436
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>  <p>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 “Release Information for Junos OS with Upgraded FreeBSD” on page 32.</p> </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 437 request system partition compact-flash (If No) on page 437
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	<code>request system power-off</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	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 430
List of Sample Output	request system power-off on page 438
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 440](#)
 [Syntax \(EX Series Switches\) on page 440](#)
 [Syntax \(TX Matrix Router\) on page 440](#)
 [Syntax \(TX Matrix Plus Router\) on page 440](#)
 [Syntax \(MX Series Router\) on page 440](#)

Syntax request system reboot
 <at *time*>
 <both-routing-engines>
 <in *minutes*>
 <media (compact-flash | disk | removable-compact-flash | usb)>
 <message "*text*">
 <other-routing-engine>

Syntax (EX Series Switches) request system reboot
 <all-members>
 <at *time*>
 <both-routing-engines>
 <in *minutes*>
 <local>
 <media (external | internal)>
 <member *member-id*>
 <message "*text*">
 <other-routing-engine>
 <slice *slice*>

Syntax (TX Matrix Router) request system reboot
 <all-chassis | all-lcc | lcc *number* | scc>
 <at *time*>
 <both-routing-engines>
 <in *minutes*>
 <media (compact-flash | disk)>
 <message "*text*">
 <other-routing-engine>

Syntax (TX Matrix Plus Router) request system reboot
 <all-chassis | all-lcc | lcc *number* | sfc *number*>
 <at *time*>
 <both-routing-engines>
 <in *minutes*>
 <media (compact-flash | disk)>
 <message "*text*">
 <other-routing-engine>
 <partition (1 | 2 | alternate)>

Syntax (MX Series Router) request system reboot
 <all-members>
 <at *time*>
 <both-routing-engines>


```

<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.



NOTE: Starting from Junos OS Release 17.2R1, PTX10008 routers do not support the **request system reboot** command to reboot the Junos OS software package or bundle on the router. Use the **request vmhost reboot** command instead. See [request vmhost reboot](#).

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	<ul style="list-style-type: none">• <i>clear system reboot</i>• <i>request system halt</i>• Routing Matrix with a TX Matrix Plus Router Solutions Page• request vmhost reboot on page 588
List of Sample Output	request system reboot on page 444 request system reboot (at 2300) on page 444 request system reboot (in 2 Hours) on page 444 request system reboot (Immediately) on page 444 request system reboot (at 1:20 AM) on page 445
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	Command introduced in Junos OS Release 9.0 for EX Series switches. Option partition changed to slice in Junos OS Release 10.0 for EX Series switches.
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 <i>member-id</i>—(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 <i>member-id</i>—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	<p>Command introduced in Junos OS Release 15.1 for MX240, MX480, MX960, MX2010, and MX2020 routers and EX9200 switches.</p> <p>Command introduced in Junos OS Release 15.1X53-D30 for QFX5200 switches.</p>
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 465](#)
- [show system snapshot \(Junos OS with Upgraded FreeBSD\) on page 568](#)
- *clear system reboot*
- *request system halt*
- [Release Information for Junos OS with Upgraded FreeBSD on page 32](#)

List of Sample Output

- [request system reboot on page 450](#)
- [request system reboot \(at 2300\) on page 450](#)
- [request system reboot \(in 2 Hours\) on page 451](#)
- [request system reboot \(Immediately\) on page 451](#)
- [request system reboot \(at 1:20 AM\) on page 451](#)

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">• Release Information for Junos OS with Upgraded FreeBSD on page 32• Changes in Disk Volumes for Junos OS with Upgraded FreeBSD on page 39• Changes in Use of Snapshots for Junos OS with Upgraded FreeBSD on page 41

List of Sample Output [request system recover junos-volume \(While booted on the junos volume\) on page 453](#)
[request system recover junos-volume \(While booted on the oam volume\) on page 453](#)
[request system recover oam-volume on page 453](#)

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 455](#)
- [request system scripts rollback on page 456](#)
- [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 454](#)
- [request system scripts rollback on page 456](#)
- *request system scripts event-scripts*

request system scripts rollback

Syntax	<code>request system scripts rollback</code>
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 454• request system scripts delete on page 455• <i>request system scripts event-scripts</i>

request system snapshot

List of Syntax	Syntax on page 457 Syntax (ACX Series Routers) on page 457 Syntax (EX Series Switches) on page 457 Syntax (MX Series Routers) on page 457 Syntax (TX Matrix Routers) on page 457 Syntax (TX Matrix Plus Routers) on page 457
Syntax	request system snapshot <partition>
Syntax (ACX Series Routers)	request system snapshot <media type> <partition>
Syntax (EX Series Switches)	request system snapshot <all-members local member <i>member-id</i> > <media type> <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 [“Release Information for Junos OS with Upgraded FreeBSD” on page 32](#).

- 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 187.](#)

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 [“Release Information for Junos OS with Upgraded FreeBSD”](#) on page 32.

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 information in [“Release Information for Junos OS with Upgraded FreeBSD” on page 32](#).

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 [“Release Information for Junos OS with Upgraded FreeBSD” on page 32](#).

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 465](#)
- [show system snapshot on page 565](#)
- [show system auto-snapshot on page 543](#)

List of Sample Output

- [request system snapshot \(Routers\) on page 462](#)
- [request system snapshot \(EX Series Switches\) on page 462](#)
- [request system snapshot \(When the Partition Flag Is On\) on page 463](#)
- [request system snapshot \(MX104 Routers When Media Device is Missing\) on page 463](#)
- [request system snapshot \(When Mirroring Is Enabled\) on page 463](#)
- [request system snapshot all-lcc \(Routing Matrix\) on page 463](#)
- [request system snapshot all-members \(Virtual Chassis\) on page 463](#)

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.
 Command introduced for all platforms using Junos OS with upgraded FreeBSD. To find which platforms in which releases use Junos with upgraded FreeBSD, see [Feature Explorer](#), enter **freebsd**, and select **Junos kernel upgrade to FreeBSD 10+**.

Description Junos OS with upgraded FreeBSD has two types of snapshots: recovery snapshots and non-recovery snapshots. Non-recovery snapshots are the ones taken with the **request system snapshot** command. Non-recovery snapshots are essentially lists of software components and configuration files, which can be helpful when major software or configuration changes are occurring and establishment of a known stable system baseline is required.

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.

The snapshot script (which is the script that generates output for non-recovery snapshots) does not generate XML output. In such cases, the `<output>` tag is used.

```
user@host> request system snapshot | display xml
<rpc-reply xmlns:junos="http://xml.juniper.net/junos/18.1I0/junos">
  <output>
    NOTICE: Snapshot snap.20180105.165049 created successfully
  </output>
  <cli>
    <banner></banner>
  </cli>
</rpc-reply>
```

This is documented in `<rpc-reply>` in the *Junos XML Management Protocol Developer Guide*.

Required Privilege Level maintenance

Related Documentation

- [Changes in Use of Snapshots for Junos OS with Upgraded FreeBSD on page 41](#)

List of Sample Output [request system snapshot recovery on page 466](#)
[request system snapshot delete on page 467](#)
[request system snapshot on page 467](#)

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
<config-partition
<media (compact-flash | hard-disk | internal | usb)>
<partition>
<root-partition
<factory>
<node (all | local | node-id | primary)>
<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**
- **config-partition**—(Optional) Specify the configuration partition to be 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.
- **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

- **root-partition**(Optional) Specify the root partition to be included in the snapshot.

- **factory**— (Optional) Specifies that only the files shipped from the factory are included in the snapshot.
- **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**—Specify for primary nodes.
- **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
List of Sample Output	request system snapshot config-partition on page 469 request system snapshot root-partition on page 469 request system snapshot media hard-disk on page 470 request system snapshot media usb (when usb device is missing on page 470 request system snapshot media compact-flash on page 470 request system snapshot partition on page 470
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system snapshot config-partition

```
user@host> request system snapshot config-partition
Doing the initial labeling...
Verifying compatibility of destination media partitions...
Running newfs (391MB) on hard-disk media /config partition (ad1s1e)...
Copying '/dev/ad0s1e' to '/dev/ad1s1e' .. (this may take a few minutes)
The following filesystems were archived: /config
```

request system snapshot root-partition

```
user@host> request system snapshot root-partition
Doing the initial labeling...
Verifying compatibility of destination media partitions...
```

```
Running newfs (3GB) on hard-disk media / partition (ad1s1a)...
Copying '/dev/ad0s1a' to '/dev/ad1s1a' .. (this may take a few minutes)
The following filesystems were archived: /
```

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/dals1a' .. (this may take a few minutes)
Copying '/dev/da0s2e' to '/dev/dals1e' .. (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/dals1a' to '/dev/da0s1a' .. (this may take a few minutes)
Copying '/dev/dals1e' 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 471](#)

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 472](#)
 [Syntax \(EX Series Switches\) on page 472](#)
 [Syntax \(TX Matrix Router\) on page 472](#)
 [Syntax \(TX Matrix Plus Router\) on page 473](#)
 [Syntax \(MX Series Router\) on page 473](#)
 [Syntax \(QFX Series\) on page 473](#)
 [Syntax \(OCX Series\) on page 474](#)

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>
<partition>
<reboot>
<unlink>
<upgrade-with-config>

```

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: Starting from Junos OS Release 17.2R1, PTX10008 routers do not support the `request system software add` command. Starting from Junos OS Release 17.4R1, PTX10016 routers do not support the `request system software add` command. Use the `request vmhost software add` command instead of the `request system software add` command on the PTX10008 and PTX10016 routers to install or upgrade the Junos OS software package or bundle on the router. See [request vmhost software add](#).



NOTE: Starting from Junos OS Release 18.1R1, MX10003 routers do not support the `request system software add` command. Use the `request vmhost software add` command instead of the `request system software add` command on the MX10003 routers to install or upgrade the Junos OS software package or bundle on the router. See [request vmhost software add](#).



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 (not available for limited editions).
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, `jkernl`, last. Add the operating system package, `jkernl`, 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/jkernl
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.

When you install a software package and request a system reboot or install a package that rebuilds the Junos OS schema, such as an OpenConfig package, the device deletes all ephemeral configuration data in the process. To restore the ephemeral configuration data, you must load and commit the data to the ephemeral configuration database.

Required Privilege Level maintenance

Related Documentation

- *Format for Specifying Filenames and URLs in Junos OS CLI Commands*
- [request system software delete on page 489](#)
- [request system software rollback on page 498](#)
- [request system storage cleanup on page 511](#)
- [Upgrading Software on page 79](#)
- *Upgrading Software on a QFabric System*
- *Managing Satellite Software Upgrade Groups in a Junos Fusion*
- [request system software add \(Maintenance\) on page 486](#)
- *Routing Matrix with a TX Matrix Plus Router Solutions Page*

List of Sample Output

[request system software add validate on page 482](#)
[request system software add /var/tmp/ no-validate on page 482](#)
[request system software add no-copy no-validate reboot on page 483](#)
[request system software add validate-on-host on page 483](#)
[request system software add \(Mixed EX4200 and EX4500 Virtual Chassis\) on page 484](#)
[request system software add component all \(QFabric Systems\) on page 485](#)
[request system software add upgrade-group \(Junos Fusion\) on page 485](#)

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 /var/tmp/ no-validate

```
user@host> request system software add no-validate
/var/tmp/junos-install-mx-x86-32-15.1R1.9.tgz
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 ...

```

request system software add no-copy no-validate reboot

```

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 ...

```

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...
PACKAGETYPE: 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 488
List of Sample Output	request system software configuration-backup on page 487
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 487• <i>Performing a QFabric System Recovery Installation on the Director Group</i>
List of Sample Output	request system software configuration-restore on page 488
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 489](#)
 [Syntax \(TX Matrix Router\) on page 489](#)
 [Syntax \(TX Matrix Plus Router\) on page 489](#)

Syntax `request system software delete software-package`
 `<force>`
 `<reboot>`
 `<set [package-name package-name]>`
 `<upgrade-group [all | upgrade-group-name]>`
 `<version version-string>`

Syntax (TX Matrix Router) `request system software delete software-package`
 `<force>`
 `<lcc number | scc>`
 `<reboot>`
 `<set [package-name package-name]>`

Syntax (TX Matrix Plus Router) `request system software delete software-package`
 `<force>`
 `<lcc number | sfc number>`
 `<reboot>`
 `<set [package-name package-name]>`

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.
 set [*package-name package-name*] option added in Junos OS Release 12.2 for M Series, MX Series, and T Series routers.
 reboot option introduced in Junos OS Release 12.3.
 Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
 upgrade-group, and **version** options introduced in Junos OS Release 14.2R3 for Junos Fusion.

Description Remove a software package or bundle from the router or switch.



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.

Options ***software-package***—Software package or bundle name. You can delete any or all of the following software bundles or packages:

- **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 472](#)
- [request system software rollback on page 498](#)
- [request system software validate on page 504](#)
- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)

List of Sample Output [request system software delete jdocs on page 491](#)

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 download


Syntax (QFabric System)	request system software download <i>path package-name</i>
Release Information	Command introduced in Junos OS Release 11.3 for the QFX Series.
Description	Download a software package from a location on the Director device, mounted external USB flash drive, remote FTP or SCP location, or other location.
Options	<p><i>path</i>—Location where the software package is located. For example:</p> <ul style="list-style-type: none"> • <i>/pbdata/packages/package-name</i>—For a software package that is being installed from a local directory on the switch. • <i>protocol://hostname/pathname/package-name</i>—For a software package or bundle that is to be downloaded and installed from a remote location. Replace <i>protocol</i> with one of the following: <ul style="list-style-type: none"> • <i>ftp</i>—File Transfer Protocol. Use <i>ftp://hostname/pathname/package-name</i>. To specify authentication credentials, use <i>ftp://<username>:<password>@hostname/pathname/package-name</i>. To have the system prompt you for the password, specify <i>prompt</i> in place of the password. If a password is required, and you do not specify the password or <i>prompt</i>, an error message is displayed. • <i>scp</i>—Secure copy (available only for Canada and U.S. version). Use <i>scp://hostname/pathname/package-name</i>. To specify authentication credentials, use <i>scp://<username>:<password>@hostname/pathname/package-name</i>.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none"> • request system software add on page 472 • request system software delete on page 489 • request system software rollback on page 498 • request system storage cleanup on page 511 • Upgrading Software on page 79 • Upgrading Software on a QFabric System
List of Sample Output	request system software download on page 494
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system software download

```
user@switch> request system software download
ftp://ftp.install-directory/jinstall-qfabric-11.3X30.6.rpm
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100  186M  100  186M    0     0  18.4M      0  0:00:10  0:00:10 --:--:-- 18.6M
```

request system software recover-from-restore-point

Syntax	request system software recover-from-restore-point
Release Information	Command introduced in Junos OS Release 14.1X53-D15 for the QFX Series.
Description	Rollback to a previously created restore-point.
	<div>  NOTE: Rolling back to a previously created restore-point might disrupt traffic, as both Director devices reboot from the restore-point partition. </div>
Required Privilege Level	configure
Related Documentation	<ul style="list-style-type: none"> request system software restore-point on page 496
List of Sample Output	request system software recover-from-restore-point on page 495

Sample Output

request system software recover-from-restore-point

```

root@qfabric> request system software recover-from-restore-point
Start Restore
Checking if the restore-point exists
LogVol100 has the root filesystem
Found Restore-Point:  Fri Aug 15 07:42:39 UTC 2014 /dev/VolGroup00/LogVol103
Mounting restore-volume LogVol103
Checking the sanity of restore-point
Checking if the restore DB is present
Checking if the restore grub is present
Checking the current state of the system
Checking the state of cluster services
Checking the inventory
Checking if the peer is reachable
Checking if peer is reachable via Compute Node Monitor
Successfully communicated with peer over 169.254.0.2
Intimating the peer to do stage INITIATE_PEER_INITIAL_STAGE of downgrade
Preparing the system to downgrade
Prepping all Junos devices
Checking status at Peer
Downgrade first stage at peer concluded successfully
Initiating final stage of downgrade in peer
Intimating the peer to do stage INITIATE_PEER_FINAL_STAGE of downgrade
Modify loader to boot from restore-point
Move mount points to new filesystem
Force Reboot
Rebooting....

```

request system software restore-point

Syntax	<code>request system software restore-point</code>
Release Information	Command introduced in Junos OS Release 14.1X53-D15 for the QFX Series.
Description	Creates a restore-point. A restore-point is a snapshot of the software on the QFabric system as well as the configuration that can be rolled back to in cases where a software upgrade or configuration changes have made the QFabric system unstable or inoperable.
Required Privilege Level	configure
Related Documentation	<ul style="list-style-type: none">• request system software recover-from-restore-point on page 495
List of Sample Output	request system software restore-point on page 496

Sample Output

request system software restore-point

```
root@qfabric> request system software restore-point
Checking if director-device upgrade is currently in progress.
Checking VM status.
Checking for communication between director devices.
Checking inventory status of all components.
Checking Server INE passwords.
Checking FC passwords.
Checking CCPC passwords.
Checking FM-0 passwords.
Checking DRE-0 passwords.
Checking NW-NG-0 passwords.
Checking chassis alarms.
0
sent command to peer to start operation
sanity checks passed
Performing fdisk
restore partition created
creating restore partition on physical disk
device /dev/sda: start 0 size -388718592
gpt: 0 slices
dos: 4 slices
# 1:      63-    208844 (    208782 sectors,    106 MB)
# 2:    208845-1048771394 (1048562550 sectors, 536864 MB)
# 3: 1048771395-1146446594 ( 97675200 sectors,  50009 MB)
# 4: 1146446595-2146460714 (1000014120 sectors, 512007 MB)
performing physical volume creation
Physical volume "/dev/sda4" successfully created
"/dev/sda4" is a new physical volume of "476.84 GB"
PV Name                /dev/sda4
extending volume group 00
Volume group "VolGroup00" successfully extended
Creating Logical Volume
Logical volume "LogVol03" created
```



```
LV Name /dev/Vo1Group00/LogVo103
Restore volume selected is /dev/Vo1Group00/LogVo103
Formatting restore volume
mke2fs 1.39 (29-May-2006)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
62504960 inodes, 124993536 blocks
6249676 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=4294967296
3815 block groups
32768 blocks per group, 32768 fragments per group
16384 inodes per group
Superblock backups stored on blocks:
 32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
102400000

Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 22 mounts or
180 days, whichever comes first. Use tune2fs -c or -i to override.
/dev/Vo1Group00/LogVo103: UUID="a9fafbaf-da3c-417f-bd53-def01fbf3936"
SEC_TYPE="ext2" TYPE="ext3"
Restore Volume mounted
backing up root filesystem..this will take a few minutes
Copying files from tmp..this may take a few minutes
Dumping databases...this may take a few minutes
backing up shared filesystem..this may take a few minutes
Restore point creation finished for dg0 on /dev/Vo1Group00/LogVo103
waiting 10 mins for peer dg to finish
Restore point creation success on both DGs
```

request system software rollback

List of Syntax	Syntax on page 498 Syntax (EX Series Switches) on page 498 Syntax (TX Matrix Router) on page 498 Syntax (TX Matrix Plus Router) on page 498 Syntax (MX Series Router) on page 498
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 “[Release Information for Junos OS with Upgraded FreeBSD](#)” on [page 32](#).

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 472](#)
- [request system software delete on page 489](#)
- [request system software validate on page 504](#)
- *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 502](#)

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</code> <code><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. Command introduced in Junos OS Release 17.4R1 for SRX4100 and SRX4200 devices.
Description	Revert to the software that was loaded at the last successful request system software add command. The FreeBSD 11 Junos OS image provides an option to save a recovery image in an Operation, Administration, and Maintenance (OAM) partition, but that option will save only the Junos OS image, not the Linux image. If a user saves the Junos OS image and recovers it later, it might not be compatible with the Linux software loaded on the system.
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 504 Syntax (TX Matrix Router) on page 504 Syntax (TX Matrix Plus Router) on page 504 Syntax (MX Series Router) on page 504
Syntax	<code>request system software validate <i>package-name</i></code> <code><on (host <i>host</i> <username <i>username</i>> routing-engine <i>routing-engine</i>)></code> <code><set [<i>package-name package-name</i>]></code> <code><upgrade-with-config></code> <code><upgrade-with-config-format <i>format</i>></code>
Syntax (TX Matrix Router)	<code>request system software validate <i>package-name</i></code> <code><lcc <i>number</i> scc></code> <code><on (host <i>host</i> <username <i>username</i>> routing-engine <i>routing-engine</i>)></code> <code><set [<i>package-name package-name</i>]></code> <code><upgrade-with-config></code> <code><upgrade-with-config-format <i>format</i>></code>
Syntax (TX Matrix Plus Router)	<code>request system software validate <i>package-name</i></code> <code><lcc <i>number</i> sfc <i>number</i>></code> <code><on (host <i>host</i> <username <i>username</i>> routing-engine <i>routing-engine</i>)></code> <code><set [<i>package-name package-name</i>]></code> <code><upgrade-with-config></code> <code><upgrade-with-config-format <i>format</i>></code>
Syntax (MX Series Router)	<code>request system software validate <<i>package-name</i>></code> <code><member <i>member-id</i>></code> <code><on (host <i>host</i> <username <i>username</i>> routing-engine <i>routing-engine</i>)></code> <code><set [<i>package-name package-name</i>]></code> <code><upgrade-with-config></code> <code><upgrade-with-config-format <i>format</i>></code>
Release Information	Command introduced before Junos OS Release 7.4. 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. set [<i>package-name package-name</i>] option added in Junos OS Release 12.2 for M Series, MX Series, T Series routers. 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. Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series. 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. Command introduced in Junos OS Release 17.4 for EX Series switches.
Description	Validate candidate software against the current configuration of the router, the switch, or a remote host.

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 472](#)
- [request system software delete on page 489](#)
- [request system software rollback on page 498](#)
- [Routing Matrix with a TX Matrix Plus Router Solutions Page](#)

List of Sample Output [request system software validate \(Successful Case\) on page 506](#)
[request system software validate \(Failure Case\) on page 507](#)

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	<code>request system software validate on</code> <code><host <i>host-name</i> [username <i>user-name</i>]></code> <code><routing-engine (re0 re1)></code>
Release Information	Command introduced in Junos OS Release 15.1 for supported platforms. See Feature Explorer .
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 449• show system snapshot (Junos OS with Upgraded FreeBSD) on page 568• Release Information for Junos OS with Upgraded FreeBSD on page 32
List of Sample Output	request system software validate on host on page 509 request system software validate on routing-engine on page 509
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

```

```
v1ans': not found  
mgd: commit complete  
Validation succeeded
```

request system storage cleanup

List of Syntax	Syntax on page 511 Syntax (EX Series Switches) on page 511 Syntax (MX Series Router) on page 511 Syntax (QFX Series) on page 511 Syntax (SRX Series) on page 511
Syntax	<pre>request system storage cleanup <dry-run> <no-confirm> <re0 re1 routing-engine (backup both local master other)></pre>
Syntax (EX Series Switches)	<pre>request system storage cleanup <all-members> <dry-run> <local> <member member-id> <no-confirm> <re0 re1 routing-engine (backup both local master other)> <satellite [slot-id slot-id device-alias alias-name]></pre>
Syntax (MX Series Router)	<pre>request system storage cleanup <all-members> <dry-run> <local> <member member-id> <no-confirm> <re0 re1 routing-engine (backup both local master other)> <satellite [slot-id slot-id device-alias alias-name]></pre>
Syntax (QFX Series)	<pre>request system storage cleanup <component (serial number UUID all)> <director-group name> <dry-run> <infrastructure name> <interconnect-device name> <name-tag name-tag> <no-confirm> <node-group name> <prune> <qfabric (component name) dry-run name-tag repository> <repository (core log)> <re0 re1 routing-engine (backup both local master other)></pre>
Syntax (SRX Series)	<pre>request system storage cleanup <dry-run> <no-confirm> <re0 re1 routing-engine (backup both local master other)></pre>

- 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 514 request system storage cleanup on page 515 request system storage cleanup director-group (QFabric Systems) on page 515 request system storage cleanup infrastructure device-name (QFabric Systems) on page 517 request system storage cleanup interconnect-device device-name (QFabric Systems) on page 518 request system storage cleanup node-group group-name (QFabric Systems) on page 519 request system storage cleanup qfabric component device-name (QFabric Systems) on page 520 request system storage cleanup qfabric component device-name repository core (QFabric Systems) on page 520 request system storage cleanup qfabric component all (QFabric Systems) on page 520
Output Fields	Table 43 on page 514 describes the output fields for the request system storage cleanup command. Output fields are listed in the approximate order in which they appear.

Table 43: 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/DCF-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.lock	
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.lock	

```

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 522 request system storage cleanup on page 524
Output Fields	Table 43 on page 514 describes the output fields for the request system storage cleanup command. Output fields are listed in the approximate order in which they appear.

Table 44: 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/eedebug_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 upgraded FreeBSD kernel (10+), 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 Junos with upgraded FreeBSD.

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 457](#)
- *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 249](#)
- [Reverting to the Default Factory Configuration by Using the request system zeroize Command on page 249](#)

List of Sample Output [request system zeroize on page 527](#)
[request system zeroize media on page 528](#)

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 50000000 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)
```

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 141
List of Sample Output	show chassis hardware detail on page 531 show chassis usb storage on page 531

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	Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 9.0 for EX Series switches. Command supported in Junos OS Release 12.2 for ACX Series Universal Access Routers. Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
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 533
Output Fields	Table 45 on page 533 describes the output fields for the show system autoinstallation status command. Output fields are listed in the approximate order in which they appear.

Table 45: show system autoinstallation status Output Fields

Field Name	Field Description
Autoinstallation status	<p>Display autoinstallation status information:</p> <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 420
List of Sample Output	show system autorecovery state on page 534
Output Fields	Table 46 on page 534 lists the output fields for the show system autorecovery state command. Output fields are listed in the approximate order in which they appear.

Table 46: 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	Saved	Passed	None
s4	Saved	Passed	None

show system boot-messages

List of Syntax	Syntax on page 536 Syntax (EX Series Switches) on page 536 Syntax (TX Matrix Router) on page 536 Syntax (TX Matrix Plus Router) on page 536 Syntax (MX Series Router) on page 536 Syntax (QFX Series) on page 536
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 <code>/var/run/dmesg.boot</code> .
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 538](#)
[show system boot-messages lcc \(TX Matrix Router\) on page 539](#)
[show system boot-messages \(TX Matrix Plus Router\) on page 540](#)
[show system boot-messages \(QFX3500 Switch\) on page 540](#)

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 187
List of Sample Output	show system auto-snapshot on page 544
Output Fields	Table 47 on page 543 describes the output fields for the show system auto-snapshot command. Output fields are listed in the approximate order in which they appear.

Table 47: 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 426 Understanding Download Manager for SRX Series Devices on page 52
List of Sample Output	show system download on page 545 show system download 1 on page 546
Output Fields	Table 48 on page 545 lists the output fields for the show system download command. Output fields are listed in the approximate order in which they appear.

Table 48: 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	show system license <installed key-content <i>filename</i> keys revoked-info usage>
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>Command introduced in Junos OS Release 13.3 for the MX104 3D Universal Edge Routers.</p> <p>Customer ID added to output of data center users in Junos OS Release 15.1.</p> <p>Corrected output for duration of license added in Junos OS Release 17.4R1.</p>
Description	Display licenses and information about how they are used.
Options	<p>none—Display all license information.</p> <p>key-content <i>filename</i>—(Optional) Display license key contents of the specified filename.</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>revoked-info—(Optional) Display information about revoked licenses.</p> <p>usage—(Optional) Display the state of licensed features.</p>
Required Privilege Level	maintenance
Related Documentation	
List of Sample Output	<p>show system license (vMX Routers with Juniper Agile Licensing) on page 548</p> <p>show system license on page 549</p> <p>show system license installed on page 550</p> <p>show system license keys on page 550</p> <p>show system license usage on page 550</p> <p>show system license (MX104 Routers) on page 550</p> <p>show system license installed (MX104 Routers) on page 551</p> <p>show system license keys (MX104 Routers) on page 551</p> <p>show system license usage (MX104 Routers) on page 551</p> <p>show system license (MX104 Routers) on page 551</p> <p>show system license installed (MX104 Routers) on page 552</p> <p>show system license keys (MX104 Routers) on page 552</p> <p>show system license usage (MX104 Routers) on page 552</p> <p>show system license (MX104 Routers) on page 553</p> <p>show system license installed (MX104 Routers) on page 553</p>

[show system license keys \(MX104 Routers\) on page 553](#)
[show system license usage \(MX104 Routers\) on page 554](#)
[show system license \(QFX Series\) on page 554](#)
[show system license \(QFX5110 Switch with Disaggregated Feature License\) on page 554](#)
[show system license key-content srx_1year_sub.lic on page 555](#)

Output Fields Table 49 on page 548 lists the output fields for the **show system license** command. Output fields are listed in the approximate order in which they appear.

Table 49: 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 key-content srx_1year_sub.lic`

```
License Key Content:
License Id: LICENSE-1
License version: 4
Valid for device: CW2716AF0740
Features:
  idp-sig          - IDP Signature
  date-based, 2016-07-03 00:00:00 GMT - 2017-07-03 00:00:00 GMT
```

show system license (View)

Syntax	<code>show system license</code> <code><installed keys status usage></code>
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 362
List of Sample Output	<p>show system license on page 557</p> <p>show system license installed on page 557</p> <p>show system license keys on page 558</p> <p>show system license usage on page 558</p> <p>show system license status logical-system all on page 558</p>
Output Fields	Table 50 on page 556 lists the output fields for the show system license command. Output fields are listed in the approximate order in which they appear.

Table 50: 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 50: 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 419](#)

List of Sample Output [show system login logout on page 559](#)

Output Fields [Table 51 on page 559](#) lists the output fields for the **show system login logout** command. Output fields are listed in the approximate order in which they appear.

Table 51: 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 rollback

Syntax `show system rollback number`
`<compare number>`

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 14.1X53-D20 for OCX Series switches.
 Command introduced in Junos OS Release 11.1 for the QFX Series.

Description Display the contents of a previously committed configuration, or the differences between two previously committed configurations.



NOTE: The `show system rollback` command is a purely operational mode command and cannot be issued with `run` from the configuration mode.

Options *number*—Number of a configuration to view. The output displays the configuration. The range of values is 0 through 49.

compare number —(Optional) Number of another previously committed (rollback) configuration to compare to rollback *number*. The output displays the differences between the two configurations. The range of values is 0 through 49.

Required Privilege Level view

List of Sample Output [show system rollback compare on page 560](#)

Sample Output

show system rollback compare

```
user@host> show system rollback 3 compare 1
[edit]
+ interfaces {
+   ge-1/1/1 {
+     unit 0 {
+       family inet {
+         filter {
+           input mf_plp;
+         }
+         address 10.1.1.1/10;
+       }
+     }
+   }
+   ge-1/2/1 {
+     unit 0 {
+       family inet {
+         filter {
```



```
+           input mf_plp;
+       }
+       address 10.1.1.1/10;
+   }
+ }
+ ge-1/3/0 {
+     unit 0 {
+         family inet {
+             filter {
+                 input mf_plp;
+             }
+             address 10.1.1.1/10;
+         }
+     }
+ }
+ }
```

show system software restore-point-status

Syntax `show system software restore-point-status`

Release Information Command introduced in Junos OS Release 14.1X53-D15 for the QFX Series.

Description Display the status of the restore point for the QFabric system. A restore-point contains both a snapshot of the software and a configuration file for the QFabric system. You can only create one restore-point. When you create a new restore-point, the existing restore-point, if available, is erased.

Required Privilege Level view

Related Documentation

- [request system software restore-point on page 496](#)

List of Sample Output [show system software restore-point status on page 562](#)

Output Fields [Table 52 on page 562](#) lists the output fields for the `show system software restore-point status` command. Output fields are listed in the approximate order in which they appear.

Table 52: show system software restore-point status Output Fields

Field Name	Field Description
Member	Name of the Director device.
Creation Time	Time when the restore-point was created.
Status	Status of restore-point creation.
Restore volume	Name and path to restore volume used to create the restore-point.

Sample Output

show system software restore-point status

```

user@qfabric> show system software restore-point status
Member  Creation Time  Status  Restore volume
-----
dg0    Aug 15 07:42:39 2014  success  /dev/VolGroup00/LogVol103
dg1    Aug 15 07:42:27 2014  success  /dev/VolGroup00/LogVol103

```

show system software usb-software-version

Syntax `show system software usb-software-version`

Release Information Command introduced in Junos OS Release 14.1X53-D40 for the QFX Series.

Description (QFabric systems only) Display the version of software present on a standard USB installer key attached to each Director Group (DG) device.

Additional Information When issuing the **show system software usb-software-version** command, the USB installer key must be attached to either or both DGs.

The format of the USB installer key (including partitions) must conform to the standard specifications of the Juniper-provided USB installer.

A Juniper-provided or Juniper-recommended USB installer device should have the following partitions:

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1	*	1	75	295244	4	FAT16 <32M
/dev/sdb2		76	709	2496058	83	Linux

Required Privilege Level view

Related Documentation

- *Performing a Nonstop Software Upgrade on the QFabric System*
- *Verifying Nonstop Software Upgrade for QFabric Systems*
- *Upgrading Software on a QFabric System*

List of Sample Output [show system software usb-software-version on page 564](#)

Output Fields [Table 53 on page 563](#) lists the output fields for the **show system software usb-software-version** command. Output fields are listed in the approximate order in which they appear.

Table 53: show system software usb-software-version Output Fields

Field Name	Field Description
Node	Node supporting the device.
Device	Device on which the software is present.
Version	Version of the software present.
Filename	Software filename.

Sample Output

show system software usb-software-version

```
user@host> show system software usb-software-version

      NODE      DEVICE      FILENAME      VERSION
      ----      -
      dg0    /dev/sdb    14.1-20160516_x141X53_vjqfd.0
jinstall-qfabric-14.1-20160516_x141X53_vjqfd.0.rpm
      dg1    /dev/sdb    14.1-20160516_x141X53_vjqfd.0
jinstall-qfabric-14.1-20160516_x141X53_vjqfd.0.rpm
```

show system snapshot

List of Syntax [Syntax on page 565](#)
 [Syntax \(EX Series Switches\) on page 565](#)

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 “[Release Information for Junos OS with Upgraded FreeBSD](#)” on [page 32](#).

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 457](#)

List of Sample Output

- [show system snapshot \(Router\) on page 566](#)
- [show system snapshot media external \(Switch\) on page 566](#)
- [show system snapshot media internal \(Switch\) on page 567](#)

Output Fields Table 54 on page 566 lists the output fields for the **show system snapshot** command. Output fields are listed in the approximate order in which they appear.

Table 54: 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 show system snapshot

Release Information Command introduced starting in Junos OS Release 15.1 for supported platforms. See [Feature Explorer](#).
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 465](#)
- [request system reboot \(Junos OS with Upgraded FreeBSD\) on page 449](#)
- [Release Information for Junos OS with Upgraded FreeBSD on page 32](#)

List of Sample Output

[show system snapshot on page 568](#)
[show system snapshot \(updated as of Junos OS Release 17.2\) on page 568](#)

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 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 220
List of Sample Output	show system snapshot media internal on page 570 show system snapshot media usb on page 570

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

List of Syntax	Syntax (EX Series) on page 571 Syntax (SRX Series) on page 571
Syntax (EX Series)	show system storage partitions <all-members> <local> <member <i>member-id</i> >
Syntax (SRX Series)	show system storage partitions
Release Information	Command introduced in Junos OS Release 10.2 for SRX300, SRX320, SRX340, SRX345, and SRX550HM devices. Command introduced in Junos OS Release 11.1 for EX Series switches.
Description	Display information about the disk partitioning scheme.
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 147 • Example: Installing Junos OS on SRX Series Devices Using the Partition Option on page 196
List of Sample Output	show system storage partitions (EX Series) on page 572 show system storage partitions (SRX Series, Dual Root Partitioning) on page 572 show system storage partitions (SRX Series, Single Root Partitioning) on page 573 show system storage partitions (SRX Series, USB) on page 573
Output Fields	Table 55 on page 572 describes the output fields for the show system storage partitions command. Output fields are listed in the approximate order in which they appear.

Table 55: 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.
Partitions information	Information about partitions on the boot media: <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 (EX Series)

```

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 (SRX Series, 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 (SRX Series, 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 (SRX Series, 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	<code>show version</code> <code><brief detail></code> <code><node <i>node-id</i> local primary></code>
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	none —Display standard information about the hostname and version of the software running on the device. brief —Display brief output. detail —Display detailed output. node <i>node-id</i> —Display the software version on a specific node. Range: 0 through 1 local —Display the software version on the local node. primary —Display the software version on the primary node.
Required Privilege Level	view
List of Sample Output	show version on page 574 show version for MX Series Junos OS with Upgraded FreeBSD on page 574 show version for SRX Series Junos OS with Upgraded FreeBSD on page 575

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]
```

show version for MX Series Junos OS with Upgraded FreeBSD

```
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]

```

...

show version for SRX Series Junos OS with Upgraded FreeBSD

```

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]

```

...

CHAPTER 16

VM Host Administration 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 511](#)

Output Fields When you enter this command, you are provided feedback on the status of your request.

request vmhost file-copy

Syntax	<code>request vmhost file-copy (crash log) j-node <i>host file-name</i> v-junos <i>host file-name</i></code>
Release Information	<p>Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.</p> <p>Command 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>
	<p> NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.</p> <p>Command introduced in Junos OS Release 17.1R1 for EX9200 switches.</p>
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	<p>crash—Files in <code>/var/crash</code> on the host.</p> <p>from j-node <i>filename</i>—Name of the host file to be copied.</p> <p>log—Files in <code>/var/log</code> on the host.</p> <p>to v-junos <i>filename</i>—Name of the Junos OS file to which the host file is copied.</p>
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 580

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 411](#)

List of Sample Output [request vmhost halt on page 582](#)

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 411](#)

List of Sample Output [request vmhost hard-disk-test on page 583](#)

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	request vmhost power-off <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 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 411](#)

List of Sample Output [request vmhost power-off on page 585](#)

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 585](#)
- [request vmhost reboot on page 588](#)

List of Sample Output [request vmhost power-on on page 587](#)

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 440](#)
- [request vmhost power-on on page 587](#)

List of Sample Output [request vmhost reboot on page 589](#)

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.
 Command introduced in Junos OS Release 17.2R1 for the PTX1000 routers.

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.

For PTX1000 routers, since there is only one RE, re0|re1 is not supported.

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 625](#)

List of Sample Output [request vmhost snapshot on page 591](#)
[request vmhost snapshot recovery on page 591](#)

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 597](#)

request vmhost software add

List of Syntax [MX Series on page 593](#)
 [PTX Series on page 593](#)

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.



NOTE: You must load the PTX1000, PTX10008, and PTX10016 routers only with **junos-vmhost-install-x.tgz** package using the **request vmhost software add** command. The **junos-vmhost-install-x.tgz** package upgrades both the host software and Junos OS. The PTX1000, PTX10008, and PTX10016 routers does not support Junos only upgrade. If you try to load Junos only image, then these routers goes down or vmhost commands does not work or router state is unpredictable.



NOTE: To install Junos OS on the PTX1000, PTX10008, and PTX10016 routers, you must specify the `junos-vmhost-install-x.tgz` package in the `request vmhost software add` command. The `junos-vmhost-install-x.tgz` package upgrades both the host software and Junos OS. The PTX1000, PTX10008, and PTX10016 routers do not support Junos OS-only upgrade. If you try to load a Junos OS-only image, then these routers do not function properly or their state is unpredictable, or the VM host commands do not work.

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) (not available for limited editions).
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 472](#)
 - [request vmhost software rollback on page 599](#)

List of Sample Output [request vmhost software add \(Multiple Packages\) on page 596](#)

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) (not available for limited editions).

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 598](#)

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 593](#)
- [request vmhost software abort on page 592](#)

List of Sample Output [request vmhost software rollback on page 600](#)

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.
Command introduced in Junos OS Release 17.2R1 for PTX1000 routers.

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 590](#)

List of Sample Output [request vmhost zeroize on page 603](#)

Sample Output

request vmhost zeroize

```
user@host> request vmhost zeroize
VMHost Zeroization : Erase all data, including configuration and log files ?
[yes,no] (no) yes

re0:
-----
warning: Vmhost will reboot and may not boot without configuration
warning: Proceeding with vmhost zeroize
Zeroize secondary internal disk ...
Proceeding with zeroize on secondary disk
Mounting device in preparation for zeroize...
Cleaning up target disk for zeroize ...
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...
Cleaning up target disk for zeroize ...
Zeroize done on target disk.
Zeroize of primary disk completed
Zeroize done
mv: cannot stat '/tmp/zero-UytUWY/tgt_jlvmrootfs/etc/fstab': No such file or
directory mv: cannot stat '/tmp/zero-UytUWY/fstab': No such file or directory mv:
cannot stat '/tmp/zero-6gvrWj/tgt_jlvmrootfs/etc/fstab': No such file or directory
mv: cannot stat '/tmp/zero-6gvrWj/fstab': No such file or directory
warning: Proceeding with vmhost reboot

*** FINAL System shutdown message from root@user ***

System going down IMMEDIATELY

Initiating vmhost reboot... ok
Initiating Junos shutdown... shutdown: [pid 8565]
Shutdown NOW!
ok
Junos shutdown is in progress...
Shutdown NOW!

System shutdown time has arrived\x07\x07

re1:
-----
warning: Vmhost will reboot and may not boot without configuration
warning: Proceeding with vmhost zeroize
Zeroize secondary internal disk ...
Proceeding with zeroize on secondary disk
```


CHAPTER 17

VM Host 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 607](#)

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 608](#)

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	<p>disk <i>disk-name</i>— Display the name of the SSD.</p> <p>status—Display the status of the test.</p>
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none"> • request vmhost hard-disk-test on page 583
List of Sample Output	show vmhost hard-disk-test on page 609

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
data collection:                ( 30) seconds.
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 612](#)

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 614](#)

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
<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 **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 615](#)

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 617](#)

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 619](#)

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 621](#)

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 623](#)

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 599](#)
 - [request vmhost snapshot on page 590](#)

List of Sample Output [show vmhost snapshot on page 626](#)

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 628](#)

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 630](#)

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 632](#)

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
```