Juniper Networks | Engineering Simplicity

Junos® OS

Software Installation and Upgrade Guide

Published
2020-03-30

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

Junos® OS Software Installation and Upgrade Guide

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

Table of Contents

2

System Back Up and Recovery

12

Configuration Statements

About the Documentation

IN THIS SECTION

Use this guide for information relevant to upgrading Junos OS and related software: software packages, upgrading and downgrading Junos OS releases, system backup and recovery procedures, installing and upgrading firmware, storage media, zero touch provisioning.

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.

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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 xxvi 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 xxvi 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`
Italic text like this	• Introduces or emphasizes important new terms. • Identifies guide names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS CLI User Guide* • RFC 1997, *BGP Communities Attribute*

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

Convention	Description	Examples
Italic text like this	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# **set system domain-name** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	• To configure a stub area, include the **stub** statement at the **[edit protocols ospf area area-id]** hierarchy level. • The console port is labeled **CONSOLE**.
< > (angle brackets)	Encloses optional keywords or variables.	**stub <default-metric *metric*>;**
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	**broadcast | multicast** **(*string1* | *string2* | *string3*)**
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	**rsvp { # Required for dynamic MPLS only**
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	**community name members [*community-ids*]**

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

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		

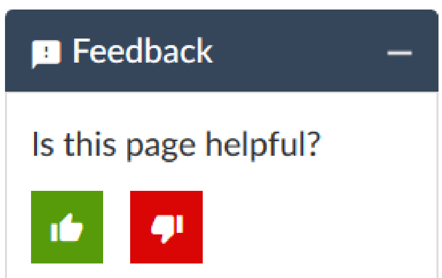
Table 2: Text and Syntax Conventions *(continued)*

Convention	Description	Examples
Bold text like this	Represents graphical user interface (GUI) items you click or select.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select **Protocols>Ospf**.

Documentation Feedback

We encourage you to provide feedback so that we can improve our documentation. You can use either of the following methods:

- Online feedback system—Click TechLibrary Feedback, on the lower right of any page on the Juniper Networks TechLibrary site, and do one of the following:



  - Click the thumbs-up icon if the information on the page was helpful to you.
  - Click the thumbs-down icon if the information on the page was not helpful to you or if you have suggestions for improvement, and use the pop-up form to provide 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 Juniper Care or Partner Support Services 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/
- Create a service request online: https://myjuniper.juniper.net

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

Creating a Service Request with JTAC

You can create a service request with JTAC on the Web or by telephone.

- Visit https://myjuniper.juniper.net.
- 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://support.juniper.net/support/requesting-support/.

1
CHAPTER

Junos OS Overview

Junos OS Overview

IN THIS SECTION

Junos OS is the single operating system that powers Juniper's broad portfolio of physical and virtual networking and security products.

Junos OS Overview

IN THIS SECTION

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.

Junos OS includes the following architecture variations:

- Junos OS FreeBSD 6 on bare metal. This is Junos OS based on a FreeBSD 6 kernel.
- Junos OS FreeBSD 10 on bare metal. This is Junos OS based on an upgraded FreeBSD kernel. Starting with Junos OS Release 15.1, certain hardware platforms run Junos OS with upgraded FreeBSD. Starting in Junos OS Release 16.1, Junos OS with upgraded FreeBSD can run as a guest virtual machine (VM) on a Linux VM host. For more on which platforms run Junos OS with upgraded FreeBSD, see "Release Information for Junos OS with Upgraded FreeBSD" on page 35.
- Junos OS Evolved

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

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.

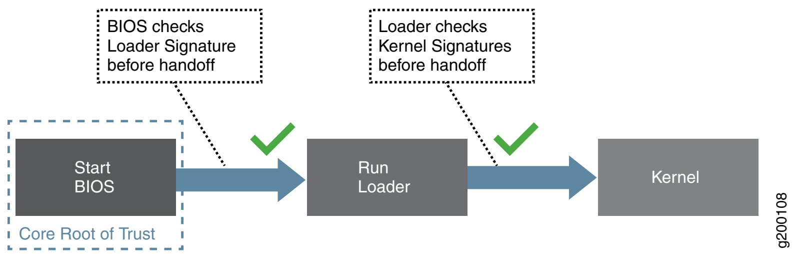
Juniper Networks routing platforms run only binaries supplied by Juniper Networks, and currently do not support third-party binaries. Each Junos OS image includes a digitally signed manifest of executables that are registered with the system only if the signature can be validated. Junos OS will not execute 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.”



Secure Boot requires no actions on your part to implement. It is implemented on supported hardware by default.

For information on which Junos OS releases and hardware support Secure Boot, see Feature Explorer and enter **Secure Boot**.

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.

Junos OS with Upgraded FreeBSD

IN THIS SECTION

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 a clean-slate implementation of Junos OS on top of a pristine (minimally modified) and current version of FreeBSD. Another advantage of using the upgraded FreeBSD is access to sophisticated processing, efficiency, and security features which do not have to be reproduced in Junos OS.

Certain changes came with Junos OS with Upgraded FreeBSD: how Junos OS installation packages are named, some CLI commands and statements are deprecated and others are introduced, and how disk volumes and system backup (snapshots) work.

For more information on changes in Junos OS installation package names, see "Junos OS Installation Package Names" on page 110. For other changes in Junos OS with upgraded FreeBSD, see the following subsections:

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 282
> - Guest VM: "Installing, Upgrading, Backing Up, and Recovery of VM Host" on page 326

Processing Changes in Junos OS with Upgraded FreeBSD

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, the system boots up with default factory configuration. Before attempting to downgrade from Junos OS Release 17.4 to any previous releases, the IDP configuration must be deleted.

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.

Changes in Package Names for Junos OS with Upgraded FreeBSD

IN THIS SECTION

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

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 38.
- 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 39.
- 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 plaform 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 dataplace 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 41. For details on the changes listed in Table 3 on page 41, see the topics covering the specific command or statement.

For changed actions for VM hosts, see "VM Host Operations and Management" on page 350.

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

Command or Statement	Change
request system snapshot delete *snapshot*	New action
request system snapshot recovery	New action
request system snapshot load *snapshot*	New action
request system recover *volume*	New action: *volume* is either **/junos-volume** or **/oam-volume**
request system snapshot	Changed action
show system snapshot	Changed action
request system reboot (junos | network | oam | usb}	Changed action with new media options
request system reboot	Changed action
request system software validate on	Changed action
request system software rollback	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 42. 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	
request system partition abort	see Feature Explorer.
request system partition compact-flash	
request system partition hard-disk	
request system snapshot <config-partition>	
request system snapshot <root-partition>	
request system snapshot <slice>	
request system software delete-backup	
request system software rollback <force>	
show system processes providers	
show system snapshot <slice>	
Deprecated Configuration Statement	
set system mirror-flash-on-disk	see Feature Explorer.

SEE ALSO

request system snapshot (Junos OS with Upgraded FreeBSD) | 603
show system snapshot (Junos OS with Upgraded FreeBSD) | 760
request system reboot (Junos OS with Upgraded FreeBSD) | 580

Changes in Disk Volumes for Junos OS with Upgraded FreeBSD

IN THIS SECTION

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

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 44.

/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 44.

Changes in Use of Snapshots for Junos OS with Upgraded FreeBSD

IN THIS SECTION

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

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 exits 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) | 760
request system snapshot (Junos OS with Upgraded FreeBSD) | 603

RELATED DOCUMENTATION

Junos OS Installation Package Names | 110
Junos OS Editions | 115
Upgrading and Downgrading to Junos with Upgraded FreeBSD | 281

2

CHAPTER

System Back Up and Recovery

Backing Up an Installation Using Snapshots

IN THIS SECTION

The installation process removes all stored files on the device except the juniper.conf and SSH files. Therefore, you should back up your current configuration in case you need to return to the current software installation after running the installation program. You can also recover the configuration file and the Junos OS if required.

Understanding How to Back Up an Installation on Switches

IN THIS SECTION

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.

On devices running Junos OS Evolved, this command only copies the contents of the **/soft** directory, which includes the running version of Junos OS Evolved.

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 Chasis 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.

Creating a Snapshot and Using It to Boot a QFX Series Switch

IN THIS SECTION

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.

> **NOTE:** EX4600 and QFX Series products except for QFabric only support snapshot via external USB. QFabric does not support request system snapshot at all.

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:** EX4600 and most QFX Series switches support snapshot via external USB.

> **NOTE:** The following products do not support system snapshot: QFabric and QFX5200 and QFX10000 switches.

This topic includes the following tasks:

Creating a Snapshot on an External USB Flash Drive and Using It to Boot a QFX Series Switch

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#
```

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

IN THIS SECTION

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

> **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 an EX Series Switch

IN THIS SECTION

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

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 media usb**

Creating a Snapshot and Using It to Boot an SRX Series device

IN THIS SECTION

Creating a Snapshot and Using It to Boot an SRX Series device

IN THIS SECTION

This example shows how to configure a boot device.

Requirements

Before you begin, ensure that the backup device has a storage capacity of at least 1 GB. See “Ensuring Sufficient Disk Space for Junos OS Upgrades on SRX Devices” on page 133.

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.

Backing Up the Current Installation on SRX Series Devices

IN THIS SECTION

This topic includes the following sections:

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

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

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

When the **request system snapshot** command is issued, the **/root** file system is backed up to **/altroot**, and **/config** is backed up to **/altconfig**. The **/root** and **/config** file systems are on the devices's CompactFlash card, and the **/altroot** and **/altconfig** file systems are on the devices's hard disk or or solid-state drive (SSD). When the backup is completed, the current and backup software installations are identical.

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

user@host> request system snapshot media

Backing Up the Current Installation on SRX300, SRX320, SRX340, SRX345, SRX380, SRX550M, SRX3400, and SRX3600 Devices

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

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

user@host> request system snapshot media

Following options are supported:

- **internal**— Copies the snapshot to internal media.
- **usb**— Copies the snapshot to the USB storage device. This is the default option for SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550M devices.
- **external**— Copies the snapshot to an external storage device. This option is available for the compact flash on the SRX650 Services Gateway only.

Configuring External CompactFlash for SRX650 Devices

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

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

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

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

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

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

2. Reboot the device from the USB storage device using the **request system reboot media usb** command.

3. Go to the U-boot prompt.

4. Stop at U-boot and set the following variables:

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

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

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

Creating a Snapshot and Using It to Boot an ACX Series Router

IN THIS SECTION

Understanding System Snapshot on an ACX Series Router

The system snapshot feature enables you to create copies of the software running on an ACX Series router. You can use the system snapshot feature to take a “snapshot” of the files currently used to run the router—the complete contents of the root (/) and **/config** directories, which include the running Juniper Networks Juniper operating system (Junos OS) and the active configuration—and copy all of these files to another media, such as a universal serial bus (USB) storage device, the active slice of a dual-root partitioned router, or the alternate slice of a dual-root partitioned router.

> **NOTE:** Junos OS automatically uses the backup software if the currently running software goes bad. For example, if the **da0s1** slice goes bad, Junos OS automatically comes up using the **da0s2** slice, and takes a snapshot of the **da0s2** slice and copies it to the **da0s1** slice if the auto snapshot functionality is configured, which is disabled by default. However, you can also do this manually using the system snapshot feature.

> **NOTE:** In ACX5048 and ACX5096 routers, the system snapshot feature is applicable only when a USB storage device is used.

Typically, you can take a snapshot prior to the upgrade of an image on the dual internal NAND flash device (**da0s1** or **da0s2**), or to remedy a bad image, thereby preventing the bad image from rendering the system useless. A snapshot to another media ensures that the device can boot from the other media in case the system does not boot up from the current image.

You can take a snapshot of the currently running software and configuration on a router in the following situations:

- The router's active slice (for example, **da0s1**) is updated with a new Junos OS image (using the jinstall package). In such a case, you must update the other slice (**da0s2**) with the new image.

  > **NOTE:** The active slice can be **da0s1** or **da0s2**.

- The router's active slice (for example, **da0s1**) is corrupted and the router is rebooted from the backup slice (that is, from **da0s2**). Therefore, you must restore a new image on the active slice—that is, on **da0s1**.
- Both slices of the router's dual internal NAND flash device are corrupted and the router continues trying to reboot. In this situation, you can insert a USB storage device, boot the router from that device, and restore the NAND flash device slices—**da0s1** and **da0s2**.

  > **NOTE:** Before you attempt to take a snapshot from the USB storage device, ensure that the USB storage device contains an image of Junos OS from which the router can boot up.

SEE ALSO

> *request system snapshot (ACX Series)*

Example: Taking a Snapshot of the Software and Configuration

IN THIS SECTION

This example includes six scenarios in which you can take a snapshot of the currently running software and configuration on an ACX Series router, prior to the upgrade of an image or to remedy a bad image, thereby preventing the bad image from rendering the system useless.

Requirements

This example uses the following hardware and software components:

- One ACX Series router
- Junos OS Release 12.2 or later

Overview

In this example, the **request system snapshot** command is used to take a copy of the currently running software and configuration on another media—for example, a universal serial bus (USB) storage device, the active slice (**da0s1** or **da0s2**) of a dual-root partitioned router, or the alternate slice (**da0s1** or **da0s2**) of a dual-root partitioned router. A snapshot to another media ensures that the device can boot from the other media in case the system does not boot up from the current image.

> **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.

Taking a Snapshot

Scenario: To take a snapshot from a NAND flash device slice to a USB storage device:

1. Boot up the router from the NAND flash device and make sure that a formatted USB storage device is plugged in to the router's USB port. The USB storage device must be formatted for the root (/) and **/config** directories.

2. Issue the **request system snapshot** command.

   user@host> **request system snapshot**

   ```
   Verifying compatibility of destination media partitions...
   Running newfs (254MB) on usb media  / partition (da1s1a)...
   Running newfs (47MB) on usb media  /config partition (da1s1e)...
   Copying '/dev/da0s2a' to '/dev/da1s1a' .. (this may take a few minutes)
   Copying '/dev/da0s2e' to '/dev/da1s1e' .. (this may take a few minutes)
   The following filesystems were archived: / /config
   ```

   The root (/) and **/config** directories from the currently mounted NAND flash slice are copied to the USB storage device.

Scenario: To take a snapshot from a NAND flash device slice to a USB storage device with formatting:

1. Boot up the router from the NAND flash device and make sure that a USB storage device is plugged in to the router's USB port.

   NOTE: Formatting a USB storage device deletes all the data on the USB storage device.

2. Issue the **request system snapshot partition** command.

   user@host> **request system snapshot partition**

   ```
   clearing current label...
   Partitioning usb media (da1) ...
   Partitions on snapshot:

       Partition  Mountpoint  Size    Snapshot argument
           a      /           312MB   root-size
           e      /config     47MB    config-size
           f      /var        620MB   var-size
   Running newfs (312MB) on usb media  / partition (da1s1a)...
   Running newfs (47MB) on usb media  /config partition (da1s1e)...
   ```

```
Running newfs (620MB) on usb media  /var partition (da1s1f)...
Copying '/dev/da0s2a' to '/dev/da1s1a' .. (this may take a few minutes)
Copying '/dev/da0s2e' to '/dev/da1s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

After the USB storage device is formatted, the root (/) and **/config** directories from the currently mounted NAND flash slice are copied to the USB storage device.

Scenario: To take a snapshot from the active slice of the NAND flash device to the alternate slice:

1. Boot up the router from the NAND flash device.

2. Issue the **request system snapshot slice alternate** command.

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

   ```
   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/da0s2a' to '/dev/da0s1a' .. (this may take a few minutes)
   Copying '/dev/da0s2e' to '/dev/da0s1e' .. (this may take a few minutes)
   The following filesystems were archived: / /config
   ```

   The root (/) and **/config** directories from the currently mounted NAND flash slice are copied to the other slice.

Scenario: To take a snapshot from an active slice of the NAND flash device to the alternate slice after partitioning:

1. Boot up the router from the NAND flash device.

2. Issue the **request system snapshot partition slice alternate** command.

   user@host> **request system snapshot partition slice alternate**

   ```
   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/da0s2a' to '/dev/da0s1a' .. (this may take a few minutes)
   Copying '/dev/da0s2e' to '/dev/da0s1e' .. (this may take a few minutes)
   The following filesystems were archived: / /config
   ```

The BSD label (disk partitioning information) for the active flash slice is installed and then the root (/) and **/config** directories from the currently mounted NAND flash slice are copied to the other slice.

Scenario: To take a snapshot from a USB storage device to the active slice of the NAND flash device:

1. Boot up the router from a USB storage device containing the required Junos OS image.

2. Issue the **request system snapshot** command.

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

   ```
   Verifying compatibility of destination media partitions...
   Running newfs (439MB) on internal media  / partition (da0s1a)...
   Running newfs (46MB) on internal media  /config partition (da0s1e)...
   Copying '/dev/da1s1a' to '/dev/da0s1a' .. (this may take a few minutes)
   Copying '/dev/da1s1e' to '/dev/da0s1e' .. (this may take a few minutes)
   The following filesystems were archived: / /config
   ```

   The root (/) and **/config** directories from the USB storage device are copied to the active NAND flash slice.

Scenario: To take a snapshot from a USB storage device to the active slice of the NAND flash device after partitioning:

1. Boot up the router from a USB storage device containing the required Junos OS image.

2. Issue the **request system snapshot partition** command.

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

   ```
   Verifying compatibility of destination media partitions...
   Running newfs (439MB) on internal media  / partition (da0s1a)...
   Running newfs (46MB) on internal media  /config partition (da0s1e)...
   Copying '/dev/da1s1a' to '/dev/da0s1a' .. (this may take a few minutes)
   Copying '/dev/da1s1e' to '/dev/da0s1e' .. (this may take a few minutes)
   The following filesystems were archived: / /config
   ```

   The BSD label (disk partitioning information) for the active flash slice is installed and then the root (/) and **/config** directories from the USB storage device are copied to the active NAND flash slice.

SEE ALSO

Understanding System Snapshot on an ACX Series Router | 60

request system snapshot (ACX Series)

Creating a Snapshot and Using It to Boot an MX Router

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 282. For information on which platforms use Junos OS with upgraded FreeBSD, see "Release Information for Junos OS with Upgraded FreeBSD" on page 35.

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
--------------------------------------------------
```

SEE ALSO

request system snapshot | 590
Routing Matrix with a TX Matrix Plus Router Solutions Page

Recovery Using an Emergency Boot Device

IN THIS SECTION

If Junos OS software is damaged on your device, the emergency boot device helps you to recover the software.

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 ed 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 35.)

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 400.
- ***installMedia***—Refers to the installation media ed 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 35.)

7. Log out as **su**:

```
exit
```

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 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. Use gunzip to unzip the image file.

3. Insert a USB device into the USB port.

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

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

5. 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.

6. Enter the following command on the device:

   `root@device%` **dd if=/var/tmp/*filename* of=/dev/da1 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)
```

7. Log out of the shell:

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

Recovering the 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 for QFX Series Switches" on page 70 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 QFX Series Switch" on page 50.

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

   CAUTION: 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.

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 for Routers" on page 69
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).

   CAUTION: 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 35.

```
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 81.

Rescue and Recovery of Configuration File

IN THIS SECTION

In the event of software failure, a rescue configuration helps to load a known working configuration. No need to remember the rollback number; if you saved a configuration, you can use it anytime when needed.

Saving and Reverting a Rescue Configuration File

IN THIS SECTION

Saving a Rescue Configuration File

IN THIS SECTION

A rescue configuration file is helpful in the event that your device's configuration file has been misconfigured. A rescue configuration allows you to define a known working configuration or a configuration with a known state that you can roll back to at any time. This alleviates the necessity of having to remember the rollback number with the rollback command. 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.

As of Junos OS Release 16.1, for devices running Junos OS with upgraded FreeBSD, provided you have saved a rescue configuration on the device, there is an automatic device recovery mode that goes into action should the system fail to activate the current configuration (amnesiac mode).

> **NOTE:** To determine which platforms run Junos OS with upgraded FreeBSD, see Feature Explorer, enter **freebsd**, and select **Junos kernel upgrade to FreeBSD 10+**.

You can identify that the device has recovered automatically from amnesiac mode by the following:

- The syslog **UI_DEVICE_IN_RECOVERY_MODE** is generated, which indicates that there was a problem in the normal boot time commit and that Junos OS has activated the rescue configuration as the device's configuration.
- The CLI displays the banner **Device is running in Recovery Mode** in both operational and configuration modes.

This topic covers the following procedures:

Saving a Rescue Configuration

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.
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 as **rescue.conf.gz**.

Validating the Rescue Configuration

You can verify that the syntax of a configuration file is correct and check for commit check errors by using the **test configuration *filename*** command.

To verify if a rescue configuration file is correct:

- Issue the **test configuration *filename*** command from the CLI operational mode.

  ```
  user@host> test configuration /config/rescue.conf.gz
  configuration check succeeds
  ```

If the configuration contains any syntax or commit check errors, a message is displayed to indicate the line number and column number in which the error was found. This command only accepts text files.

Copying the Configuration to a Remote Server

This task is optional but recommended.

To copy the rescue configuration to a remote server:

1. Start the device shell.

```
user@host> start shell
```

2. Go to the **/config** directory and list the rescue configuration file..

```
% cd /config
% ls -lrt rescue.conf.gz
-rw-r----- 1 root wheel 1483 Dec 14 10:50 rescue.conf.gz
```

3. FTP the configuration file to the remote host.

   % **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.
```

Rolling Back to Troubleshoot the Failed Configuration

Your rescue configuration is probably not exactly the configuration you want or need on your system. Therefore, you will want to examine the failures that occurred when you tried to activate the current configuration and make corrective actions.

To correct the failed configuration:

1. Log in to the device through the management IP (or the console if permitted).

2. Load the failed configuration.

```
user@host# rollback 1
```

If you are doing this step right after the recovery mode, **rollback 1** will be the configuration that cause the amnesiac mode.

3. Make corrections to the configuration.

4. Do a commit check.

```
user@host># commit check
```

5. If there are other corrections to make, make them.

6. Commit configuration.

Rolling Back to the Rescue Configuration

Not all platforms run Junos OS with updated FreeBSD. Those that do not or are releases earlier than Junos OS Release 16.1, do not have the automatic recovery mode. You will need to rollback to rescue configuration manually to bring the device back to normal running mode.

To roll back to the rescue configuration:

1. Log in to the device through the console.

2. Issue the **rollback rescue** command from the configuration mode of the CLI.

   `user@host#` **rollback rescue**

```
load complete
```

3. Commit the configuration.

```
user@host#commit
```

4. Fix the failed configuration. See “Rolling Back to Troubleshoot the Failed Configuration” on page 78.

Deleting an Existing Rescue Configuration

To delete an existing rescue configuration:

- Issue the **request system configuration rescue delete** command:

```
user@host> request system configuration rescue delete
```

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
   ```

Copy Backup Configurations and Restoring a Saved Configurations

IN THIS SECTION

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
```

Restoring a Saved Configuration

To restore a saved configuration, perform the following tasks:

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.

SEE ALSO

request system zeroize | 703

Recovery of Junos OS

IN THIS SECTION

In case of failed software installation or failure after installing Junos OS such as CLI not working, you can recover the failed software. You can recover the software by installing Junos OS and remove the existing Junos OS image to install a new image.

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 Series Switch* or "Creating a Snapshot and Using It to Boot a QFX Series Switch" on page 50.

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, **tftp://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.

SEE ALSO

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

Creating a Snapshot and Using It to Boot a QFX Series Switch | 50

Recovering Junos OS on a Device Running Junos OS with Upgraded FreeBSD

Juniper Networks devices that run Junos OS with upgraded FreeBSD have two separate volumes:

- **dev/gpt/junos** (**/junos** for short) volume that is used to run Junos OS and to store the configuration and log files
- **dev/gpt/oam** (**/oam** for short), an Operations, Administration, and Maintenance (OAM) volume that is used to store a complete backup of Junos OS and the configuration.

In case of damage to the device's software or failure of the **/junos** volume, you can use the backed up software and configuration stored in the **/oam** volume to boot the system and restore Junos OS with the recovery configuration. To perform this reboot and restore the configuration, the **/oam** volume must have all of the information required to provide the system with a running configuration. This information is provided by the recovery snapshot, created using the **request system snapshot recovery** command.

> **NOTE:** You need console access to perform the following procedure to recover Junos OS.

To recover Junos OS by using the recovery snapshot stored in the **/oam** volume:

1. Power off the device, such as a router or a switch, by pressing the power button on the front panel.

2. Connect and configure the management device, such as a PC or a laptop, as follows:

   a. Turn off the power to the management device.

   b. Plug one end of the Ethernet rollover cable supplied with the device into the RJ-45–to–DB-9 serial port adapter supplied with the device.

   c. Plug the RJ-45–to–DB-9 serial port adapter into the serial port on the management device.

   d. Connect the other end of the Ethernet rollover cable to the console port on the device.

   e. Turn on the power to the management device.

   f. On the management device, start your asynchronous terminal emulation application (such as Microsoft Windows Hyperterminal) and select the appropriate communication (COM) port to use (for example, COM1).

   g. Configure the port settings as follows:

- Bits per second: 9600
- Data bits: 8
- Parity: None
- Stop bits: 1
- Flow control: None

3. Power on the device by pressing the power button on the front panel.

   Verify that the **POWER** LED on the front panel turns green.

   The terminal emulation screen on your management device displays the boot sequence of the device.

4. Access the Junos Main Menu.

   - In released before Junos OS Release 17.3, the Junos Main Menu appears for 3 seconds on startup before automatically booting the **/junos** volume. Press any key within the 3-second window to stop the automatic boot sequence and display the Junos Main Menu.

     > **NOTE:** The Junos Main Menu will appear every time you reboot the router while connected to the console.

   - Starting in Junos OS Release 17.3, press Ctrl+c within the 3-second window to stop the automatic boot sequence and display the Junos Main Menu.

   ```
   Main Menu

   1.  Boot [J]unos volume
   2.  Boot Junos volume in [S]afe mode

   3.  [R]eboot

   4.  [B]oot menu
   5.  [M]ore options

   Choice:
   ```

5. At the Choice: prompt in Junos Main Menu, enter **B** or **4** to choose **4. [B]oot menu** :

   ```
   Boot Menu
   ```

```
1. Boot [P]revious installed Junos packages
2. Boot Junos in [S]ingle user mode
3. Boot from [R]ecovery snapshot

4. Boot from [U]SB

5. Boot to [O]AM shell

6. Snapshot [B]oot menu

7. [M]ain menu

Choice:
```

6. At the Choice: prompt in Boot Menu, enter **R** or **3** to choose the **3. Boot from [R]ecovery snapshot** option. The device reboots into recovery mode. The following sample output shows the messages displayed on the terminal when you recover Junos OS on an EX2300 switch.

```
Booting from recovery snapshot ...
-
/boot/junos/boot/os-kernel/kernel data=0xe8c000 syms=[0x4+0x6b020+0x4+0x72cfe]
/boot/junos/boot/os-kernel/ex2300-48mp.dtb size=0x18b8
/boot/junos/boot/os-kernel/ex2300.dtb size=0x1e67
/boot/junos/boot/junos-modules/fips_core.ko text=0x13bc data=0x275+0x7
syms=[0x4+0x7a0+0x4+0x518]
loading required module 'netstack'
/boot/junos/boot/netstack/netstack.ko text=0x910a3c data=0x3ae2f+0x10dded
syms=[0x4+0xf0570+0x4+0xdc394]
loading required module 'crypto'

[...Output truncated...]

/var/pdb/profile_db initialized

Profile database initialized
realpath: /dev/dumpdev: No such file or directory
/etc/rc: WARNING: Dump device does not exist.  Savecore not run.
Prefetching /usr/sbin/rpd ...
Prefetching /usr/sbin/lacpd ...
Prefetching /usr/sbin/chassisd ...
mkdir: /packages/sets/active: Read-only file system
Starting jlaunchhelperd.
sysctl: unknown oid 'kern.rtc_retries'
```

```
Starting cron.

Fri Jun 22 01:25:20 PDT 2018

FreeBSD/arm (device-name) (ttyu0)

login:
```

7. Log in to the device and verify that the software is properly restored.

```
[...Output truncated...]
login: root

--- JUNOS 18.1-20180125.0 built 2018-01-25 20:34:55 UTC

root@:RE:0%
```

SEE ALSO

Changes in Disk Volumes for Junos OS with Upgraded FreeBSD | 43

How to Recover Junos OS with Upgraded FreeBSD

IN THIS SECTION

Ways to Recover Junos OS with Upgraded FreeBSD Without the Use of the CLI

IN THIS SECTION

If a device running Junos OS with upgraded FreeBSD has a damaged operating system or configuration that prevents the system from booting normally, or you need to recover the root password, the CLI is unavailable to you. But you can access and use the Junos Main Menu and Boot Menu. These menus have options such as booting from a USB device or a previously installed version of Junos OS, or using CLI Recovery mode to change you root password. To see how to access these menus, see "How to Access the Junos Main Menu, Boot Menu, and Options Menu" on page 94.

Boot from the /junos Volume

Juniper Networks devices that run Junos OS with upgraded FreeBSD have two separate volumes:

- **dev/gpt/junos** (**/junos** for short) volume that is used to run Junos OS and to store the configuration and log files
- **dev/gpt/oam** (**/oam** for short), an Operations, Administration, and Maintenance (OAM) volume that is used to store a complete backup of Junos OS and the configuration.

If a device running Junos OS with upgraded FreeBSD has a damaged operating system or configuration, preventing the system from booting normally, you can still boot from the **/junos** volume without using the CLI command **request system reboot**. Access the Junos Main Menu. Booting the **/junos** volume is the first option on the Junos Main Menu.

Boot from Safe Mode

Safe mode is a diagnostic mode of a computer's operating system that has reduced functionality, making the task of isolating problems easier since many non-core components are disabled. In Junos OS with upgraded FreeBSD, safe mode boots the entire Junos OS and FreeBSD but with a few kernel features disabled.

One other difference between normal mode and safe mode is that for EX3400 devices, symmetric multiprocessing (SMP) in normal mode uses a dual core, whereas in safe mode, it uses a single core.

An installation that has a major problem (such as disk corruption or the installation of poorly configured software) that prevents the operating system from booting into its normal operating mode may boot in safe mode and allow you to diagnose the problem.

Booting from Safe Mode is the second option on the Junos Main Menu.

Boot from a Previously Installed Release of Junos OS with Upgraded FreeBSD

With devices running Junos OS with upgraded Freebsd, you can boot from a previous release of the OS, provided there was a previous image on the device and it is still there. Previous image files can be found in the **/packages/sets/previous** directory. Some platforms do not keep an older image due to storage space limitations (for example, EX2300 and EX3400 do not have a **/packages/sets/previous** directory).

The following is sample output from an EX9200 switch, showing the previous image:

`root@:/ #` **ls -al /packages/sets/previous/**

```
total 20
drwxr-xr-x  4 root  wheel  1536 Mar 30 15:45 .
drwxrwxrwx  4 root  wheel   512 Mar 30 18:47 ..
drwxr-xr-x  2 root  wheel   512 Mar 30 15:45 boot
lrwxr-xr-x  1 root  wheel    66 Mar 30 15:44 jail-runtime ->
/packages/db/jail-runtime-x86-32-20171012.356211_builder_stable_10
lrwxr-xr-x  1 root  wheel    62 Mar 30 15:44 jdocs ->
/packages/db/jdocs-x86-32-20171121.225603_builder_junos_161_r6
lrwxr-xr-x  1 root  wheel    63 Mar 30 15:44 jpfe-X ->
/packages/db/jpfe-X-x86-32-20171121.225603_builder_junos_161_r6

...
```

To see if there are previous packages on the device, do one of the following:

- From a UNIX shell, issue the **ls /packages/sets/previous** command.
- From the CLI operational mode, use the **file list /packages/sets/previous** command.

Booting from a previously installed release of Junos OS with upgraded FreeBSD is the first option on the Boot Menu.

System boots the previous Junos OS with upgraded FreeBSD image. If there is no previous image, system boots from the currently installed image.

Boot into Single-User Mode

Single-user mode is a mode in which a multi-user computer operating system boots into a single superuser. It is mainly used for maintenance of multi-user environments.

For devices running Junos OS with upgraded FreeBSD, single-user mode puts you in a shell with a prompt. There is limited support and password recovery is not possible using this option. But you can do few file operations.

Booting into single-user mode is the second option on the Boot Menu..

Boot from a Recovery Snapshot

A recovery snapshot for devices running Junos OS with upgraded FreeBSD is taken with the **request system snapshot recovery** command. Recovery snapshots are full copies of the packages and configuration taken at the time the snapshot command is issued.

Booting from a recovery snapshot is the third option on the Boot Menu.

Boot from a USB Device

If you want to boot from a USB device, you must connect the USB device to the router or switch. Then select the fourth option on the Boot Menu. If no USB device is connected, you will see a message **No USB media found**.

Boot to the OAM Shell

The Boot to the OAM Shell option is similar to the single-user mode except that you are put into the oam shell or volume. The compact flash drive is the /oam volume and stores recovery snapshot backup information. In case of failure of the /junos volume, the /oam volume can be used to boot the system.

Booting to the oam shell is the fifth option on the Boot Menu.

CLI Recovery Mode

If you choose the CLI Recovery Mode option, you end up at a **root>** prompt. Enter **configure** at the prompt to enter the configuration CLI mode. From there you can change the root password to recover your access to the device (see *Recovering the Root Password on Routers*).

The CLI Recovery Mode is the second option on the Options Menu.

Check File System

The check file system option lets you make sure there are no issues or corrupted files. The system boots from the OAM volume to perform disk checks. This is the third option on the Options Menu.

Enable/Disable Verbose Boot

Choosing the fourth option on the Options Menu either enables verbose boot, which lets you see the whole boot scroll by, or disables verbose boot.

Boot Prompt

The Boot Prompt option displays an OK prompt from which you can type one of the following commands:

- **menu**—Takes you back to the Junos Main Menu.
- **boot-junos**—Boots the device to the current version of Junos OS.
- **reboot**—Reboots the system.

You can also type **?** at the OK prompt to see several other available commands. The boot prompt option is the fifth option on the Options Menu.

How to Access the Junos Main Menu, Boot Menu, and Options Menu

IN THIS SECTION

If a device running Junos OS with upgraded FreeBSD has a damaged operating system or configuration, preventing the system from booting normally, you can still boot using an option on the Junos Main Menu, Boot Menu, or Options Menu. The following procedures show you how to access these menus.

How to Access the Junos Main Menu

You access the Junos Main Menu by interrupting the reboot of a device.

> **NOTE:** You need console access (either direct access to console or via a console server) to perform the following procedure.

You can either perform the entire procedure or power-cycle the device and start the procedure from Step 4. (You can also perform these reboots by rebooting the device via the CLI if that is available.)

To boot a device running Junos OS with upgraded FreeBSD without using the CLI:

1. Power off the device, such as a router or a switch, by pressing the power button on the front panel.

2. Connect and configure the management device, such as a PC or a laptop, as follows:

   a. Turn off the power to the management device.

   b. Plug one end of the Ethernet rollover cable supplied with the device into the RJ-45–to–DB-9 serial port adapter supplied with the device.

   c. Plug the RJ-45–to–DB-9 serial port adapter into the serial port on the management device.

   d. Connect the other end of the Ethernet rollover cable to the console port on the device.

   e. Turn on the power to the management device.

   f. On the management device, start your asynchronous terminal emulation application (such as Microsoft Windows Hyperterminal) and select the appropriate communication (COM) port to use (for example, COM1).

   g. Configure the port settings as follows:

      - Bits per second: 9600
      - Data bits: 8
      - Parity: None
      - Stop bits: 1
      - Flow control: None

3. Power on the device by pressing the power button on the front panel.

   Verify that the **POWER** LED on the front panel turns green.

The terminal emulation screen on your management device displays the boot sequence of the device.

4. Access the Junos Main Menu.

   Do one of the following

   - Prior to Junos OS Release 17.3, the Junos Main Menu appears for three seconds on startup before automatically booting from the **/junos** volume. Press any key within the three-second interval to stop the automatic boot sequence and display the Junos Main Menu.
   - Starting in Junos OS Release 17.3, press Ctrl+c at the following part in the reboot:

     ```
     FreeBSD/x86 bootstrap loader, Revision 1.1
     (builder@feyrith.juniper.net, Sun Feb  4 13:06:24 PST 2018)
     /
     Autoboot in 1 seconds... (press Ctrl-C to interrupt)
     ```

   The Junos Main Menu is displayed:

   ```
   Main Menu

   1.  Boot [J]unos volume
   2.  Boot Junos volume in [S]afe mode

   3.  [R]eboot

   4.  [B]oot menu
   5.  [M]ore options

   Choice:
   ```

5. At the **Choice:** prompt in the Junos Main Menu, enter the number representing the option you want to use. Alternatively, you can enter the letter in square brackets to choose an option.

How to Access the Boot Menu

The Boot Menu is one of two menus you can access from the Junos Main Menu.

> **NOTE:** You need console access to perform the following procedure.

You must first access the Junos Main Menu. See "How to Access the Junos Main Menu" on page 95.

To access the Boot Menu:

1. At the **Choice:** prompt in the Junos Main Menu, enter **4** or **B** to choose **4. [B]oot menu**.

   The Boot Menu is displayed.

   ```
   Boot Menu

   1. Boot [P]revious installed Junos packages
   2. Boot Junos in [S]ingle user mode
   3. Boot from [R]ecovery snapshot

   4. Boot from [U]SB

   5. Boot to [O]AM shell

   6. Snapshot [B]oot menu

   7. [M]ain menu

   Choice:
   ```

2. At the **Choice:** prompt in the Boot Menu, enter the number representing the option you want to use. Alternatively, you can enter the letter in square brackets to choose an option.

How to Access the Options Menu

The Options Menu is one of two menus you can access from the Junos Main Menu.

> **NOTE:** You need console access to perform the following procedure.

You must first access the Junos Main Menu. See "How to Access the Junos Main Menu" on page 95.

To access the Options Menu:

1. At the **Choice:** prompt in the Junos Main Menu, enter **5** or **M** to choose **5. [M]ore options**.

   The Options Menu is displayed.

   ```
   Options Menu 4
   ```

```
1.  Recover [J]unos volume
2.  Recovery mode - [C]LI

3.  Check [F]ile system

4.  Enable [V]erbose boot

5.  [B]oot prompt

6.  [M]ain menu

Choice:
```

2. At the **Choice:** prompt in the Options Menu, enter the number representing the option you want to use. Alternatively, you can enter the letter in square brackets to choose an option.

Autorecovery of Configuration, Licenses, and Disk Information on SRX Series Devices

IN THIS SECTION

Autorecovery helps to detect and recover information on disk partitioning, configuration, and licenses in the event of disk becomes corrupted

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 lables (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 5 on page 100 lists types of autorecovery alarms, descriptions, and required actions.

Table 5: Autorecovery Alarms

Alarm	Alarm Type	Description	Action Required
Autorecovery information needs to be saved	Minor	This alarm indicates: • Unsaved data needs to be saved, or saved data contains problems and another save is required.	• Ensure that the system has all required licenses and configuration. • Execute the **request system autorecovery state save** command.
Autorecovery has recovered corrupted information	Minor	This alarm indicates: • Boot time integrity check failed for certain items; however, the items have been recovered successfully.	• No action is required. • Alarm is cleared on next bootup.

Table 5: Autorecovery Alarms *(continued)*

Alarm	Alarm Type	Description	Action Required
Autorecovery was unable to recover data completely	Major	This alarm indicates: • Boot time integrity check failed for certain items, which could not be recovered successfully.	• 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

Creating a Snapshot and Using It to Boot an SRX Series device | 56
Example: Installing Junos OS Upgrade Packages on SRX Series Devices | 255
Reverting the Junos OS Software Image Back to the Previous Version | 263

3

CHAPTER

Installing, Upgrading, and Downgrading Software

Software Installation and Upgrade Overview

IN THIS SECTION

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

Software Installation and Upgrade Overview

IN THIS SECTION

A Juniper Networks device is delivered with the Juniper Networks operating system (Junos OS) preinstalled. When you power on the device, it starts (boots) using the installed software. As new features and software fixes become available, you must upgrade your software to use them.

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 newly installed software.

Before installing software, back up the system, select the software installation package you require, and download it from the Juniper Networks downloads page. If you encounter any difficulties during software installation, you can use the recovery installation procedure to install Junos OS on the device. After a successful upgrade, back up the new existing configuration to a secondary device.

NOTE: Before installing software on a device that has one or more custom YANG data models added to it, back up and remove the configuration data corresponding to the custom YANG data models from the active configuration. For more information see "Managing YANG Packages and Configurations During a Software Upgrade or Downgrade" on page 150.

To understand more about Junos OS Software Licensing, see the Juniper Licensing Guide. Please refer to the product Data Sheets accessible from Products & Services for details, or contact your Juniper Account Team or Juniper Partner.

- For features on EX Series Switches that require license, see Understanding Software Licenses for EX Series Switches
- For features on M Series Routers that require license, see Software Features That Require Licenses on M Series Routers Only
- For features on M Series, MX Series, and T Series Routers that require license, see Software Features That Require Licenses on M Series, MX Series, and T Series Routers
- For features on MX Series Routers that require license, see Software Features That Require Licenses on MX Series Routers Only
- For features on QFX Series Switches that require license, see Software Features That Require Licenses on the QFX Series.
- For features on SRX Series devices that require license, see Software Feature Licenses for SRX Series Devices.

The following subsections introduce the overall considerations in installing Junos OS:

Types of Junos OS Installation

The three types of installations used to upgrade or downgrade your device 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 information on the different installation packages available, see "Junos OS Installation Packages Prefixes" on page 111.

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 a 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.

Backing Up the Current System's Files

Creating a backup of the current system on your device 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.

During a successful upgrade, the upgrade package completely reinstalls the existing Junos OS. It retains only the **juniper.conf** and SSH files. Other information is removed. Therefore, you should back up your existing configuration in case you need to return to it after running the installation program.

You can create copies of the software running on a device using the system snapshot feature. The system snapshot feature takes a "snapshot" of the files currently used to run the device—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 device at the next boot up or as a backup boot option. When the backup is completed, the existing and backup software installations are identical.

NOTE: Snapshots taken with the **request system snapshot** command in a Junos OS with upgraded FreeBSD system are not the same as those snapshots taken with the **request system snapshot** command in a Junos OS (as in legacy Junos OS) system. To back up your Junos OS with upgraded FreeBSD system devices, use the **request system snapshot recovery** command.

When the correct 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 devices's CompactFlash card, and the **/altroot** and **/altconfig** file systems are on the devices's hard disk or or solid-state drive (SSD).

Determining Software Installation Package

All Junos OS releases are delivered in signed packages that contain digital signatures to ensure official Juniper Networks software. To see which software packages are currently 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 can either download software to the **/var/tmp** directory of your device, or install it directly from the downloads page.

For more information about signed software packages, see the [the packages topic]

Connecting to the Console

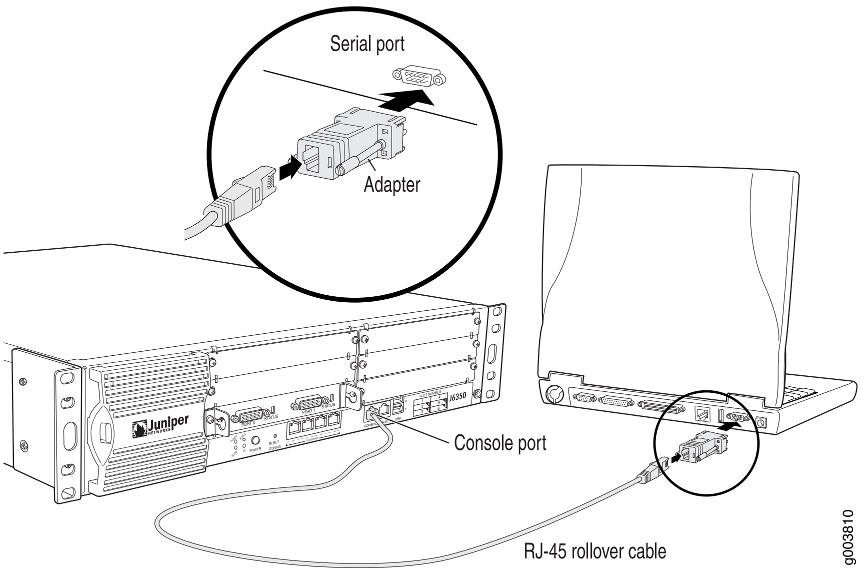
We recommend that you upgrade all individual software packages using an out-of-band connection from the console or management Ethernet interface, because in-band connections can be lost during the upgrade process.

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 1 on page 108.

Figure 1: Connecting to the Console Port on a Junos OS Device



Validating the Installation Package with the Current Configuration

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.

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.

If you do not want to validate when upgrading, you must specify the **no-validate** option.

Dual-Root and Single-Root Partitioning (SRX Series Only)

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

CAUTION: 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.

Junos OS Installation Package Names

IN THIS SECTION

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

The first part of the installation package filename is a combination of a standard prefix and product designation. Table 6 on page 111 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 6 on page 111 also indicates the prefixes used for the different platforms running Junos OS with upgraded FreesBSD.

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

junos-*product*

Table 6 on page 111 lists several examples of installation package prefixes for the SRX Series.

Table 6 on page 111 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 6: 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-acx5k*	Junos OS for the ACX5000 Series routers.
jinstall-host-nfx-2*	Junos OS with upgraded FreeBSD for NFX2xx platforms that are Linux based; this prefix indicates the image includes the host and Junos OS. 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 36.

Table 6: Installation Package Prefixes *(continued)*

Installation Package Prefix	Description
jinstall-host-qfx*	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. 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 36.
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 36.
junos-install*	Junos OS with upgraded FreeBSD for EX Series and MX Series routers that support Junos OS with upgraded FreeBSD. For specific examples of package names for Junos OS with upgraded FreeBSD, see "Changes in Package Names for Junos OS with Upgraded FreeBSD" on page 36.
junos-evo-install*	Junos OS Evolved, introduced as of Junos OS Evolved Release 18.3R1. For Junos OS Evolved, there is a single image for all fixed form (versus chassis) platforms, and a platform image name can also be distinguished as merchant silicon (ms). Here are some examples: • **junos-evo-install-qfx-ms-fixed-x84-64-*version*.iso**—Single image for all QFX platforms based on merchant silicon. It could be Broadcom family or any other vendor. • **junos-evo-install-ptx-fixed-x84-64-*version*.iso**—All fixed PTX platform variants (that is, PTX 10003, and so on) have a single ISO image. For PTX orders, this image is installed as factory default. • **junos-evo-install-qfx-fixed-x84-64-*version*.iso**—All fixed QFX platform variants have a single ISO image. For QFX orders, this image is installed as factory default. • **junos-evo-install-ptx-chassis-x84-64-*version*.iso**—One single ISO image for PTX chassis platforms. • **junos-evo-install-qfx-chassis-x84-64-*version*.iso**—One single ISO image for PTX chassis platforms.

Table 6: Installation Package Prefixes *(continued)*

Installation Package Prefix	Description
junos-evo-profile1-install*	Junos OS Evolved, introduced as of Junos OS Evolved Release 19.1R1. For example, junos-evo-profile1-install-qfx-ms-fixed-x86-64-19.1-20190226.5-EVO.iso.
junos-install*	Junos OS with upgraded FreeBSD for EX Series and MX Series routers that support Junos OS with upgraded FreeBSD. For specific examples of package names for Junos OS with upgraded FreeBSD, see "Changes in Package Names for Junos OS with Upgraded FreeBSD" on page 36.
junos-install-media-pxe-srxhe-x86* (USB) junos-install-media-usb-srxhe-x86* (PXE) junos-srxhe-x86* (CLI)	Junos OS for SRX4600
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, SRX380, and SRX550M .
junos-vmhost-install*	Junos OS with upgraded FreeBSD on VM Host.
junos-install-media-pxe-srxhe-x86* (USB) **junos-install-media-usb-srxhe-x86*** (PXE) **junos-srxhe-x86*** (CLI)	Junos OS for SRX4600

SEE ALSO

show version

Junos OS Release Numbers

NOTE: Junos OS Evolved uses the same release numbering system as Junos OS.

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, 15.1, or 17.1. 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 Junos OS for a particular Junos OS release number.

In this example, we dissect the format of the software release number to show what it indicates. The generalized format is as follows:

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 7 on page 114.
- *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 7: 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.

Table 7: Software Release Types *(continued)*

Release Type	Description
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	Special (eXception) release software. X releases follow a numbering system that differs from the standard Junos OS release numbering. Starting with Junos OS Release 12.1X44-D10, SRX Series devices follow a special naming convention for Junos OS releases. For more information, refer to the Knowledge Base article KB30092 at https://kb.juniper.net/InfoCenter/index?page=home.

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 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.

Boot Sequence on Devices with Routing Engines

IN THIS SECTION

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

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)

The router attempts to boot from the storage media in the following order:

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.

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

Storage Media and Routing Engines | 396

Preparing for Software Installation and Upgrade

IN THIS SECTION

Before you install or upgrade Junos OS, you must ensure some basic checks such as sufficient disk space availability and backing up configurations in place.

Reinstalling Junos OS

IN THIS SECTION

Checklist for Reinstalling Junos OS

Table 8 on page 120 provides links and commands for reinstalling Junos OS.

Table 8: Checklist for Reinstalling Junos OS

Tasks	Command or Action
Before You Reinstall Junos OS	
1. Log the Software Version Information on page 122	**show version | save *filename***
2. Log the Hardware Version Information on page 123	**show chassis hardware | save *filename***
3. Log the Chassis Environment Information on page 125	**show chassis environment | save *filename***
4. Log the System Boot-Message Information on page 125	**show system boot-messages | save *filename***
5. Log the Active Configuration on page 128	**show configuration | save *filename***
6. Log the Interfaces on the Router on page 128	**show interface terse | save *filename***

Table 8: Checklist for Reinstalling Junos OS *(continued)*

Tasks	Command or Action
7. Log the BGP, IS-IS, and OSPF Adjacency Information on page 129	**show bgp summary** | **save** ***filename*** **show isis adjacency brief** | **save** ***filename*** **show ospf neighbor brief** | **save** ***filename***
8. Log the System Storage Information on page 131	**show system storage** | **save** ***filename***
9. *Back Up the Currently Running and Active File System*	**request system snapshot**
"Reinstall Junos OS" on page 145	Insert the floppy and reboot the system.
"Reconfigure Junos OS" on page 146	
1. Configure Host Names, Domain Names, and IP Addresses on page 146	Log in as root. Start the CLI. Enter configuration mode: **configure** **set system host-name** ***host-name*** **set system domain-name** ***domain-name*** **set interfaces fxp0 unit 0 family inet address** ***address/prefix-length*** **set system backup-router** ***address*** **set system name-server** ***address***
2. Protecting Network Security by Configuring the Root Password on page 147	**set system root-authentication plain-text-password** **set system root-authentication encrypted-password password** **set system root-authentication ssh-rsa key** **commit** **exit**
3. Check Network Connectivity on page 149	**ping** ***address***
4. Copy Backup Configurations to the Router on page 80	**file copy var/tmp** **configure** **[edit]** **load merge /config/*filename*** **or load replace /config/*filename*** **[edit]** **commit**
After You Reinstall Junos OS	

Table 8: Checklist for Reinstalling Junos OS *(continued)*

Tasks	Command or Action
1. *Compare Information Logged Before and After the Reinstall*	**show version** | **save *filename*** **show chassis hardware** | **save *filename*** **show chassis environment** | **save *filename*** **show system boot-messages** | **save *filename*** **show configuration** | **save *filename*** **show interfaces terse** | **save *filename*** **show bgp summary** **show isis adjacency brief** **show ospf neighbor brief** | **save *filename*** **show system storage** | **save *filename***
2. *Back Up the New Software*	**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.

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
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.

Validating the Configuration Image Before Upgrading or Downgrading the Software

If you upgrade or downgrade the Junos OS image on a switch that supports configuration image validation (see Feature Explorer for feature support per EX Series switch), the system validates that the existing configuration is compatible with the new image before the actual upgrade or downgrade commences.

Benefits of image validation—If validation fails, the new image is not loaded, and an error message provides information about the failure. If you upgrade or downgrade the software on a system that does not support validation, configuration incompatibilities between the existing and new image or insufficient memory to load the new image might cause the system to lose its current configuration or go offline.

Here are some validation guidelines to keep in mind:

- Validation is set to on by default. You do not need to configure it or issue any command to start it on a switch that supports image validation. You can disable validation (the procedure is given below) and then re-enable it.
- Validation slows down the upgrade or downgrade process by as much as 7 minutes.
- Image validation is supported only on the **jinstall** package.
- If you invoke validation from an image that does not support validation, the new image is loaded but validation does not occur.
- Validation does not work in a *downgrade* to an image that does not support validation if your system is configured for graceful routing switchover (GRES) or if you run image loading without nonstop software upgrade (NSSU). See the procedure below for steps to use validation in this type of scenario.

To disable validation, re-enable or invoke validation manually, or use validation when downgrading to an image that does not support it:

- To disable validation, issue **request system software add *image-name* reboot no-validate** command.
- To re-enable or invoke validation manually, choose one of the following methods:
  - Issue **request system software add *image-name***.
  - Issue **request system software nonstop-upgrade *image-name***.
  - Issue **request system software validate** to run just configuration validation.
- To use validation when downgrading to an image that does not support it, choose one of the following methods:

- Remove the graceful-switchover configuration and then issue the **request system software add *image-name* reboot** command.
- Use NSSU by issuing the **request system software nonstop-upgrade *image-name*** command.

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-blocks       Used       Avail   Capacity   Mounted on
/dev/da0s2a             300196     154410     121772        56%  /
devfs                        1          1          0       100%  /dev
/dev/md0                409000     409000          0       100%  /junos
/cf                     300196     154410     121772        56%  /junos/cf
devfs                        1          1          0       100%  /junos/dev/
procfs                       4          4          0       100%  /proc
/dev/bo0s3e              25004         52      22952         0%  /config
/dev/bo0s3f             350628     178450     144128        55%  /cf/var
/dev/md1               171860      16804     141308        11%  /mfs
/cf/var/jail           350628     178450     144128        55%  /jail/var
/cf/var/log             350628     178450     144128        55%  /jail/var/log
devfs                        1          1          0       100%  /jail/dev
```

```
/dev/md2                 40172          4       36956          0%  /mfs/var/run/utm
/dev/md3                 1884         138       1596           8%  /jail/mfs
```

Cleaning Up the System File Storage Space

When the system file storage space on the device is full, rebooting the device does not solve the problem. The following error message is displayed during a typical operation on the device after the file storage space is full.

```
user@host% cli
user@host> configure/var: write failed, filesystem is full
```

You can clean up the file storage on the device by deleting system files using the **request system storage cleanup** command as shown in following procedure:

1. Request to delete system files on the device.

   ```
   user@host> request system storage cleanup
   ```

   The list of files to be deleted is displayed.

   ```
   List of files to delete:

            Size Date         Name
     11B Oct 28 23:40 /var/jail/tmp/alarmd.ts
    92.4K Jan 11 17:12 /var/log/chassisd.0.gz
    92.4K Jan 11 06:06 /var/log/chassisd.1.gz
    92.5K Jan 10 19:00 /var/log/chassisd.2.gz
    92.5K Jan 10 07:53 /var/log/chassisd.3.gz
    92.2K Jan 10 15:00 /var/log/hostlogs/auth.log.1.gz
    92.2K Jan  1 18:45 /var/log/hostlogs/auth.log.2.gz
    92.1K Jan  4 17:30 /var/log/hostlogs/auth.log.3.gz
    92.2K Jan  1 18:45 /var/log/hostlogs/auth.log.4.gz
    79.0K Jan 12 01:59 /var/log/hostlogs/daemon.log.1.gz
    78.8K Jan 11 23:15 /var/log/hostlogs/daemon.log.2.gz
    78.7K Jan 11 20:30 /var/log/hostlogs/daemon.log.3.gz
    79.1K Jan 11 17:44 /var/log/hostlogs/daemon.log.4.gz
    59.1K Jan 11 21:59 /var/log/hostlogs/debug.1.gz
    59.2K Jan 11 17:44 /var/log/hostlogs/debug.2.gz
    59.2K Jan 11 13:29 /var/log/hostlogs/debug.3.gz
    59.3K Jan 11 09:14 /var/log/hostlogs/debug.4.gz
   ```

```
186.6K Oct 20 16:31 /var/log/hostlogs/kern.log.1.gz
238.3K Jan 11 23:15 /var/log/hostlogs/lcmd.log.1.gz
238.4K Jan 11 17:30 /var/log/hostlogs/lcmd.log.2.gz
238.6K Jan 11 11:45 /var/log/hostlogs/lcmd.log.3.gz
238.5K Jan 11 06:00 /var/log/hostlogs/lcmd.log.4.gz
372.5K Jan 11 17:00 /var/log/hostlogs/syslog.1.gz
372.5K Jan 11 04:45 /var/log/hostlogs/syslog.2.gz
371.9K Jan 10 16:30 /var/log/hostlogs/syslog.3.gz
372.7K Jan 10 04:15 /var/log/hostlogs/syslog.4.gz
 10.1K Jan 12 02:03 /var/log/messages.0.gz
 55.1K Jan  6 21:25 /var/log/messages.1.gz
 81.5K Dec  1 21:30 /var/log/messages.2.gz
```

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

2. Enter the option **yes** to proceed with deleting of the files.

Verifying Junos OS and Boot Loader Software Versions on an EX Series Switch

IN THIS SECTION

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

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.

SEE ALSO

Downloading Software

IN THIS SECTION

Downloading Software Using a Browser

You download the software package you need from the Juniper Networks Downloads page at https://support.juniper.net/support/downloads/.

> **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. Using a Web browser, navigate to https://support.juniper.net/support/downloads/.

   The Download Results page appears.

2. Find the software package that you want to download and click the item in the Downloads column.

   A login screen appears.

3. Log in with your username and password.

4. 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 "Downloading Software Using the Command-Line Interface" on page 141 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.

Downloading Software Using the Command-Line Interface

Download the software package you need from the Juniper Networks Downloads page athttps://support.juniper.net/support/downloads/, 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
   ```

Downloading Software Using Download Manager (SRX Series Only)

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.

The download manager is supported on SRX300, SRX320, SRX340, SRX345, and SRX380 devices.

Be aware of the following considerations when using the download manager:

- 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.

NOTE: The download manager supports only the FTP and HTTP protocols.

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.

Before you begin, you must have the following:

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

To download the Junos OS image to your device:

1. 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. To verify that the file has been downloaded, use the **show system download** command. The command displays the state as "completed" when the downloaded file is ready to be installed.

3. To install the downloaded image file from the **/var/tmp** directory, use the **request system software add** command.

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

Table 9 on page 145 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 9: show system download Output Fields

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

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:

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 this account 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.

To set the root password, you have a few options as shown in 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.

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: retype password here
     ```

     As you enter a plain-text password into the CLI, Junos OS hides it from view and encrypts it immediately. You do not have to configure Junos OS to encrypt the password as in some other systems. In the resulting configuration, the encrypted password is marked as **## SECRET-DATA** so that it cannot be seen.

   - 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-ecdsa | ssh-rsa key)
```

2. If you used a configuration group, apply it with the command **set apply-groups**, replacing <*group name*> with the configuration group name.

   For example:

```
[edit]
user@host# set apply-groups <group name>
```

3. Commit the changes.

```
root@# commit
```

SEE ALSO

Accessing a Junos OS Device the First Time

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.

RELATED DOCUMENTATION

Managing YANG Packages and Configurations During a Software Upgrade or Downgrade

IN THIS SECTION

Certain devices running Junos OS enable you to load custom YANG modules on the device to add data models that are not natively supported by Junos OS. When you add, update, or delete a YANG data model, Junos OS rebuilds its schema and then validates the active configuration against the updated schema.

When you upgrade or downgrade Junos OS, by default, the system validates the software package or bundle against the current configuration. During the installation, the schema for custom YANG data models is not available. As a result, if the active configuration contains dependencies on these models, the software validation fails, which causes the upgrade or downgrade to fail.

In addition, devices that are running Junos OS based on FreeBSD version 6 remove custom YANG packages from the device during the software installation process. For this Junos OS variant, if the active configuration contains dependencies on custom YANG data models, the software installation fails even if you do not validate the software against the configuration, because the configuration data cannot be validated during the initial boot-time commit.

For these reasons, before you upgrade or downgrade the Junos OS image on a device that has one or more custom YANG modules added to it, you must remove all configuration data corresponding to the custom YANG data models from the active configuration. After the software installation is complete, add the YANG packages and corresponding configuration data back to the device, if appropriate. The tasks are outlined in this topic.

NOTE: You do not need to delete configuration data corresponding to OpenConfig packages before upgrading or downgrading Junos OS.

Backing up and Deleting the Configuration Data

If the configuration contains dependencies on custom YANG data models:

1. If you plan to restore the configuration data that corresponds to the nonnative YANG data models after the software is updated, save a copy of either the entire configuration or the configuration data corresponding to the YANG data models, as appropriate.

   - To save the entire configuration:

     ```
     user@host> show configuration | save (filename | url)
     ```

   - To save configuration data under a specific hierarchy level:

     ```
     user@host> show configuration path-to-yang-statement-hierarchy | save (filename | url)
     ```

2. In configuration mode, delete the portions of the configuration that depend on the custom YANG data models.

   ```
   [edit]
   user@host# delete path-to-yang-statement-hierarchy
   ```

3. Commit the changes.

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

4. Prior to performing the software installation, ensure that the saved configuration data and the YANG module and script files are saved to a local or remote location that will preserve the files during the installation and that will be accessible after the installation is complete.

Restoring the YANG Packages and Configuration Data

After the software installation is complete, load the YANG packages onto the device (where required), and restore the configuration data associated with the packages, if appropriate. During a software upgrade or downgrade, devices running Junos OS with upgraded FreeBSD preserve custom YANG packages, whereas devices running Junos OS based on FreeBSD version 6 delete the packages.

1. Load the YANG packages (devices running Junos OS based on FreeBSD version 6 only).

```
user@host> request system yang add package package-name module [modules] deviation-module [modules]
    translation-script [scripts] action-script [scripts]
```

2. When the system prompts you to restart the Junos OS CLI, press **Enter** to accept the default value of **yes**.

```
...
WARNING: cli has been replaced by an updated version:
...
Restart cli using the new version ? [yes,no] (yes)

Restarting cli ...
```

> **NOTE:** To prevent CLI-related or configuration database errors, we recommend that you do not perform any CLI operations, change the configuration, or abort the operation while a device is in the process of adding, updating, or deleting a YANG package and modifying the schema.

3. In configuration mode, load the configuration data associated with the YANG packages.

   For example, to load the configuration data from a file relative to the top level of the configuration statement hierarchy:

```
[edit]
user@host# load merge (filename | url)
```

> **NOTE:** For more information about loading configuration data, see the *CLI User Guide*.

4. Commit the changes.

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

RELATED DOCUMENTATION

Managing YANG Packages, Modules, and Scripts on Devices Running Junos OS

Installing Software on Routing Devices

IN THIS SECTION

Routing devices are delivered with Junos OS preinstalled on them. As new features and software fixes become available, you must upgrade Junos OS to use them. You can install software on single and redundant routing engines.

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

Before you install a new software release on a device, you should back up the current system.

In Junos OS Evolved, multiple releases of the software can be installed on the device simultaneously as long as there is space. If there is no more space, you must delete an older image of the software before installing the new one.

For more information on backing up Junos OS, see "Backing Up the Existing Installation on Routers" on page 66.

To upgrade the software on a router or switch:

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

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

   The variable ***installation-package*** is the name of the installation package. Specify the absolute path on the local disk, for example, for Junos OS Evolved, **/var/tmp/junos-linux-install-ptx.iso**. For package name prefixes, see "Junos OS Installation Packages Prefixes" on page 111.

NOTE: (Does not apply to Junos OS Evolved) To install multiple software packages at one time, you can use the **request system software add set** command. For more information on this command, see the **set** option in request system software add.

CAUTION: 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.

2. Reboot the device to start the new software:

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

NOTE: You must reboot the device to load the new software release on the device.

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

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

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

3. Log in and verify the release of the software installed:

- To verify release for installation of a Junos OS Evolved release, use the **show system software list** command.

```
user@host>  show system software list
```

- To verify release for installation of a Junos OS release, use the **show version** command.

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

SEE ALSO

request system software add | 613

show version

Installing the Software Package on a Device with Redundant Routing Engines

If the device has two Routing Engines, perform a Junos OS installation on each Routing Engine separately to minimize disruption to network operation.

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. For detailed procedures, see the following subsections:

Preparing the Device for the Installation

Determine if this is the best procedure for upgrading your device:

- 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)*.
- To upgrade two different Junos OS releases with no disruption on the control plane and with minimal disruption of traffic, you can use In-Service Software Upgrade, see *Getting Started with Unified In-Service Software Upgrade* for routers and switches, and *Upgrading a Chassis Cluster Using In-Service Software Upgrade* for SRX Series devices.
- To upgrade the software running on EX Series Ethernet Switches with redundant Routing Engines and all member switches in EX Series Virtual Chassis with a single command, you can use Nonstop Software Upgrade, see *Understanding Nonstop Software Upgrade on EX Series Switches*.
- 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 167.

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.

To ensure GRES and NSR are disabled:

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 device.

2. From the CLI operational prompt, enter configuration mode:

```
{master}
user@host> configure
Entering configuration mode

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

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

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

4. Disable nonstop-bridging if it is enabled:

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

5. Disable Routing Engine redundancy if enabled:

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

6. Save the configuration change on both Routing Engines:

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

> **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@host# exit
```

Installing Software on the Backup Routing Engine

After the device 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 110.

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.

CAUTION: 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 device. 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 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 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 140.

   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 Enigne 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.

> **CAUTION:** 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 device. 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 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 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 140.

   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 Device for the Installation” on page 155):

```
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)
outing 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 44.
   - 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.

SEE ALSO

Understanding Routing Engine Redundancy on Juniper Networks Routers

Repartitioning Routing Engine System Storage to Increase the Swap Partition | 397

Installing Software on EX Series Switches

IN THIS SECTION

EX Series devices are delivered with preinstalled Junos operating system (Junos OS). Before you start this procedure, decide which software package you need and download it. For information on which packages to use for which upgrades, see "Junos OS Installation Package Names" on page 110.

Understanding Software Installation on EX Series Switches

IN THIS SECTION

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

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 following 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.

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 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 for Switches” on page 196.

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 167 to complete the software installation.

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 169 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
```

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

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.

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)*.

CAUTION: 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 167.

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:

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 protocols layer2-control 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
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)
```

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     U-Boot 1.1.6 (Jan 13 2009 - 06:55:22) 2.3.0
                   loader     FreeBSD/PowerPC U-Boot bootstrap loader 2.2
FPC 7              U-Boot     U-Boot 1.1.6 (Jan 13 2009 - 06:55:22) 2.3.0
                   loader     FreeBSD/PowerPC U-Boot bootstrap loader 2.2
Routing Engine 0   U-Boot     U-Boot 1.1.6 (Mar 11 2011 - 04:29:01) 3.5.0
                   loader     FreeBSD/PowerPC U-Boot bootstrap loader 2.4
Routing Engine 1   U-Boot     U-Boot 1.1.6 (Mar 11 2011 - 04:29:01) 2.3.0
                   loader     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 Download page (https://support-www.juniper.net/support/downloads/) 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://support-www.juniper.net/support/downloads/. Select the OS type and the release. Then find and click the download image.

A login screen appears.

3. Log in with you user name and password.

4. 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
```

5. 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-8200-11.3*build*-signed.tgz**.

6. 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....
```

7. Confirm the loader software upgrade:

user@switch> **show system firmware**

```
Part               Type            Tag Current   Available Status
                                       version   version
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.

8. 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.

9. After the line card restart has completed, confirm the loader software version update:

   user@switch> **show chassis firmware**

```
Part               Type            Tag Current   Available Status
                                       version   version
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.

10. 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.

11. Repeat Steps 4 through 8 for all other line cards that require a line card loader version upgrade.

SEE ALSO

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

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 55. 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**.

SEE ALSO

EX4300 Switches Hardware Overview for USB port location.

Switch Fabric and Routing Engine (SRE) Module in an EX6200 Switch for USB port location.

Switch Fabric and Routing Engine (SRE) Module in an EX8208 Switch for USB port location.

Routing Engine (RE) Module in an EX8216 Switch for USB port location.

RELATED DOCUMENTATION

Upgrading Software on an EX6200 or EX8200 Standalone Switch Using Nonstop Software Upgrade (CLI Procedure)

Troubleshooting Software Installation | 481

Installing Software on QFX Series Devices

IN THIS SECTION

QFX Series devices are delivered with the Junos operating system (Junos OS) preinstalled. Before you start this procedure, decide which software package you need and download it. For information on which packages to use for which upgrades, see "Junos OS Installation Package Names" on page 110.

Installing Software Packages on QFX Series Devices

IN THIS SECTION

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 the Software on QFX10002-60C Switches

This section explains how to upgrade the software, which includes both the host OS and the Junos OS. This upgrade requires that you use a VM host package—for example, a **junos-vmhost-install-x.tgz** .

During a software upgrade, the alternate partition of the SSD is upgraded, which will become primary partition after a reboot. If there is a bootfailure on the primary SSD, the switch can boot using the snapshot available on the alternate SSD.

> **NOTE:** The QFX10002-60C switch supports 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 files under /config and /var (except /var/etc) are preserved after the upgrade.

To upgrade the software, you can use the following methods:

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

For example:

```
user@switch> request vmhost software add /var/tmp/junos-vmhost-install-qfx-x86-64-18.1R1.9.tgz
```

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

For example:

```
user@switch> request vmhost software add
ftp://ftpserver/directory/junos-vmhost-install-qfx-x86-64-18.1R1.9.tgz
```

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 QFX5000 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://ftpserver/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 10: 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.

Table 10: Caveats for Downgrading from Junos OS Release 15.1X53-D60 to Previous Software Releases *(continued)*

Junos OS Software Releases	Using the CLI	Using a USB Stick
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 11: 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.

Table 11: Caveats for Downgrading from Junos OS Release 15.1X53-D60 to Previous Software Releases *(continued)*

Junos OS Software Releases	Using the CLI	Using a USB Stick
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.

> **CAUTION:** 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:

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

Upgrading Software by Using Automatic Software Download for Switches

IN THIS SECTION

The automatic software download feature uses the Dynamic Host Configuration Protocol (DHCP) message exchange process to download and install software packages. You configure the automatic software download feature on switches that act as DHCP clients. You must enable automatic software download on a switch before the software upgrade can occur.

You configure a path to a software package file on the DHCP server. The server communicates the path to the software package file through DHCP server messages.

If you enable automatic software download, the DHCP client switch compares the software package name in the DHCP server message with the name of the software package that booted the switch. If the software packages are different, the DHCP client switch downloads and installs the software package specified in the DHCP server message.

Complete the following tasks in order:

Configuring DHCP Services for the Switch

Before you upgrade software by using automatic software download, ensure that you have configured DHCP services for the switch, including configuring a path to a boot server and a boot file.

To configure a path to a boot server and a boot file:

1. Configure the name of the boot server advertised to DHCP clients. The client uses a boot file located on the boot server to complete DHCP setup. This configuration is equivalent to DHCP option 66:

   ```
   [edit system services dhcp]
   user@switch# set boot-server (address | hostname)
   ```

2. Set the boot file advertised to DHCP clients. After the client receives an IP address and the boot file location from the DHCP server, the client uses the boot image stored in the boot file to complete the DHCP setup. This configuration is equivalent to DHCP option 67:

   ```
   [edit system services dhcp]
   user@switch# set boot-file filename
   ```

Enabling Automatic Software Download on a Switch

To enable automatic software download on a switch that acts as a DHCP client:

```
[edit chassis]
user@switch# set auto-image-upgrade
```

After automatic software download is enabled on your DHCP client switch and after DHCP services are enabled on your network, an automatic software download can occur at any time as part of the DHCP message exchange process.

If an automatic software download occurs, you see the following message on the switch:

Auto-image upgrade started
On successful installation system will reboot automatically

The switch reboots automatically to complete the upgrade.

Verifying That Automatic Software Download Is Working Correctly

Purpose

Verify that the automatic software download feature is working correctly.

Action

Use the **show system services dhcp client *interface-name*** command to verify that the automatic software download feature has been used to install a software package.

`user@switch>` **show system services dhcp client ge-0/0/1.0**

```
 Logical Interface Name        ge-0/0/1.0
  Hardware address         00:0a:12:00:12:12
  Client Status            bound
  Vendor Identifier        ether
  Server Address           10.1.1.1
  Address obtained         10.1.1.89
  Lease Obtained at        2009-08-20 18:13:04 PST
  Lease Expires at         2009-08-22 18:13:04 PST

DHCP Options :
  Name: name-server, Value: [ 10.209.194.131, 203.0.113.2, 203.0.113.3 ]

  Name: server-identifier, Value: 10.1.1.1
  Name: router, Value: [ 10.1.1.80 ]
  Name: boot-image,
  Value: jinstall-ex-4200-9.6R1.5-domestic-signed.tgz
  Name: boot-image-location,
  Value: 10.1.1.25:/bootfiles/
```

Meaning

The output from this command shows the name and location of the software package under DHCP options when automatic software download was last used to install a software package. The sample output in DHCP options shows that the last DHCP server message to arrive on the DHCP client had a boot server address of 10.1.1.1 and a boot file named **jinstall-ex-4200-9.6R1.5-domestic-signed.tgz**. If automatic software download was enabled on this client switch during the last DHCP message exchange, these values were used by the switch to upgrade the software.

SEE ALSO

Configuring a DHCP Server on Switches

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 12 on page 199, Table 13 on page 200, and Table 14 on page 200 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 15 on page 201 to see which Uboot software versions are available and the filenames of the Jloader software packages.

Table 12: Junos OS and Jloader Software Compatibility Matrix for the QFX3500 Switch and QFX3500 Node Device

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

Table 12: 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
12.2X50-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 13: 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 14: 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

Table 14: Junos OS and Jloader Software Compatibility Matrix for the QFX3600-I Interconnect Device and QFX3600 Node Device *(continued)*

Junos OS Software Version	1.1.2	1.1.4	1.1.5	1.1.8
12.2X50-D20 and later (QFX3600 switch)	-	-	Supported	Supported and recommended

Table 15: 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
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://support.juniper.net.

   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.

4. Read the End User License Agreement, click the **I agree** radio button, and then click **Proceed**.

5. 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.

6. Log in to the QFX3500 switch and enter the shell. We recommend using a console connection.

7. 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`

8. 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"`

9. 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**

10. 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
    ```

11. 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"

12. 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             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://support.juniper.net.

   The Downloads page appears.

7. Select the product you want software for.

8. Find and click the file you want to download.

   A login screen appears.

9. Enter your username and password, and press **Enter**.

10. Read the End User License Agreement, click the **I agree** radio button, and then click **Proceed**.

11. 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.

12. 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**

13. 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
    ```

14. 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 /pdbata/export/rlogs**

    ```
    02de4930-828b-11e1-a319-00e081c57938  c9898afe-828b-11e1-956c-00e081c57938
    04103b2a-29d5-e011-bf8a-0e6bdf3aa1e6  eeba4aac-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.

15. 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**

16. 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
```

17. 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.el5 <<>> -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.
dcfnode---DCF-ROOT.pkg.test.device.net. 45 IN A 169.254.192.17
dcfnode---DRE-0.pkg.test.device.net. 45 IN A   169.254.3.3
dcfnode-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.

18. 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-rtsdb                                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 a QFabric System

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 "Installing Software Packages on QFX Series Devices" on page 182.

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://ftpserver/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
interfaces et-0/0/2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/2:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/2:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/2:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/2:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/3:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/3:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/3:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/3:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/4 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/4:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/4:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/4:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/4:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/5 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/5:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/5:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/5:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/5:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/6 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/6:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/6:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/6:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/6:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/7 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/7:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/7:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/7:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/7:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/8 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/8:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/8:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/8:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/8:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
```

```
interfaces et-0/0/9 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/9:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/9:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/9:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/9:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/10 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/10:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/10:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/10:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/10:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/11 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/11:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/11:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/11:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/11:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/12 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/12:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/12:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/12:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/12:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/13 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/13:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/13:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/13:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/13:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/14 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/14:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/14:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/14:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/14:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/15 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/15:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/15:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/15:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/15:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/16 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/16:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/16:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/16:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/16:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/17 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/17:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/17:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/17:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
```

```
interfaces xe-0/0/17:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/18 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/18:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/18:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/18:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/18:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/19 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/19:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/19:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/19:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/19:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/20 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/20:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/20:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/20:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/20:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/21 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/21:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/21:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/21:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/21:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/22 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/22:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/22:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/22:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/22:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/0/23 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/23:0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/23:1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/23:2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces xe-0/0/23:3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/1/0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/1/1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/1/2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/1/3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/2/0 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/2/1 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/2/2 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
interfaces et-0/2/3 unit 0 family inet dhcp vendor-id Juniper-qfx5100-24q-2p
forwarding-options storm-control-profiles default all
protocols lldp interface all
protocols lldp-med interface all
protocols igmp-snooping vlan default
vlans default vlan-id 1
```

SEE ALSO

Overview of Junos Automation Enhancements on Devices Running Junos OS with Enhanced Automation

How To Install Third-Party Software on Devices Running Junos OS Evolved

Third-party software is software that is not part of the normal Junos release cadence for a given target chassis. In the case of Junos OS Evolved, third-party software refers to the following types of software delivered to a node or a cluster of nodes running Junos OS Evolved:

- Private software developed by external customers
- Software or tools developed by internal engineers

Third parties package their software as **.tgz** files. The package filename contains the component name and its version as well as the architecture and SDK version. You install the third-party software package on a device running Junos OS Evolved using the **request system software add *filename*** command. This is the same command you use to install different releases of the Junos OS Evolved software on a device. The only difference is that third-party software are **.tgz** files and the Junos OS Evolved software are **.iso** files.

The procedure is the same as installing software on any device running Junos OS. You back up the current system and you place the software on the device, usually in the **/var/tmp** directory of the active Routing Engine

For example, if you have third-party software developed by Acme with the filename **acmeMonitor-1.2.3_Wrl_9.0_x86_64.tgz**, use the following command to install it on a device running Junos OS Evolved:

```
user@host> request system software add /var/tmp/acmeMonitor-1.2.3_Wrl_9.0_x86_64.tgz
```

NOTE: The **reboot** command is not needed to install third-party applications on devices running Junos OS Evolved.

NOTE: For Junos OS Evolved, if you are trying to reinstall an already installed application, use the **force** option. The **force** option will cause the program to remove the existing application before reinstalling it.

The program detects third-party components already installed in the current version that collide with new components in acmeMonitor-1.2.3_Wrl_9.0_x86_64.tgz. Without using the **force** option, a reinstall of a third-party application fails.

Use the **show version** command to see a list of the current components installed that are not part of the released BOM. The list is tagged as "External Software" and gives the name of each third-party component name and version, as well as the SDK version that was used to create it.

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

```
Model: ptx10003-160c
Junos: 20.1-201910240713.0-EVO
Junos Package: junos-evo-install-ptx-fixed-x86-64-20.1-201910240713.0-EVO.iso
Yocto: 2.2.1
Linux Kernel: 4.8.28-WR2.2.1_standard-g21fb4b9
SDK version: 5.6.7
External Software:
   acmeMonitor-1.2.3 [sdk 5.6.7]
   acmeLoadShare-3.4.5 [sdk 5.6.8]
```

You remove third-party software the same way you remove versions of Junos OS Evolved, for example, to remove the Acme software, use this command:

```
user@host> request system software delete acmeMonitor
```

If you want to delete all third-party software use the **request system software delete all-third-party-packages** command.

RELATED DOCUMENTATION

request system software delete | 641

request system software add | 613

show version

Personality Upgrade Process

IN THIS SECTION

Understanding the Personality Upgrade Process for a Device

Personality of a device can be defined as a combination of the purpose of the device and the solution that the device provides. For instance, a switch is a Layer 2 (Data Link Layer) device that is designed to connect two or more networking devices on a network. Most switches (except a few Layer 3 switches) act as bridge devices that receive data packets from a source device, process the data, and forward it to a destination device within the same network. A router, in contrast, connects multiple networks. It is typically a Layer 3 (Network Layer) device because its primary function is to forward packets destined either for its own network or for other networks.

Starting in Junos OS Release 18.2R1, you can upgrade the personality of a device from the installed personality to a new personality without having to upgrade the entire device.

The build image loaded on a device defines the personality of the device. For instance, suppose you purchase a core router such as a PTX10008. The build image loaded on the router indicates its installed personality—that is, PTX10008. You can upgrade its personality and use it as an edge router such as an MX10008, in which case the device personality changes to MX10008. Similarly, you can change the device personality from an MX10008 to a PTX10008. You can also alternate between a switch (for example, QFX10002-60C) and a router (say, PTX10002-60C) by simply upgrading the device personality.

To upgrade the device personality from one device to another, you need certain common hardware components supported by both the devices. In the case of an MX10008 and a PTX10008, the presence of the common Routing and Control Board (RCB)—JNP10K-RE1—and the eight-slot universal chassis—JNP10008—enables you to upgrade from one device to the other seamlessly.

NOTE: When you order a spare JNP10K-RE1 RCB, the image of the MX10008 build is installed on that RCB. The spare JNP10K-RE1 also contains an image of the PTX10008 build at the **/var/tmp** location. You can upgrade an MX10008 router to an PTX10008 by using that image.

You can upgrade the personality of the device to a new personality by:

- Using the USB flash drive
- Using the Junos OS CLI
- Using the PXE boot server

Benefits of Personality Upgrade

- Reuse—The same device (universal chassis) can be used as an edge router or a core router or a switch.
- Time-saving—You can quickly deploy the new device personality in the network.
- Lower capital expenditure and operating costs—You can upgrade the device personality instead of purchasing a new device.
- Network Growth management—Upgrading the personality of your device helps you manage the network growth when growth forecasts are discouraging.
- Lower inventory and storage costs for distributors.

Guidelines and Restrictions

This section describes the guidelines to consider when you upgrade the personality of a device:

- If you attempt to upgrade the personality of the device without using the recommended CLI command, the device can become inaccessible and unstable. For more information, see “Upgrading the Personality of a Device by Using CLI” on page 224.
- There are no in-built restrictions or checks to validate the image that you plan to install on the device.
- Verify that the installed image supports the required command to upgrade to the new personality. If it does not, upgrade to a later version of the image before you upgrade to the new personality.
- When you upgrade the personality of the device, the configuration present in the device is migrated to the new personality. This is similar to a Junos OS upgrade. Therefore, any configuration that is not supported on the new personality must be deleted before you upgrade the personality. If any unsupported configuration is retained in the device after it reboots with the new image, the device returns to the factory-default configuration.
- Retain the minimum configuration required on the device, so the management interface is accessible.

NOTE: Juniper Networks does not support using the **request vmhost software rollback** command to revert to the previously installed personality.

Supported Personality Upgrades

Table 16 on page 223 displays the various combinations of device personality upgrades that are supported by Junos OS.

Table 16: Supported Personality Upgrades on Junos OS

Installed Personality	New Personality	Initial Junos OS Release	Common HW Component
MX10008	PTX10008	18.2	Routing and Control Board (JNP10K-RE1)
PTX10008	MX10008	18.2	Routing and Control Board (JNP10K-RE1)
QFX10002-60C	PTX10002-60C	18.2	
PTX10002-60C	QFX10002-60C	18.2	
MX10016	PTX10016	18.4	Routing and Control Board (JNP10K-RE1)
PTX10016	MX10016	18.4	Routing and Control Board (JNP10K-RE1)

There are multiple ways to change the personality of a device:

- Upgrading the Personality of a Device by Using a USB Flash Drive on page 224
- Upgrading the Personality of a Device by Using CLI on page 224
- Upgrading the Personality of a Device by Using a PXE Boot Server on page 228

Upgrading the Personality of a Device by Using a USB Flash Drive

The build image loaded on the device defines the personality of the device. You can change the personality of the device by upgrading it.

In a USB upgrade, the content of the SSDs are erased and the image is installed from the USB flash drive to both the primary and secondary disks. Based on the image used, the device comes up as a PTX10008 or an MX10008. This is irrespective of the previously installed personality of the device.

> **NOTE:** When you order a spare JNP10K-RE1 RCB, the image of the MX10008 build is installed on that RCB. The spare RCB also contains an image of the PTX10008 build at the **/var/tmp** location. You can upgrade an MX10008 router to an PTX10008 by using that image.

To upgrade the personality of the device by using a USB flash drive:

1. Insert the external USB flash drive. The external flash drive is detected.

2. Reboot the device.

```
user@host# run request vmhost reboot usb
OR
user@host# run request vmhost reboot
```

3. When prompted, unplug the USB flash drive after the system reboots.

> **NOTE:** Juniper Networks does not support using the **request vmhost software rollback** command to revert to the previously installed personality.

Upgrading the Personality of a Device by Using CLI

The build image loaded on the device defines the personality of the device. You can change the personality of the device by upgrading it.

You can upgrade the personality of the device by using CLI configuration on devices running Junos OS.

NOTE: When you order a spare JNP10K-RE1 RCB, the image of the MX10008 build is installed on that RCB. The spare RCB also contains an image of the PTX10008 build at the **/var/tmp** location. You can upgrade an MX10008 router to an PTX10008 by using that image.

How to Upgrade the Personality of a Device on Junos OS

Use the following CLI procedure to upgrade the personality of a device running Junos OS.

- Verify that the installed image supports the required CLI command to upgrade to the new personality. If it does not, upgrade to a later version of the image before you upgrade to the new personality.
- Delete any configuration that is not supported or is not compatible with the new personality before you upgrade the personality. If any unsupported configuration is retained in the device after it reboots with the new image, the device returns to the factory-default configuration.

To upgrade the device to a new personality by using the Junos OS CLI:

1. In operational mode, verify the installed personality of the device. If you have purchased an MX10008 device, the installed personality of the device is displayed as **mx10008**. If you have purchased a PTX10008 device, the installed personality of the device is displayed as **ptx10008**.

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

   ```
   Hostname: host
   Model: mx10008
   ```

   ```
   Hostname: host
   Model: ptx10008
   ```

2. Download the software package or build image from https://www.juniper.net/support/. For information about downloading software packages, see “Downloading Software” on page 140. Save the software package to the **/var/*path*/*package-name*** directory on the router. For example, you can save the software package to the **/var/tmp** directory.

   NOTE: Download the software package specific to the personality you want to upgrade to. The software package for PTX Series routers is different from the software package for MX Series routers.

3. In configuration mode, install the software package by using the **request vmhost software add *path/package-name*** command. Install the software package based on the new personality you want to upgrade to, as follows:

```
user@host# run request vmhost software add /var/tmp/junos-vmhost-install-ptx-x86-64-xyz.tgz
  upgrade-to-model ptx10008 no-validate
```

```
user@host# run request vmhost software add /var/tmp/junos-vmhost-install-mx-x86-64-zyx.tgz
  upgrade-to-model mx10008 no-validate
```

NOTE: If you do not specify the **no-validate** option, the router displays the following error message: **error: Upgrading to a different model is supported only with no-validate option.**

4. Reboot the router so the new package is loaded.

```
user@host# run request vmhost reboot
```

5. Run the **show version** command to verify that the upgrade is successful. If you have upgraded the personality of the device to an MX10008 device, the new personality of the device is displayed as **mx10008**. If you have upgraded the personality of the device to a PTX10008 device, the new personality of the device is displayed as **ptx10008**.

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

```
Hostname: host
Model: ptx10008
```

```
Hostname: host
Model: mx10008
```

NOTE: Juniper Networks does not support using the **request vmhost software rollback** command to revert to the previously installed personality.

To ensure that all four partitions are upgraded to the same personality, follow these steps:

1. Boot from the solid-state drive (SSD) Disk 2 by using the **request vmhost reboot** command.

   ```
   user@host>request vmhost reboot disk2
   ```

2. Upgrade to the new personality by using the **upgrade-to-model** and **no-validate** options. This command upgrades both partitions on the SSD Disk 1.

   ```
   user@host# run request vmhost software add junos-vmhost-install-x.tgz upgrade-to-model X no-validate
     reboot
   ```

   If you are upgrading to PTX10008, include the package for the PTX Series routers and replace **X** with ***ptx10008*** before the **no-validate** option. If you are upgrading to MX10008, include the package for the MX Series routers and replace **X** with ***mx10008*** before the **no-validate** option.

3. After the device boots up from SSD Disk 1, take a snapshot from SSD Disk 1 to Disk 2.

   ```
   user@host> request vmhost snapshot partition
   ```

   This step ensures that both partitions on Disk 2 are upgraded to the new personality.

After you complete Step 1 through Step 3, all four partitions are upgraded to new personality.

Upgrading the Personality of a Device by Using a PXE Boot Server

The build image loaded on the device defines the personality of the device. You can change the personality of the device by upgrading it.

You can upgrade the personality of a device by using the Preboot Execution Environment (PXE) boot server. 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. You can have a separate PXE boot server for each image.

> **NOTE:** When you order a spare JNP10K-RE1 RCB, the image of the MX10008 build is installed on that RCB. The spare RCB also contains an image of the PTX10008 build at the **/var/tmp** location. You can upgrade an MX10008 router to an PTX10008 by using that image.

To upgrade the personality of a device from the installed personality to the new personality by using the PXE boot server method:

- Copy the image you want installed on the device to the PXE boot server.
- Reboot the device to install the image.

> **NOTE:** If you have already copied the image to the PXE boot server, reboot the device to install the image.

To copy the image you want installed to the PXE boot server and install the image:

1. Copy the downloaded installation media to the **/var/tmp** directory in the PXE boot server.

   For example:

   ```
   scp
   /volume/build/junos/18.2/release/zyx/ship/junos-vmhost-install-net-x86-64-xyz.tgz
   user@host:/var/tmp/
   ```

2. Log in to the PXE boot server and verify the installation file.

   For example:

   `user@host>` **ls -lh junos-vmhost-install-net-x86-64-xyz.tgz**

   ```
   -rw-r--r-- 1 root root 1.8G May 24 00:42 junos-vmhost-install-net-x86-64-xyz.tgz
   ```

3. Extract the **junos-vmhost-install-net** TAR file.

For example:

user@host> **tar xvzf junos-vmhost-install-net-x86-64-xyz.tgz –C /var/tmp**

```
attributes
junos-vmhost-install-ptx.tgz
manifest
manifest.certs
manifest.ecerts
manifest.esig
manifest.sig
package.xml
pkg_add_vmhost.sh
vmhost-install-net-x86_64-xyz.tgz
```

4. Remove the previously installed files, if any, from the **/tftpboot** directory.

   user@host> **rm -f /tftpboot**

   user@host> **mkdir /tftpboot**

5. Extract the network installation package.

   For example:

   user@host> **tar xvzf /var/tmp/vmhost-install-net-x86_64-xyz.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-xyz.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.

   For example, this sample output is specific to the PTX Series device installation package.

   user@host>**tar xvzf /var/tmp/junos-vmhost-install-ptx.tgz -C /var/install**

```
./
./vmhost-pkgs-version
./vm/
./vm/note
./vm/grub.cfg.ngre
./vm/vsmartd-1.0-0.x86_64.rpm
./vm/re_fpga-1.0-0.x86_64.rpm
./vm/veccd-1.0-0.x86_64.rpm
./vmhost-version.sh
./vmhost/
./vmhost/vmhost-x86_64-xyz.img.gz
...
...
./junos/junos-mtre-upgrade.sh
./vmhost-core-x86_64-15.1I20151019_1021_builder.tgz
./junos/
./junos/junos-install-x86-64-xyz.img.gz
```

8. Verify that the **/var/install** folder contains the **attributes** file. If the file does not exist in the specified location, copy the attribute file.

   NOTE: The attribute file consists of the personality information of the image. If the attributes file is not present, the device is unable to upgrade to the new personality even when the PXE boot server has the relevant image.

```
user@host> mv /var/tmp/attributes /var/install
```

9. Set permissions for the files in the **/var/install** and **/tftpboot** directories.

```
user@host> chown root:root /tftpboot/*
user@host>chmod a+rwx /tftpboot/*
user@host>chown -R root:root /var/install
user@host>chmod -R a+rwx /var/install
```

10. Exit the PXE boot server.

```
user@host> exit
```

11. After you copy the image to the PXE boot server, to install the image on the device, reboot the device to install the image.

```
user@host> request vmhost reboot network
```

The router boots from the PXE server and installs the image on both the SSDs.

If the device fails to reboot, you can use the USB disk installation option. For more information, see "Upgrading the Personality of a Device by Using a USB Flash Drive" on page 224. However, after using USB disk installation, if the router fails to reboot or is not accessible, follow these steps on the console:

1. Power cycle the chassis or remove the RCB (JNP10K-RE1) and plug it back in.

2. Press the **ESC** button to go to the Boot Manager Menu.

3. Select **Boot Manager**, and then press Enter.

4. Select the **ETH00 (xx:xx:xx:xx:xx:xx)** option. A warning message is displayed. At the prompt, select **y** to install the image on both the primary and secondary disks.

```
WARNING: The installation will erase the contents of your disks.
```

```
Install vmhost and Junos Software on Primary and Secondary disk [y/n]
```

```
y
```

5. In operational mode, verify that the upgrade is successful. If you have upgraded the personality of the device to an MX10008, the new personality of the device is **mx10008**. If you have upgraded the personality of the device to a PTX10008, the new personality of the device is **ptx10008**.

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

```
Hostname: host
Model: ptx10008
```

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

```
Hostname: host
Model: mx10008
```

> **NOTE:** Juniper Networks does not support using the **request vmhost software rollback** command to revert to the previously installed personality.

Release History Table

Release	Description
18.2R1	Starting in Junos OS Release 18.2R1, you can upgrade the personality of a device from the installed personality to a new personality without having to upgrade the entire device.

RELATED DOCUMENTATION

Upgrading the Personality of QFX10002-60C and PTX10002-60C Switches Using the PXE Boot Server

Upgrading the Personality of QFX10002-60C and PTX10002-60C Devices

IN THIS SECTION

The installed image on your devices determines the personality of the device. Juniper Networks offers benefits of changing the personality of your device. You can install the image of QFX10002-60C in

PTX10002-60C device and vice versa. You can install the new personality via Preboot Execution Environment (PXE) boot method boot, USB, CLI, and ZTP.

Upgrading the Personality of QFX10002-60C and PTX10002-60C Devices Using the PXE Boot Server

You can configure your QFX10002-60C device as a PTX10002-60C device or your PTX10002-60C device as a QFX10002-60C device. The image loaded on the device determines the personality of the device. For example, if you have purchased a QFX10002-60C device, the installed personality is a QFX10002-60C device. When you upgrade the QFX10002-60C device to a PTX10002-60C device, the new personality of the device is a PTX10002-60C device. Similarly, if you have purchased a PTX10002-60C device, the installed personality is a PTX10002-60C device and the new personality, if you upgrade, is a QFX10002-60C device.

You can install the new personality on the devices using Preboot Execution Environment (PXE) boot method. 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. You can have separate PXE boot servers for each image.

> **NOTE:** When you upgrade the QFX10002-60C personality to a PTX10002-60C personality, the QFX10002-60C default configuration is deleted, and the PTX10002-60C configuration becomes the default configuration. When you upgrade the PTX10002-60C personality to a QFX10002-60C personality, the PTX10002-60C default configuration is deleted, and the QFX10002-60C configuration becomes the default configuration. Additionally, the software snapshot in the secondary disk is deleted, and the new software snapshot is installed in the secondary disk. For example, if you upgrade the QFX10002-60C personality to a PTX10002-60C personality, the QFX10002-60C snapshot is deleted, and the PTX10002-60C snapshot is installed in the secondary disk.

> **NOTE:** When you order the spare JNP10002-60C-CHAS, it is preloaded with the QFX10002-60C and PTX10002-60C software images in the **/var/tmp** location. If you want to convert a QFX10002-60C device to a PTX10002-60C device, use the PTX10002-60C image. If you want to convert a PTX10002-60C device to QFX10002-60C device, use the QFX10002-60C image.

For example, to upgrade the QFX10002-60C device from the installed personality of QFX10002-60C to the new personality of PTX10002-60C device using the PXE Boot Server Option:

- Copy the image you want installed on the QFX10002-60C device to the PXE Boot Server.
- Reboot the device to install the image.

> **NOTE:** If you have already copied the image to the PXE Boot server, reboot the device to install the image.

To copy the image you want installed to the PXE Boot Server:

1. Copy the downloaded installation media to the **/var/tmp** directory in the PXE boot server.

   For example:

   ```
   scp
   /volume/build/junos/18.2/release/zyx/ship/junos-vmhost-install-ptx-x86-64-xyz.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-ptx-x86-64-xyz.tgz**

   ```
   -rw-r--r-- 1 root root 1.8G May 24 00:42 junos-vmhost-install-net-x86-64-xyz.tgz
   ```

3. Extract the **junos-vmhost-install-net** TAR file.

   `user@host>` **tar xvzf junos-vmhost-install-ptx-x86-64-xyz.tgz –C /var/tmp**

   ```
   attributes
   junos-vmhost-install-ptx.tgz
   manifest
   manifest.certs
   manifest.ecerts
   manifest.esig
   manifest.sig
   package.xml
   pkg_add_vmhost.sh
   vmhost-install-net-x86_64-xyz.tgz
   ```

4. Remove the previously installed files, if any, from the **/tftpboot** directory.

   `user@host>` **rm -f /tftpboot**

   `user@host>`**mkdir /tftpboot**

5. Extract the network installation package.

user@host> **tar xvzf /var/tmp/junos-vmhost-install-ptx-x86-64-xyz.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-xyz.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/junos-vmhost-install-ptx-x86-64.tgz -C /var/install**

```
./
./vmhost-pkgs-version
./vm/
./vm/note
./vm/grub.cfg.ngre
./vm/vsmartd-1.0-0.x86_64.rpm
./vm/re_fpga-1.0-0.x86_64.rpm
./vm/veccd-1.0-0.x86_64.rpm
./vmhost-version.sh
./vmhost/
./vmhost/vmhost-x86_64-xyz.img.gz
...
...
./junos/junos-mtre-upgrade.sh
./vmhost-core-x86_64-15.1I20151019_1021_builder.tgz
./junos/
./junos/junos-install-x86-64-xyz.img.gz
```

8. Verify that the **/var/install** folder contains the **attributes** file. If the file does not exist in the specified location, copy the attribute file.

NOTE: The attribute file consists of the personality information of the image. If the attributes file is not present, the device is unable to upgrade to the new personality even when the PXE boot server has the relevant image.

```
user@host> mv /var/tmp/attributes /var/install
```

9. Set permissions for the files in the **/var/install** and **/tftpboot** directories.

```
user@host> chown root:root /tftpboot/*
user@host>chmod a+rwx /tftpboot/*
user@host>chown -R root:root /var/install
user@host>chmod -R a+rwx /var/install
```

10. Exit the PXE boot server.

```
user@host> exit
```

After you copy the image to the PXE Boot Server, to install the image on the device, reboot the device to install the image. You can use the **request vmhost reboot network** command to install the image. The device boots from the PXE server and installs the image on both the SSDs. However, if the device fails to reboot, you can use the USB disk installation option. If the device fails to reboot or is not accessible, follow these steps:

1. Power cycle the device.

2. Press the **ESC** button to go to the Boot Manager Menu.

3. Select **Boot Manager**, and then press Enter.

4. Select **ETH00 (xx:xx:xx:xx:xx:xx)** option. A warning message is displayed. At the prompt, select **y** to install the image on both the primary and secondary disks.

```
WARNING: The installation will erase the contents of your disks.
```

```
Install vmhost and Junos Software on Primary and Secondary disk [y/n]
```

```
y
```

5. In operational mode, verify that the upgrade is successful.

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

```
Hostname: host
Model: ptx10002-60C
```

NOTE: Juniper Networks does not support using the **request vmhost software rollback** command to revert to the previously installed personality.

Upgrading the Personality of QFX10002-60C and PTX10002-60C Devices Using the USB Option

You can configure your QFX10002-60C device as a PTX10002-60C device or your PTX10002-60C device as a QFX10002-60C device. The image loaded on the device determines the personality of the device. For example, if you have purchased a QFX10002-60C device, the installed personality is a QFX10002-60C device. When you upgrade the QFX10002-60C device to a PTX10002-60C device, the new personality of the device is a PTX10002-60 C device. Similarly, if you have purchased a PTX10002-60C device, the installed personality is a PTX10002-60C device and the new personality, if you upgrade, is a QFX10002-60C device.

In an USB upgrade, the content of the SSDs are erased and the image is installed from the USB to both the primary and secondary disks. Based on the image used, the device comes up as either a QFX10002-60C or a PTX10002-60C device. This is irrespective of the previously installed personality of the JNP10002-60C-CHAS chassis.

> **NOTE:** When you order the spare JNP10002-60C-CHAS, it is preloaded with the QFX10002-60C and PTX10002-60C software images in the **/var/tmp** location. If you want to convert a QFX10002-60C device to a PTX10002-60C device, use the PTX10002-60C image. If you want to convert a PTX10002-60C device to QFX10002-60C device, use the QFX10002-60C image.

For example, to upgrade the QFX10002-60C device from the installed personality of QFX10002-60C to the new personality of PTX10002-60C device using the USB Option:

1. Insert the external USB flash drive. The external flash drive is detected.

2. Reboot the device.

```
user@host# run request vmhost reboot usb
OR
user@host# run request vmhost reboot
```

3. Unplug the USB flash drive after the system reboots, when prompted.

> **NOTE:** Juniper Networks does not support using the **request vmhost software rollback** command to revert to the previously installed personality.

Upgrading the Personality of QFX10002-60C and PTX10002-60C Devices Using the CLI Option

You can configure your QFX10002-60C device as a PTX10002-60C device or your PTX10002-60C device as a QFX10002-60C device. The image loaded on the device determines the personality of the device. For example, if you have purchased a QFX10002-60C device, the installed personality is a QFX10002-60C device. When you upgrade the QFX10002-60C device to a PTX10002-60C device, the new personality of the device is a PTX10002-60C device. Similarly, if you have purchased a PTX10002-60C device, the installed personality is PTX10002-60C and the new personality, if you upgrade, is a QFX10002-60C device.

NOTE: When you upgrade the QFX10002-60C personality to a PTX10002-60C personality, the QFX10002-60C default configuration is deleted, and the PTX10002-60C configuration becomes the default configuration. When you upgrade the PTX10002-60C personality to a QFX10002-60C personality, the PTX10002-60C default configuration is deleted, and the QFX10002-60C configuration becomes the default configuration. Additionally, the software snapshot in the secondary disk is deleted, and the new software snapshot is installed in the secondary disk. For example, if you upgrade the QFX10002-60C personality to a PTX10002-60C personality, the QFX10002-60C snapshot is deleted, and the PTX10002-60C snapshot is installed in the secondary disk.

NOTE: When you order the spare JNP10002-60C-CHAS, it is preloaded with the QFX10002-60C and PTX10002-60C software images in the **/var/tmp** location. If you want to convert a QFX10002-60C device to a PTX10002-60C device, use the PTX10002-60C image. If you want to convert a PTX10002-60C device to QFX10002-60C device, use the QFX10002-60C image.

- Verify if the installed image supports the required command to upgrade to the new personality. If it does not, upgrade to a later version of the image before you upgrade to the new personality.
- Delete any configuration that is not supported or not compatible with the new personality before you upgrade the personality. If any unsupported configuration is retained in the device after it reboots with the new image, the device returns to the factory-default configuration.

For example, to upgrade the QFX10002-60C device from the installed personality of QFX10002-60C to the new personality of PTX10002-60C device using the CLI Option:

1. In operational mode, verify the installed personality of the device

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

```
Hostname: host
Model: QFX10002-60C
```

2. Download the software package from https://www.juniper.net/support/. For information about downloading software packages, see "Downloading Software" on page 140. Save the software package to the **/var/*path/package-name*** directory on the device. For example, you can save the software package to the **/var/tmp** directory.

   > **NOTE:** Download the software package specific to the personality you want to upgrade to. The software package for QFX Series devices is different from the software package for the PTX Series devices.

3. In configuration mode, install the software package by using the **request vmhost software add *path/package-name*** command. For example, to install the **junos-vmhost-install-ptx-x86-64-zyx.tgz** package:

```
user@host# run request vmhost software add /var/tmp/junos-vmhost-install-ptx-x86-64-zyx.tgz
  upgrade-to-model ptx10002-60C no-validate
```

   > **NOTE:** If you do not specify the **no-validate** option, the device displays the following error message **error: Upgrading to a different model is supported only with no-validate option** .

4. Run the **show version** command to verify that the upgrade is successful.

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

```
Hostname: host
Model: ptx10002-60C
```

> **NOTE:** Juniper Networks does not support using the **request vmhost software rollback** command to revert to the previously installed personality.

To ensure that all 4 partitions are upgraded to the same personality, follow these steps:

1. Boot from solid-state drive (SSD) Disk 2 using the **request vmhost reboot** command.

   ```
   user@host>request vmhost reboot disk2
   ```

2. Upgrade to the new personality using the **upgrade-to-model** and **no-validate** options. This command upgrades both partitions on SSD Disk 1.

   ```
   user@host# run request vmhost software add junos-vmhost-install-x.tgz upgrade-to-model X no-validate
     reboot
   ```

   If you are upgrading to PTX10002-60C, include the package for the PTX10002-60C and replace **X** with **PTX10002-60C** before the **no-validate** option. If you are upgrading to QFX10002-60C, include the package for the QFX10002-60C and replace **X** with **QFX10002-60C** before the **no-validate** option.

3. After booting up from SSD1, take a snapshot from SSD1 to SSD2.

   ```
   user@host>request vmhost snapshot partition
   ```

   This ensures that both partitions on SSD2 are upgraded to new personality.

Upgrading the Personality of QFX10002-60C and PTX10002-60C Devices Using Zero Touch Provisioning (ZTP)

You can configure your QFX10002-60C device as a PTX10002-60C device or your PTX10002-60C device as a QFX10002-60C device. The image loaded on the device determines the personality of the device. For example, if you have purchased a QFX10002-60C device, the installed personality is a QFX10002-60C device. When you upgrade the QFX10002-60C device to a PTX10002-60C device, the new personality of the device is a PTX10002-60C device. Similarly, if you have purchased a PTX10002-60C device, the installed personality is PTX10002-60C and the new personality, if you upgrade, is a QFX10002-60C device.

NOTE: When you upgrade the QFX10002-60C personality to a PTX10002-60C personality, the QFX10002-60C default configuration is deleted, and the PTX10002-60C configuration becomes the default configuration. When you upgrade the PTX10002-60C personality to a QFX10002-60C personality, the PTX10002-60C default configuration is deleted, and the QFX10002-60C configuration becomes the default configuration. If you have provided your own Junos OS configuration, that configuration becomes the default configuration. Additionally, the software snapshot in the secondary disk is deleted, and the new software snapshot is installed in the secondary disk. For example, if you upgrade the QFX10002-60C personality to a PTX10002-60C personality, the QFX10002-60C snapshot is deleted, and the PTX10002-60C snapshot is installed in the secondary disk.

NOTE: When you order the spare JNP10002-60C-CHAS, it is preloaded with the QFX10002-60C and PTX10002-60C software images in the **/var/tmp** location. If you want to convert a QFX10002-60C device to a PTX10002-60C device, use the PTX10002-60C image. If you want to convert a PTX10002-60C device to QFX10002-60C device, use the QFX10002-60C image.

- Verify if the installed image supports the required command to upgrade to the new personality. If it does not, upgrade to a later version of the image before you upgrade to the new personality.
- Delete any configuration that is not supported or not compatible with the new personality before you upgrade the personality. If any unsupported configuration is retained in the device after it reboots with the new image, the device returns to the factory-default configuration.

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.

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

- 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:** 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.

For example, to upgrade the QFX10002-60C device from the installed personality of QFX10002-60C to the new personality of PTX10002-60C device using ZTP:

1. In operational mode, verify the installed personality of the device

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

   ```
   Hostname: host
   Model: QFX10002-60C
   ```

2. Boot the device.

3. Make sure the device has the default factory configuration installed.

Issue the **request vmhost zeroize** command on the device that you want to provision.

4. Download the software package specific to the personality you want to upgrade from https://www.juniper.net/support/.

   The software package for QFX Series devices is different from the software package for the PTX Series devices.

5. Save the software package and the configuration file to the FTP, HTTP, or TFTP server from which the device will download these files.

6. 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.

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

   Here is an example of an ISC DHCP 4.2 server dhcpd.conf file:

```
option space NEW_OP; option;
  option NEW_OP.image-file-name code 0 = text;
  option NEW_OP.config-file-name code 1 = text;
  option NEW_OP.image-file-type code 2 = text;
  option NEW_OP.transfer-mode code 3 = text;
  option NEW_OP.alt-image-file-name code 4= text;
  option NEW_OP.jloader-file code 5 = text;
  option NEW_OP-encapsulation code 43 = encapsulate NEW_OP;
```

NOTE: Starting in Junos OS Release 18.2R1, a new DHCP option is introduced to set the timeout value for the file downloads over FTP. If the **transfer-mode** is set as FTP, the default value for the timeout is automatically set as 120 minutes, that is, in case the FTP session gets interrupted due to loss of connectivity in the middle of a file transfer, it will timeout after 120 minutes and ZTP will attempt to retry the file fetching process. This value can be overridden using the DHCP option as follows:

```
option NEW_OP.ftp-timeout code 7 = text;
option NEW_OP.ftp-timeout “val”;
```

where **“val”** is the user configurable timeout value in seconds and must be provided within quotes (like, "val”).

8. Configure the following DHCP option 43 suboptions:

NOTE: DHCP option 43 suboptions 05 through 255 are reserved.

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

  NOTE: When the DHCP server cannot use suboption 00, configure the software image filename 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";
  ```

  The following list provides the types of scripts and their associated interpreter paths:

  - Shell script interpreter path: **#!/bin/sh**
  - SLAX script interpreter path: **#!/usr/libexec/ui/cscript**
  - Python script interpreter path: **#!/usr/bin/python**

Unsigned Python scripts are only supported on limited platforms, such as the QFX5100 device. If you try to execute unsigned Python scripts on devices that do not provide support, error messages will be issued.

> **NOTE:** If the file does not contain special characters (**#!**) , ZTP determines that the file is a configuration file and loads the configuration file.

- 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 ZTP 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";
```

- Suboption 05: The name of the JLoader image file to install.

```
NEW_OP.jloader-file "jloader-qfx-5-14.1X53-D26-signed.tgz";
```

9. (Mandatory) Configure either option 150 or option 66.

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;
```

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

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

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

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

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

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

12. (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";
  option NEW_OP.jloader-file "jloader-qfx-5-14.1X53-D26-signed.tgz";
}
```

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;
    }
}
```

13. Monitor the ZTP process by looking at the following log files.

    NOTE: When SLAX (live operating system based on Linux) scripts are issued, the **op-script.log** and **event-script.log** files are produced.

    - /var/log/dhcp_logfile
    - /var/log/event-script.log
    - /var/log/image_load_log
    - /var/log/messages
    - /var/log/op-script.log
    - /var/log/script_output

    You can also monitor the ZTP process by looking at error messages and issuing operational commands. See "Monitoring Zero Touch Provisioning" on page 426 for more information.

14. Run the **show version** command to verify that the upgrade is successful.

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

```
Hostname: host
Model: ptx10002-60C
```

Upgrade the NFX250 Software to NFX250 NextGen Software

IN THIS SECTION

NFX250 NextGen Software Upgrade Overview

Starting in Junos OS Release 19.1R1, the NFX250 devices support the NFX250 NextGen software architecture. This is a reoptimized architecture that enables you to use JCP as the single point of management to manage all the NFX250 components. For more information about the NFX250 NextGen architecture, see *NFX250 NextGen Overview*.

> **NOTE:** For documentation purposes, NFX250 devices that use the reoptimized architecture are referred to as NFX250 NextGen devices.

You can upgrade the software using a USB or through a CLI. This topic provides information about prerequisites and the procedure to upgrade through a CLI from NFX250 software architecture to NFX250 NextGen software architecture.

> **NOTE:** The upgrade procedure using a USB remains the same for all NFX Series devices.

Prerequisites

To upgrade an NFX250 device, you must meet the following prerequisites:

Device-specific prerequisites

- An NFX250 device with BIOS => CBDE_SFP_00.21_01.01

  To verify the BIOS version:

  ```
  root@jdm> request execute-command "jhost dmidecode -t bios"
  ```

  For the BIOS information, see the **BIOS Information** section in the command output message.

  If the BIOS version is not CBDE_SFP_00.21_01.01, you can upgrade the BIOS:

  1. Download the BIOS from Downloads page.

  2. Copy and save the BIOS image to the **/var/third-party/firmware** directory.

  3. From the JDM CLI, access the hypervisor:

     ```
     root@jdm> ssh hypervisor
     ```

  4. Reboot the device to load new BIOS.

     a. Exit from hypervisor shell:

     ```
     root@local-node:~# exit
     logout
     Connection to hypervisor closed.
     {master:0}
     root@JDM>
     ```

     b. Reboot the device from JDM CLI.

     ```
     {master:0}
     root@porter-p2a-sys1> request system reboot
     Reboot the system ? [yes,no] (no) yes
     ```

  5. Upgrade the BIOS:

     ```
     root@host:~# rpm -ivh /var/third-party/firmware/BIOS RPM package name
     ```

     The system generates the following output:

     ```
     Preparing...            ########################################### [100%]
     1:nfx-2-secure-bios     ########################################### [100%]
     A reboot is required to install the secure BIOS
     Please reboot the system to complete the install
     ```

- An NFX250 NextGen configuration file with minimal or necessary configurations is required for remote management access to the device after migrating to NFX250 NextGen. This file is an input data for the **request system software add clean-install *package-name*** command.

Release-specific prerequisites

The NFX250 software must be compatible with the following releases:

- NFX250 software running Junos OS Release 18.4R2 or later to accept the configuration by using the command:

```
user@host> request system software add clean-install package-name
```

> **CAUTION:** The **clean-install** command removes all contents on the hard disk. To avoid data loss, copy all important files, configuration files (JDM, JCP, vSRX, and third-party VNFs), log files, and VNF disk or image file, and save them in a secure location before you upgrade the device.

- Releases prior to 18.4R2 must be upgraded to 18.4R2 or later.

> **CAUTION:** The NFX250 device will crash if you upgrade the NFX250 software image running Junos OS Release prior to 18.4R2 to a release that supports NFX250 NextGen software image.
>
> The NFX250 NextGen configuration must be compatible with the NFX250 NextGen software version. The configuration command syntax is not validated.

> **NOTE:** The NFX250 software architecture and NFX250 NextGen software architecture are different and the configurations are different for both the software.

Upgrade to NFX250 NextGen Software Architecture

Before you upgrade the NFX device:

- Create backup of the configuration files (JDM, JCP, vSRX, and third-party VNFs), log files, VNF disk or image file, and other important files stored on the device.
- Check the prerequisites.

To upgrade the NFX250 software architecture to NFX250 NextGen software architecture:

1. Copy the configuration files that are required for in-band and out-of-band management and save it in the **/var/third-party** folder. The configuration file should be of the same format as the file format obtained by running the **show configuration** CLI command.

2. Copy the NFX250 NextGen software image and save it in the **/var/third-party/images** folder.

3. Initiate the software upgrade by using the following command:

   ```
   root@jdm> request system software add clean-install reboot
   /var/third-party/images/jinstall-image.tgz upgrade-with-config
   /var/third-party/config-file
   ```

   The device is formatted and the NFX250 NextGen software image is installed. The device loads the configurations and boots up the NFX250 Nextgen software image. You can access the device remotely through the in-band and out-of-band management.

4. The device is now ready for additional configurations and third-party VNF onboarding.

Installing Software on SRX Series Devices

IN THIS SECTION

SRX Series devices are delivered with preinstalled Junos operating system (Junos OS). Before you start this procedure, decide which software package you need and download it. For information on which packages to use for which upgrades, see "Junos OS Installation Package Names" on page 110.

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

CAUTION: 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.

Example: Installing Junos OS Upgrade Packages on SRX Series Devices

IN THIS SECTION

This example shows how to install Junos OS upgrades on SRX Series devices.

Requirements

Before you begin:

- Verify the available space on the internal media.
- Download the software package. See Downloads to download the software package for your products.
- 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. To copy the software package to the **/var/tmp** directory, use the following command from the operational mode:

  ```
  user@host> file copy /var/tmp/install/image-name/var/tmp/
  ```

  Example:

  ```
  user@host> file copy /var/tmp/install/junos-srxsme-10.0R2-domestic.tgz /var/tmp/
  ```

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 (for example: 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.
- **reboot** option to reboots the device after installation is completed.

Configuration

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 option, 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. In this example, package name is

   ```
   user@host> request system software add /var/tmp/junos-srxsme-10.0R2-domestic.tgz no-copy
   ```

   NOTE: We recommend that you configure **no-validate** option only when expressly specified by the Juniper Networks Technical Assistance Center (JTAC).

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

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the Junos OS Upgrade Installation

Purpose

Verify that the Junos OS upgrade was installed.

Action

From operational mode, enter the **show version** command.

Sample Output

```
user@host> show version
Hostname: srx340-a
Model: srx345
Junos: 18.2R1-S3.2
JUNOS Software Release [18.2R1-S3.2]
```

Meaning

The **show version** command displays the hostname, model number, and the release information loaded on the device.

Example: Installing Junos OS on SRX Series Devices Using the Partition Option

IN THIS SECTION

This example shows how to install Junos OS Release 10.0 or later with the **partition** option.

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.

CAUTION: 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, SRX340, and SRX345 devices (internal NAND flash) and *not* supported on the alternate media (USB storage key). Partition install is supported on the default media on SRX380 Series devices (internal SSD) and not on 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 265.

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

IN THIS SECTION

Confirm that the configuration is working properly.

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.

SEE ALSO

Configuring Root Partitions on SRX Series Devices | 377

Reverting the Junos OS Software Image Back to the Previous Version

IN THIS SECTION

This example shows how to downgrade Junos OS on the SRX Series devices.

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

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the Junos OS Downgrade Installation

Purpose

Verify that the Junos OS downgrade was installed.

Action

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

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, SRX380, 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.

NOTE: Installing the Junos OS image using a USB flash drive is supported on SRX100, SRX110, SRX210, SRX220, and SRX240 devices.

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: 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.

4. Enter the **show system firmware** command to check whether the upgrade is successful or not.

```
root> show system firmware
Part              Type              Tag Current   Available Status
                                        version   version
FPC 1
PIC 0             MLTE_FW           1   17.2.91   0         OK
Routing Engine 0  RE BIOS           0   3.8       3.6       OK
```

```
Routing Engine 0 RE BIOS Backup    1   3.6       3.6       OK
```

```

```

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

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

```
scanning bus 0 for devices... 1 USB Device(s) found
       scanning usb for storage devices... 1 Storage Device(s) found

FreeBSD/MIPS U-Boot bootstrap loader, Revision 2.10
```

To verify the (bios) firmware 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
```

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 17 on page 271.

   Table 17: Environment Variables Settings

Environment Variables	Description
gatewayip	IP address of the gateway device
ipaddr	IP address of the SRX Series device

Table 17: Environment Variables Settings *(continued)*

Environment Variables	Description
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 198.51.100.15
=> setenv netmask 255.255.255.0
=> setenv gatewayip 198.51.100.1
=> setenv serverip 203.0.113.2
=> 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://203.0.113.2/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. Remove the USB flash drive.

> **NOTE:** On SRX300, SRX320, SRX340, SRX345, SRX380, 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.

Restarting and Halting SRX Series Devices

IN THIS SECTION

This topic includes the following sections:

Rebooting SRX Series Devices

IN THIS SECTION

This example shows how to reboot a SRX Series device.

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

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the Device Reboot

Purpose

Verify that the device rebooted.

Action

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

Halting SRX Series Devices

IN THIS SECTION

This example shows how to halt a device.

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

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the Device Halt

Purpose

Verify that the device halted.

Action

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

Bringing Chassis Components Online and Offline on SRX Series Devices

You can use the **request** commands to bring chassis components (except Power Entry Modules and fans) online and offline.

To bring chassis components online and offline, enter these **request chassis** commands:

```
user@host> request chassis <fru> slot <slot#> pic <pic#> online
```

```
user@host> request chassis <fru> slot <slot#> pic <pic#> online
```

Where **<*fru*>** in the request chassis command can be any of the following (for SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550M devices):

- **fpc**—Changes the Flexible PIC Concentrator (FPC) status.

Where **<*fru*>** in the request chassis command can be any of the following (for SRX5800, SRX5600, and SRX5400 devices):

- **cb**—Changes the control board status.
- **fabric**—Changes the fabric status.
- **fpc**—Changes the Flexible PIC Concentrator (FPC) status.
- **fpm**—Changes the craft interface status.
- **pic**—Changes the physical interface card status.
- **routing-engine**—Changes the routing engine status.

> **NOTE:** The **request chassis** command is not supported for bringing SPCs online and offline.

Example:

To bring specific pic and the corresponding fpc slot online, from operational mode enter the following **request chassis** command:

```
user@host> request chassis pic pic-slot 1 fpc-slot 1 online
```

Restarting the Chassis on SRX Series Devices

You can restart the chassis using the **restart chassis-control** command with the following options:

- To restart the process gracefully:

  user@host> restart chassis-control gracefully

- To restart the process immediately:

  user@host> restart chassis-control immediately

- To restart the process softly:

  user@host> restart chassis-control soft

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

Backing Up an Installation Using Snapshots | 48

Upgrading and Downgrading to Junos with Upgraded FreeBSD

IN THIS SECTION

You can upgrade or downgrade Junos OS with upgraded FreeBSD. You can upgrade Junos OS with upgraded FreeBSD from Junos OS based on FreeBSD 6.1 and upgrade between different releases of Junos OS with upgraded FreeBSD.

Upgrading Junos OS with Upgraded FreeBSD

IN THIS SECTION

NOTE: If you are upgrading or installing Junos OS on a VM host, see "Installing, Upgrading, Backing Up, and Recovery of VM Host" on page 326.

Starting in Junos OS Release 15.1, certain hardware platforms run an upgraded FreeBSD kernel (FreeBSD 10.x or later) instead of FreeBSD 6.1. The information in this section is about upgrading from Junos OS without upgraded FreeBSD (that is, based on FreeBSD 6.1) to Junos OS with upgraded FreeBSD. It does not address upgrading using ISSU. There are certain limitations to using ISSU when upgrading to Junos OS with upgraded FreeBSD. For more information on using ISSU, see *Example: Performing a Unified ISSU*.

When you are upgrading to a different release of Junos OS, you usually use the **request system software add validate** command. The **validate** option checks the candidate software against the current configuration of the device to ensure they are compatible. (Validate is the default behavior when the software package being added is a different release.) However, there are circumstances under which you cannot validate the running configuration in this way. One such circumstance is when you are upgrading to Junos OS with upgraded FreeBSD from Junos OS based on FreeBSD 6.1.

If you are upgrading between releases that cannot use direct validation, 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.

The following sections contain two procedures and one matrix. The procedures cover (1) upgrading to Junos OS with upgraded FreeBSD from Junos OS based on FreeBSD 6.1 and (2) upgrading between different releases of Junos OS with upgraded FreeBSD. To determine whether you are upgrading between releases that can use direct validation or not, see "Determine Which Package or Packages to Install" on page 283.

NOTE: Before installing software on a device that has one or more custom YANG data models added to it, back up and remove the configuration data corresponding to the custom YANG data models from the active configuration. For more information see “Managing YANG Packages and Configurations During a Software Upgrade or Downgrade” on page 150.

Determine Which Package or Packages to Install

To determine which software package to install to upgrade to Junos OS with upgraded FreeBSD, you will need to consult the Feature Explorer and Table 18 on page 284. In using Table 18 on page 284, be aware of the following:

- You can skip no more than two releases when upgrading (or downgrading). That means you can upgrade only to one of the three releases subsequent to your current release. If you want to upgrade across more releases than this, you need to perform multiple upgrades.
- Notice that Table 18 on page 284 separates its information between security devices and routing or switching devices. This is because security devices have been released on a different release sequence than routing and switching devices, and this in turn determines what constitutes skipping no more than two releases. 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 example, 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.

We recommend you upgrade to a 64-bit image of Junos OS with upgraded FreeBSD. In Junos OS releases earlier than 15.1, the partition swap pages are 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.

To determine which installation package and procedure you require:

1. See the **Junos kernel upgrade to FreeBSD 10+** entry in Feature Explorer.

   Click the link or go to **https://pathfinder.juniper.net/feature-explorer/**, type **freebsd**, and select **Junos kernel upgrade to FreeBSD 10+**.

   You will see a listing of platforms that run Junos OS with upgraded FreeBSD and the software release it was introduced in. Different platforms first support Junos OS with upgraded FreeBSD in different releases. Use this listing to find which release you need to install for your device to upgrade to Junos OS with FreeBSD.

2. Consult Table 18 on page 284 to determine the upgrade path to follow.

- Determine which release your device is currently running.

  Look first at the release sequence and then at the second column and find the release running on your device.

- Determine which release you need to install.

  The third column will give you the earliest release you need to install for your platform type to be running Junos OS with upgraded FreeBSD.

Table 18: Upgrade Path to Junos OS with the Upgraded FreeBSD

Release Sequence	Current Router's Junos OS Release	Earliest 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 or 16.1.
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 Junos OS based on FreeBSD 6.1 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.

   For a table listing the package prefixes, see "Junos OS Installation Package Names" on page 110. For more on the names of package name , see "Changes in Package Names for Junos OS with Upgraded FreeBSD" on page 36.

4. Continue installing a software package on a device by using one of the following procedures:

   - Installing the Software Package on a Router with a Single Routing Engine on page 153
   - Installing the Software Package on a Device with Redundant Routing Engines on page 155

Install Junos OS with Upgraded FreeBSD Over 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:** In **/var/db/config**, up to 10 rollback configurations will be saved, depending on the configuration file size.

> **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.

Install 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.

SEE ALSO

Establishing an SSH Connection for a NETCONF Session

Downgrading from Junos OS with Upgraded FreeBSD

IN THIS SECTION

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. To find which platforms support Junos OS with upgraded FreeBSD, see Feature Explorer, enter **freebsd**, and select **Junos kernel upgrade to FreeBSD 10+**.

This topic discusses the different procedures for downgrading from a release of Junos OS with upgraded FreeBSD. One procedure describes how to downgrade to legacy Junos OS. The other procedures describe how to downgrade to an earlier release of Junos OS with upgraded FreeBSD.

The main difference between the procedures is whether to use the **validate** or **no-validate** option with the **request system software add** command. If you downgrade between two versions of legacy Junos OS, **validate** works. Similarly, if you downgrade from Junos OS with upgraded FreeBSD Release 18.1 or later to Release 17.4 or later, **validate** works. However, there is one set of circumstances in which the **no-validate** option must be used when downgrading between Junos OS with upgraded FreeBSD releases, and that is when you downgrade from a Junos OS with upgraded FreeBSD Release 17.4 or later to a release earlier than 17.4, that is, Junos OS releases 15.1 through 17.3.

Select and perform the procedure that matches your set of circumstances.

Downgrading from Junos OS with Upgraded FreeBSD to Legacy Junos OS

If you have previously upgraded to Junos OS with upgraded FreeBSD, you can downgrade to an earlier version of Junos OS (that is, legacy Junos OS) as long as the downgrade conforms to the Junos OS policy of skipping at most two earlier releases.

This example uses the package **/var/tmp/jinstall-13.3R2.7-domestic-signed.tgz** to install legacy Junos OS on the master Routing Engine (**re0**).

To downgrade from Junos OS with upgraded FreeBSD to legacy 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 legacy Junos OS from Junos OS with upgraded FreeBSD.

   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
```

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

Downgrading from Junos OS with Upgraded FreeBSD Release 17.4 or Later to Release 15.1 Through 17.3

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.

To downgrade from Junos OS with upgraded FreeBSD Release 17.4 or later to a Release 15.1 through 17.3:

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

   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
```

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

Downgrading from Junos OS with Upgraded FreeBSD Release 17.3 or Earlier to Release 15.1 Through 17.2

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.

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:

   Use the **validate** and **reboot** options with the **request system software add** command. The command uses the **validate** option by default. 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
```

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

Downgrading from Junos OS with Upgraded FreeBSD Release 18.1 or Later to Release 17.4 or Later

This procedure is applicable when downgrading from Junos OS with Upgraded FreeBSD Releases 18.1 or later to a Junos OS with Upgraded FreeBSD Release 17.4 or later.

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.

To downgrade from Junos OS with upgraded FreeBSD Release 18.1 or later to Junos OS with Upgraded FreeBSD Release 17.4 or later:

1. Enter the **request system software add *package-name* validate reboot** command from the operational mode in the CLI:

   Use the **validate** and **reboot** options with the **request system software add** command. The command uses the **validate** option by default. 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
   ```

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

RELATED DOCUMENTATION

request system snapshot (Junos OS with Upgraded FreeBSD) | 603
request system reboot (Junos OS with Upgraded FreeBSD) | 580
Junos OS with Upgraded FreeBSD | 34

Installing Software on ACX Series Routers

IN THIS SECTION

ACX Series routers are delivered with preinstalled Junos operating system (Junos OS). Before you start this procedure, decide which software package you need and download it. For information on which packages to use for which upgrades, see "Junos OS Installation Package Names" on page 110.

Installing Junos OS Using a USB Storage Device on ACX Series Routers

To install the Junos OS image on ACX Series routers using a USB storage device, you must have access to the USB port physically and you must also have console access. Perform the following steps to install the Junos OS image:

1. Insert the USB storage device that has a valid installation image into the USB port.

2. Reboot the router by either pressing the power button on the chassis or switching off and turning on the power button behind the Routing Engine, or by entering the **request system reboot** command from the CLI. The system LED starts blinking in green.

   On the console, a message is displayed stating that your flash memory device (NAND Flash device) will be formatted and you will lose all the data. You are prompted to confirm the formatting of the flash memory device.

3. Press **y** to confirm and proceed with the formatting process. The flash memory device is formatted and the image is installed on both the partitions.

   After the installation is completed, a message is displayed on the console prompting you to eject the USB storage device and to press **Enter** to reboot the device.

4. After you remove the USB port and press **Enter**, the reboot begins. After the router is rebooted, the new Junos OS version is loaded and functional. The LED glows steadily in green.

> **NOTE:** If an installation error occurs, the LEDs turn red. You must have console access to the router to troubleshoot an installation error.

Installing Junos OS Upgrades from a Remote Server on ACX Series Routers

You can use the CLI to install Junos OS packages that are downloaded with FTP or HTTP from the specified location on internal media, such as the NAND Flash device.

Before you begin:

- Verify the available space on the NAND Flash device.
- Download the Junos OS package.

To install Junos OS upgrades from a remote server, enter the following command from operational mode:

```
user@host>request system software add junos-juniper-12.2R1.9-domestic.tgz no-copy no-validate reboot
```

The new Junos OS image is installed on the router and the device is rebooted.

> **NOTE:** On ACX5048 and ACX5096 routers, use the **force-host** option to force installing the latest version of the Host OS.
>
> ```
> user@host> request system software jinstall-acx5k-15.1X54-D20.6-domestic-signed.tgz force-host
>   add validate reboot
> ```

RELATED DOCUMENTATION

Configuring Root Partitions on ACX Series Routers | 385

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 "Installing Software Packages on QFX Series Devices" on page 182.

Upgrading involves these tasks:

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

You download the software package from the Juniper Networks Downloads page at https://support.juniper.net.

> **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://support.juniper.net.

2. Either click **View all products>** and select the product you are downloading software for, or type the product name.

3. Find the package you want and click the item in the Downloads column.

   A login screen appears.

4. Enter your name and password and press Enter.

5. Read the End User License Agreement, click the **I agree** radio button, and then click **Proceed**.

6. Save the Junos OS software image file to your computer.

   The Junos OS software image file name is presented in the *prefix-release-edition-signed.extension* format. For example, the image name for Junos OS Release 15.1X53-D10 on QFX10000 series switch is **jinstall-qfx-10-f-15.1X53-D10.7-domestic-signed**.

   See "Junos OS Installation Package Names" on page 110 for additional information on image file naming.

7. 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 DCHP 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

ONIE installation from a Junos OS image stored on USB media is not currently supported.

Use another procedure from this document to install ONIE.

Verifying Software Installation

Purpose

Verify that the software was installed successfully on the switch.

Action

To verify that the software was properly installed, issue the **show version** command.

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

Troubleshooting Boot Problems

Problem

Description: Junos OS does not boot.

Solution

If Junos OS does not boot, and the console displays the Yocto GNU Linux shell instead, it could mean that you have booted in the Juniper Linux Debug mode. If you see an error message that says, **"[Error] Does not seem to be an QFX10002."** could mean that the EEPROM does not contain vendor-specific information. To verify the vendor-specific information, perform an ONIE: Rescue installation, and then verify the contents of the **/var/run/*.dat** file.

1. Select **ONIE: Rescue** from the GNU GRUB menu.

2. Issue the **onie-syseeprom** at the ONIE prompt.

   For example:

   ```
   ONIE:/ # onie-syseeprom
   ```

   ```
   TlvInfo Header:
      Id String:    TlvInfo
      Version:      1
      Total Length: 315
   TLV Name             Code Len Value
   -------------------- ---- --- -----
   Base MAC Address     0x24   6 54:2A:A2:FB:DC:00
   MAC Addresses        0x2A   2 256
   Product Name         0x21  23 QFX10000-ÿÿÿÿÿÿÿ
   Serial Number        0x23  12 116G1EC00032
   Part Number          0x22  16 1AES48S6Q.A2Gÿÿÿ
   Device Version       0x26   1 1
   Manufacture Date     0x25  19 01/13/2015 21:40:30
   Vendor Name          0x2D  20 JUNIPER NETWORKS INC
   Manufacturer         0x2B  14 JUNIPER NETWORKS INC
   Vendor Extension     0xFD  48  0x00 0x00 0x7C 0x82 0x01 0x00 0x41 0x32 0xFF 0xFF
    0xFF 0xFF 0xFF 0xFF 0x0F
   Vendor Extension     0xFD  62  0x00 0x00 0x0A 0x4C 0x51 0x06 0x52 0x45 0x56 0x20
    0x30 0x31 0x52 0x0C 0x3F
   ```

```
Platform Name        0x28  37 x86_64-alpha_networks_snx60a0_486f-r0
Loader Version       0x29  23 master-201412161452.0.1
CRC-32               0xFE   4 0xB88C8885
Checksum is valid.
```

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 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-BOOT  hidden, bios_grub
 2      3146kB  137MB   134MB   ext4         ONIE-BOOT  hidden
```

RELATED DOCUMENTATION

Installing Software Packages on QFX Series Devices | 182

Upgrading Software by Using Automatic Software Download for Switches | 196

Configuring a DHCP Server on Switches

Overview of Upgrading to 64-bit Junos OS

IN THIS SECTION

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.

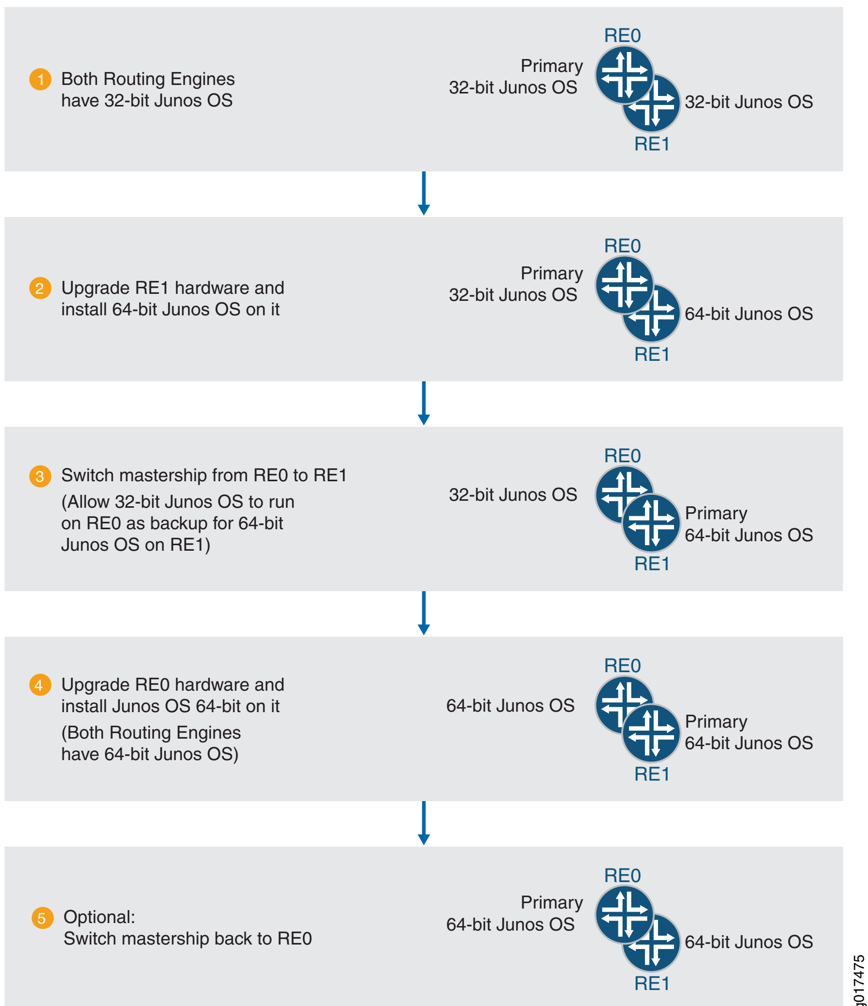
If you are starting with 32-bit Junos OS running on Routing Engines that are not 64-bit capable, there are two parts of the upgrade: upgrading the hardware and upgrading the software. This topic provides an overview of the upgrade tasks and the order in which they must be performed. For more detailed 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

For a diagram ot this procedure, see Figure 2 on page 310. For the purposes of this procedure, slot 0 has the primary Routing Engine initially.

Figure 2: 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 Device for the Installation" on page 155

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" on page 157.

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" on page 159.

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" on page 161. 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" on page 153.

3. Reboot.

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

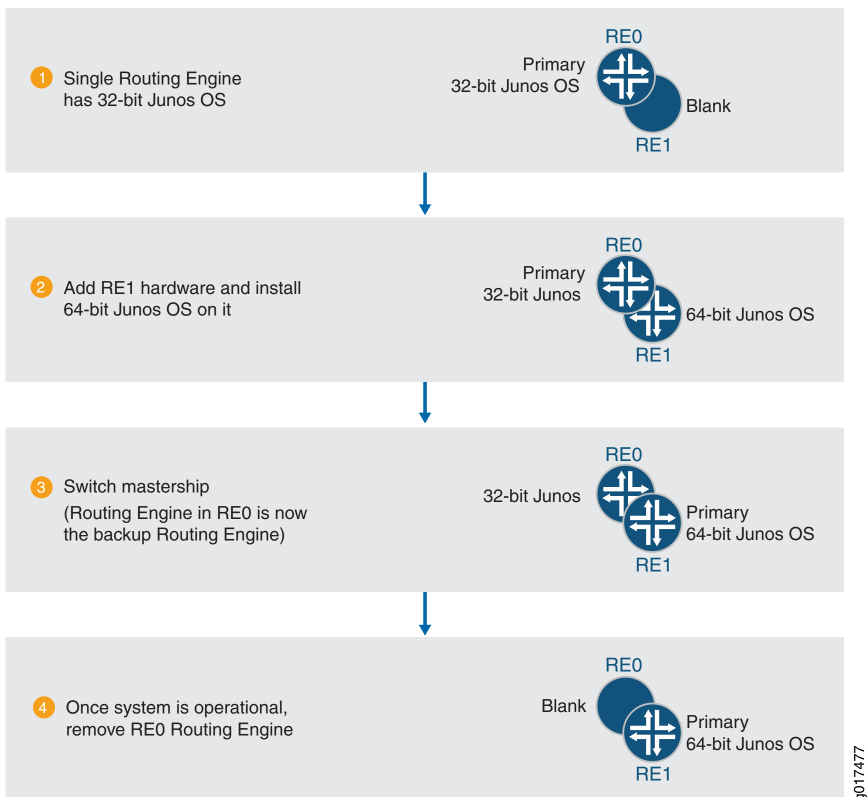
Upgrading a Single Routing Engine from 32-bit to 64-bit Junos OS Using Two Slots

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 312.

For a diagram of this procedure, see Figure 3 on page 313. In this procedure, slot 1 is initially empty.

Figure 3: 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" on page 157.

   > **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 Device with Redundant Routing Engines | 155
Installing the Software Package on a Router with a Single Routing Engine | 153

4

CHAPTER

VM Host Support on Routing Engines

VM Host Overview

IN THIS SECTION

What Are VM Hosts?

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, RE-PTX-X8, RE-QFX10002-60C.

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

The rest of this section describes the architecture of VM hosts. For more information on VM hosts, see the chapters on System Back Up and Recovery, Installing Software, Installing Firmware, and so on in this guide.

Figure 4 on page 317 illustrates the architecture of RE-MX-X6, RE-MX-X8, RE-PTX-X8, RE-QFX10002-60C Routing Engines. It comprises the following components:

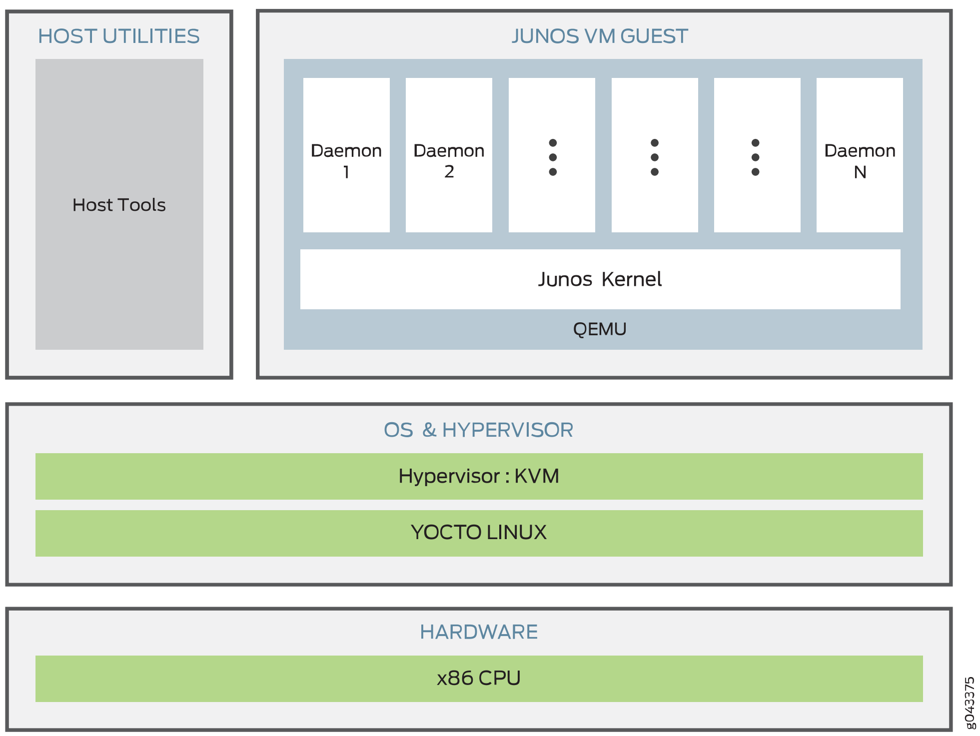
- The hardware layer
- The operating system and hypervisor layer.
- The host utilities and Junos VM guest layer.

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 4: Architecture of RE-MX-X6, RE-MX-X8, RE-PTX-X8, RE-QFX10002-60C, and RE-PTX10002-60C Routing Engines



Routing Engines with VM Host Support

The Routing Engines RE-ACX-5448, RE-MX-X6, RE-MX-X8, RE-PTX-X8, RE-QFX10002-60C, and RE-PTX10002-60C 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-ACX-5448, RE-MX-X6, RE-MX-X8, RE-PTX-X8, RCBPTX, RE-QFX10002-60C, and RE-PTX10002-60C 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.

See *Hardware Specifications of the RE-MX-X6, RE-MX-X8, RE-PTX-X8, RCBPTX, RE-QFX10002-60C, and RE-PTX10002-60C Routing Engines* for more information on hardware specifications of the Routing Engines with VMHost support.

NOTE: Platform support depends on the Junos OS release in your installation.

SEE ALSO

Supported Routing Engines by Router

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

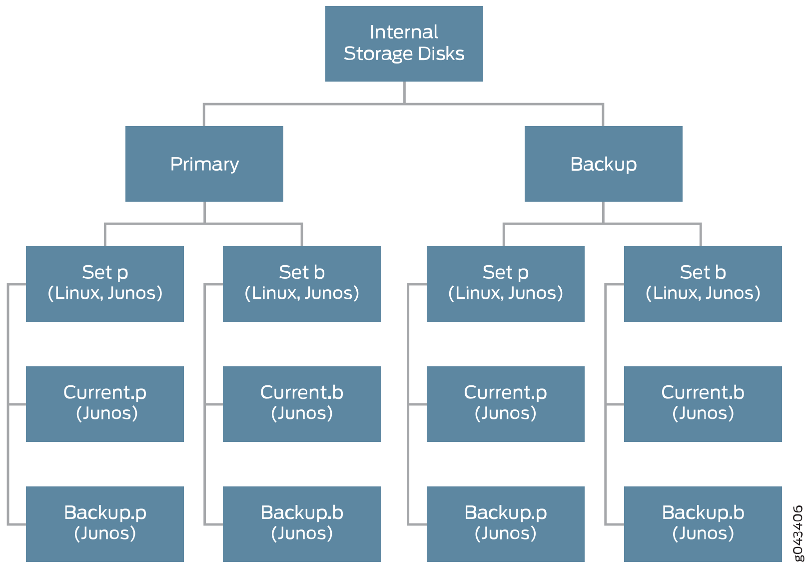
An Internal solid-state drive (SSD) is used as boot media for operating the Routing Engine. Additional options such as USB storage and network boot are available for installation and recovery purposes. A set of two 50-GB SSDs is available for normal functioning of the Routing Engine. The Routing Engine 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 operates as the primary SSD and the other as the backup SSD. Two sets of software boot images—the current set and the alternate (or previous) set are available on the primary SSD. The system boots from the current set, while the alternate set contains the previous version of the software boot image. After a software upgrade, the new version of the software is available on the alternate set. When the device is rebooted after the upgrade, the alternate set becomes the new current set and the current set, which now carries an older version of the software image, becomes the alternate set. You can switch to alternate set by using the **request vmhost software rollback** command. Until a software upgrade or a software rollback is performed, the system is programmed to boot from the same set of images on the disk.

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 5 on page 320 shows the partitioning of SSDs.

Figure 5: 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.

Starting in Junos OS Release 18.2R1, the Routing Engine on the MX2008 supports 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:

Devices	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

Devices	Routing Engine model number	SSD size
MX2010 and MX2020	RE-MX2K-X8-64G	2x100GB
	RE-MX2K-X8-64G-LT	2x100GB
	RE-MX2K-X8-128G-S	2x200GB
MX2008	REMX2008-X8-64G-LT	2x100GB
	REMX2008-X8-128G-S	2x200GB
QFX10002-60C	RE-QFX10002-60C	2x50GB
PTX10002-60C	RE-PTX10002-60C	2x50GB
MX10008	RE-X10	2x200GB

You can use the **show vmhost hardware** command to display the increased RAM size, SSD size, and other hardware information.

The following illustrations explains the partition of the host to facilitate the increased storage of core files and log files. Figure 6 on page 321 illustrates the partition of the host on MX240, MX480, MX960, MX2008, 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 6: Host partition table for Routing Engines with 200-GB SSDs

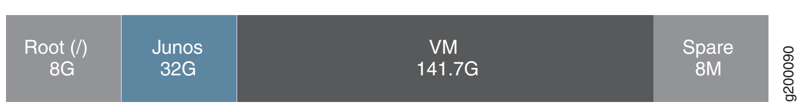
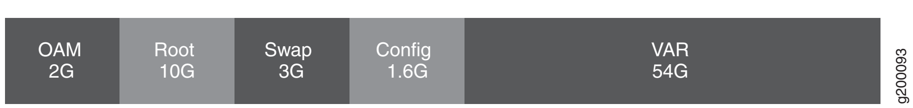


Figure 7 on page 321 illustrates the storage allocation of the guest VM.

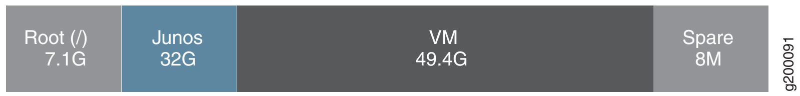
Figure 7: Partitioning of the guest VM



NOTE: For Routing Engines with 50GB SSD, the host partition remains as–is.

Figure 8 on page 322 and Figure 9 on page 322 illustrates the host partition table and the storage allocation of the guest VM for the MX2010 and MX2020 routers respectively.

Figure 8: 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 9: 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

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

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.

Release History Table

Release	Description
18.2	Starting in Junos OS Release 18.2R1, the Routing Engine on the MX2008 supports Secure Boot.

RELATED DOCUMENTATION

request vmhost snapshot | 802
request vmhost reboot | 799
request vmhost power-off | 795

Boot Process for Routers with VM Host Support

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 device 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 device 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 (PXÈ)

RELATED DOCUMENTATION

Creating an Emergency Boot Device for Routing Engines with VM Host Support | 335

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 | 523
request vmhost reboot | 799
request vmhost power-off | 795
Creating an Emergency Boot Device for Routing Engines with VM Host Support | 335

Installing, Upgrading, Backing Up, and Recovery of VM Host

You can install the Junos OS software package and host software package on the device. The following installation options are available:

NOTE: The VM Host installation works differently on the QFX10002-60C switch and PTX10002-60C router. See "Installing Software Packages on QFX Series Devices" on page 182 and *Installing the Software on PTX10002-60C Routers* for more details. However, the information on the rollback and snapshot features work the same on QFX10002-60C switches and PTX10002-60C routers.

- 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 332. After you copy the VM Host Installation Package to the PXE Boot Server, you can use the **request vmhost reboot network** command and reboot the device to install the software. The device boots from the PXE server and installs the software on both the SSDs.

You can choose to use the USB disk installation method when the device 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 335.

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 device 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 device. The files under /config and /var (except /var/etc) are preserved after the VM host upgrade.

NOTE: Before installing software on a device that has one or more custom YANG data models added to it, back up and remove the configuration data corresponding to the custom YANG data models from the active configuration. For more information see "Managing YANG Packages and Configurations During a Software Upgrade or Downgrade" on page 150.

In order to perform VM Host upgrade, use the **junos-vmhost-install-x.tgz** image. This upgrade installs the host image along with the compatible Junos OS.

NOTE: To upgrade the Junos OS on RE-S-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines, always use the VM Host Installation Package. Do not use the jinstall package.

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 device.

On the device, 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

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 devices with RE-MX-X6, RE-MX-X8,RE-PTX-X8, and RE-QFX10002-60C, and RE-PTX10002-60C Routing Engines by using the Preboot Execution Environment (PXE) boot method. This is one of the methods used for a fresh installation. A PXE boot prepares a client/server environment to boot devices by using a network interface that is independent of available data storage devices or installed operating systems. The image of the operating system is stored on a TFTP server.

To copy the installation packages to the PXE boot server:

1. Copy the downloaded installation media to the **/var/tmp** directory in the PXE boot server.

   ```
   scp
   /volume/build/junos/15.1/release/15.1F3.9/ship/junos-vmhost-install-net-x86-64-15.1F3.9.tgz
   user@host:/var/tmp/
   ```

2. Log in to the PXE boot server and verify the installation file.

   `user@host>` **ls -lh junos-vmhost-install-net-x86-64-15.1F3.9.tgz**

   ```
   -rw-r--r-- 1 root root 1.8G Oct 24 00:42
   junos-vmhost-install-net-x86-64-15.1F3.9.tgz
   ```

3. Extract the **junos-vmhost-install-net** TAR file.

   `user@host>` **tar xvzf junos-vmhost-install-net-x86-64-15.1F3.9.tgz –C /var/tmp**

```
contents/
contents/junos-vmhost-install.tgz
contents/vmhost-install-net-x86_64-15.1I20151019_1021_builder.tgz
manifest
manifest.certs
manifest.ecerts
manifest.esig
manifest.sig
package.xml
```

4. Remove the previously installed files, if any, from the **/tftpboot** directory.

   `user@host>` **rm -f /tftpboot/{vmhost-version.sh,bootpxe64.efi,vmhost-version,grub.cfg,initramfs,vmlinuz}**
   `user@host>`**ls -lh /tftpboot//**

```
total 45M

rw-r--r-- 1 root root 690K Sep  8 13:22 bootpxe.efi
-rw-rw-r-- 1  930  930  45M Oct 20 01:51
vmhost-install-net-x86_64-15.1I20151019_1021_builder.tgz
```

5. Extract the network installation package.

   `user@host>` **tar xvzf /var/tmp/contents/vmhost-install-net-x86_64-15.1I20151019_1021_builder.tgz -C /tftpboot/**

```
./
./vmhost-version.sh
./bootpxe64.efi
./vmhost-version
./grub.cfg
..
...
-rw-rw-r-- 1  930  930  45M Oct 20 01:51
vmhost-install-net-x86_64-15.1I20151019_1021_builder.tgz
-rw-rw-r-- 1  930  930    6 Oct 20 01:51 vmhost-version
-rwxrwxr-x 1  930  930  416 Oct 20 01:51 vmhost-version.sh
-rw-r--r-- 1  930  930 6.9M Oct 20 01:51 vmlinuz
```

6. Rename or delete the previously installed root file system/scripts from the **/var/install** directory. Create a new **/var/install** directory.

user@host>**mv /var/install /var/install_old**

user@host>**mkdir /var/install**

7. Extract the installation package.

   user@host>**tar xvzf /var/tmp/contents/junos-vmhost-install.tgz -C /var/install**

```
./
./vmhost-pkgs-version
./vm/
./vm/note
./vm/grub.cfg.ngre
./vm/vsmartd-1.0-0.x86_64.rpm
./vm/re_fpga-1.0-0.x86_64.rpm
./vm/veccd-1.0-0.x86_64.rpm
./vmhost-version.sh
./vmhost/
./vmhost/vmhost-x86_64-15.1I20151019_1021_builder.img.gz
...
...
./junos/junos-mtre-upgrade.sh
./vmhost-core-x86_64-15.1I20151019_1021_builder.tgz
./junos/
./junos/junos-install-x86-64-15.1F3.9.img.gz
```

8. Set permissions for the files in the **/var/install** and **/tftpboot** directories.

```
user@host> chown root:root /tftpboot/*
user@host>chmod a+rwx /tftpboot/*
user@host>chown -R root:root /var/install
user@host>chmod -R a+rwx /var/install
```

9. Exit the PXE boot server.

```
user@host> exit
```

RELATED DOCUMENTATION

Installing, Upgrading, Backing Up, and Recovery of VM Host | 326

Creating an Emergency Boot Device for Routing Engines with VM Host Support | 335

Creating an Emergency Boot Device for Routing Engines with VM Host Support

If Junos OS on your device is damaged during loading in a way that prevents it 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. 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 device with RE-MX-X6, RE-MX-X8, RE-PTX-X8, RCBPTX, RE-QFX10002-60C, and RE-PTX10002-60C Routing Engines:

1. Copy the installation media into the device's **/var/tmp** directory.

2. Insert the USB storage device into the device's USB port.

3. In the UNIX shell, navigate to the **/var/tmp** directory:

   ```
   start shell
   cd /var/tmp
   ```

4. Log in as **su**:

   Super User (su) is one of the predefined login classes with preset permissions.

   ```
   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 400.
- ***downloaded***.*img*—Refers to the installation media copied to the **/var/tmp** directory. For example, **junos-vmhost-install-usb-ptx-x86-64-15.1F6.8.img**.

The following code example can be used to create an emergency boot device by using a USB storage device:

```
dd if=/path/to/junos-vmhost-install-usb-mx-x86-64-15.1F6.8.img of=/dev/da0 bs=4M
```

> **NOTE:** In the dd command, use junos-vmhost-install-usb-mx-86 for RE-MX-X6 and RE-MX-X8 Routing Engines and junos-vmhost-install-ptx-86 for RE-PTX-X8 Routing Engine respectively.

7. Log out as **su**:

```
exit
```

RELATED DOCUMENTATION

Boot Process for Routers with VM Host Support | 324

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, on QFX10002-60C switches with the RE-QFX10002-60C Routing Engines, and PTX10002-60C routers with the RE-PTX10002-60C 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 must upgrade SSD firmware only under the direction of a Juniper Networks support representative.

> **NOTE:** On QFX10002-60C switches, you can upgrade firmware only for the FPGA and BIOS, not the SSD.

Before you begin upgrading the firmware, check the current firmware version of the SSD.

`user@host>` **show system firmware**

```
Part                Type     Tag    Current   Available Status
                                          version   version
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 you can use the following procedure 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: In releases before Junos OS Release 18.3R1, you must upgrade the SSD on a master Routing Engine only. For upgrading firmware on the backup Routing Engine, switch mastership by using the following command and then log in to the backup Routing Engine, which is now the new master Routing Engine:.

   user@host> request chassis routing-engine master switch

   Starting in Junos OS Release 18.3R1, you can upgrade the SSD firmware on the master and backup Routing Engines.

   To initiate the upgrade, use the following command:

   `user@host>` **request system firmware upgrade re ssd disk1**

```
Part     Type      Tag    Current   Available    Status
                             version   version
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.
```

   Monitor the upgrade status by using the **show system firmware** command.

   `user@host>` **show system firmware**

```
Part                 Type    Tag    Current   Available Status
                                          version   version
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   Available Status
                                          version   version
```

```
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   Available   Status
                        version   version
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 by using the **show system firmware** command.

   `user@host>` **show system firmware**

```
Part                 Type    Tag    Current   Available Status
                                          version   version
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   Available  Status
                                           version   version
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, on QFX10002-60C switches with the RE-QFX10002-60C Routing Engines, and PTX10002-60C routers with the RE-PTX10002-60C Routing Engines.

Upgrading the i40e NVM Firmware on Routing Engines with VM Host Support

In order to install VM Host image based on Linux WRL9, you have to upgrade the i40e NVM firmware to version 6.01. Starting in Junos OS Release 19.3R1, in order to install VM Host image based on Linux WRL9, you have to upgrade the i40e NVM firmware to version 6.01.

Table 19 on page 342 lists the Junos OS releases which support i40e NVM firmware upgrade.

Table 19: Junos OS Releases which Support i40e NVM Firmware Upgrade

Platform	15.x	16.x	17.x	18.x	19.x
PTX5000	15.1F6-S11	16.1R7	17.1R3 / 17.2R3 / 17.3R3 / 17.4R2	18.1R1 / 18.2R1 / 18.3R1 / 18.4R1	19.1R1 or later
PTX3000	Not applicable	Not applicable	Not applicable	18.2R3 / 18.3R3 / 18.4R2	19.1R2 / 19.2R1
MX240 / MX480 / MX960 / MX2010 / MX2020	15.1F6-S11	16.1R7	17.1R3 / 17.2R3 / 17.3R3 / 17.4R2	18.1R1 / 18.2R1 / 18.3R1 / 18.4R1	19.1R1 or later
MX2008	Not applicable	Not applicable	Not applicable	18.2R3 / 18.3R3 / 18.4R2	19.1R2 / 19.2R1
MX10016/MX10008 PTX10016/MX10008	Not applicable	Not applicable	Not applicable	18.2R3 / 18.3R3 / 18.4R2	19.1R2 / 19.2R1

You can install older Junos OS images on the Routing Engine with an upgraded i40e NVM firmware as it supports i40e 2.4.3 driver versions. If you install an older version of the VM Host image, which is not listed in Table 20 on page 342, using USB, the Routing Engine does not start up properly. In such a case, you can reinstall the VMHost image with a version which supports the new i40e NVM firmware.

Table 20: Junos OS Versions which Support i40e 2.4.3 Driver Versions

Platform	i40e-1.1.23/ NVM-4.26 Support	i40e-2.4.3/ NVM-6.01 Support				
PTX5000	15.1F3	15.1F6-S11	16.1R7	17.1R3/ 17.2R3 / 17.3R3/ 17.4R2	18.1R1 / 18.2R3 / 18.3R1 / 18.4R1	19.1R1 or later

Table 20: Junos OS Versions which Support i40e 2.4.3 Driver Versions *(continued)*

Platform	i40e-1.1.23/ NVM-4.26 Support	i40e-2.4.3/ NVM-6.01 Support				
PTX3000	16.1R4	Not applicable	16.1R7	17.3R3-S6 / 17.4R2-S7	18.2R3-S8 / 18.2R3 / 18.3R3 / 18.4R2	19.1R2/ 19.2R1
MX240/MX480/MX960	15.1F3	15.1F6-S11	16.1R7	17.1R3 / 17.2R3 / 17.3R3 / 17.4R2	18.1R1 / 18.2R1 / 18.3R1 / 18.4R1	19.1R1 or later
MX2010/MX2020	15.1F5-S1	15.1F6-S11	16.1R7	17.1R3 / 17.2R3 / 17.3R3 / 17.4R2	18.1R1 / 18.2R1 / 18.3R1 / 18.4R1	19.1R1 or later
MX2008	15.1F7	Not applicable	Not applicable	17.2R3 / 17.3R3 / 17.4R2	18.1R1 / 18.2R1 / 18.3R1 / 18.4R1	19.1R1 or later
MX10016/MX10008 PTX10016/MX10008	18.2R1	Not applicable	Not applicable	Not applicable	18.2R1 / 18.3R1 / 18.4R1	19.1R1 or later

NOTE:

- You must upgrade i40e NVM firmware only under the direction of a Juniper Networks support representative. Once you upgrade the NVM firmware, a downgrade action is not supported.
- You must implement this procedure with a router console access. Also, you have to perform power cycling of the routing Engine multiple times during the firmware upgrade process.

Before you begin upgrading the firmware, check the current firmware version of the i40e NVM.

```
user@host> show system firmware
```

```
Part              Type           Tag Current    Available    Status
                                     version    version
Routing Engine 0   RE BIOS       0      0.53.1               OK
Routing Engine 1 RE BIOS         0      0.43         0.53    OK
Routing Engine 1 RE FPGA         1      28.0.0       41.0    OK
Routing Engine 1 RE SSD1         3      0.0.0                OK
Routing Engine 1 RE SSD2         3      0.0.0                OK
Routing Engine 1 RE i40e-NVM     7      4.26                 OK
```

If the value of **Current version** is less than **6.01**, then you can use the following procedure for the i40e NVM firmware upgrade.

To upgrade i40e NVM firmware on routers with single Routing Engine:

1. Upgrade the device with the Junos OS image version which supports i40e NVM firmware upgrade. See Table 19 on page 342.

   Copy and install the jfirmware-vmhost 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 vmhost software add /var/tmp/jfirmware-vmhost-x86-64-19.2R1.tgz

2. Upgrade the NVM firmware.

   To initiate the upgrade, use the following command:

   `user@host>` **request system firmware upgrade re i40nvm**

```
Part              Type           Tag    Current    Available   Status
                                         version    version
Routing Engine 1 RE i40e-NVM    7      4.26       6.01          OK
Perform indicated firmware upgrade ? [yes,no] (no) yes

Firmware upgrade initiated, use "show system firmware" after reboot to verify
the firmware version
```

   Monitor the upgrade status by using the **show system firmware** command. If the upgrade is initiated the output displays **PROGRAMMING (0%)** as the status. However, note that the status **PROGRAMMING (0%)** does not increment during the process.

   `user@host>` **show system firmware**

```
Part              Type            Tag Current   Available Status
                                      version   version
Routing Engine 0 RE BIOS         0            0.53.1            OK
Routing Engine 1 RE BIOS         0            0.43              OK
Routing Engine 1 RE FPGA         1            28.0.0            OK
Routing Engine 1 RE SSD1         3            0.0.0             OK
Routing Engine 1 RE SSD2         3            0.0.0             OK
Routing Engine 1 RE i40e-NVM     7        4.26 6.01          PROGRAMMING   (0%)
```

3. Reboot the device by using the **request vmhost reboot** command.

   `user@host>` **request vmhost reboot**

4. Verfiy the progress of i40e NVM upgrade on the console. You may have to perform power recycle of the Routing Engine multiple times. When you are prompted for a power cycle on your console, use external power cycle for power cycling the Routing Engine.

   The following message is displayed on the console prompting you to perform a power cycle:

```
“Please Power Cycle your system now and run the NVM update utility again to
complete the update. Failure to do so will result in an incomplete NVM update.
Upgrade complete please power reboot
You may notify to  power reboot again after reboot if required”
```

5. After a successful upgrade, verify the version of the firmware.

   NOTE: The **Current version** is displayed as 6.1 instead of 6.01.

   `user@host>` **show system firmware**

```
Part              Type            Tag Current   Available Status
                                      version   version
Routing Engine 0 RE BIOS            0          0.53.1          OK
Routing Engine 1 RE BIOS            0          0.43            OK
Routing Engine 1 RE FPGA            1          28.0.0          OK
Routing Engine 1 RE SSD1            3         0.0.0           OK
Routing Engine 1 RE SSD2            3         0.0.0           OK
Routing Engine 1 RE i40e-NVM   7    6.1       6.01            OK
```

NOTE: In case, you have run the **request vmhost snapshot** command with a Junos OS image which does not support i40e NVM firmware upgrade, (if the SSD recovery snapshot has a Junos OS version older than the Junos OS versions mentioned in Table 20 on page 342) we recommend you to take a snapshot using the **request vmhost snapshot** command again. Hence, in case of a recovery process, the SSD recovery snapshot will have a Junos OS image which supports NVM 6.01.

On routers with dual Routing Engines, you must use the **request chassis cb (offline | online) slot *slot-number*** to power cycle the Routing Engine. Thereby, you can avoid using an external power cycler and avoid abrupt power cycling of backup RE, which may cause file system errors.

To upgrade i40e NVM firmware on routers with dual Routing Engines:

NOTE: You must disable GRES before proceeding with the upgrade procedure.

1. Upgrade the device with the Junos OS image version which supports i40e NVM firmware upgrade. See Table 19 on page 342.

   Copy and install the jfirmware-vmhost 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 vmhost software add /var/tmp/jfirmware-vmhost-x86-64-19.2R1.tgz

2. Upgrade the NVM firmware.

   To initiate the upgrade, use the following command:

   `user@host>` **request system firmware upgrade re i40nvm**

```
Part              Type           Tag   Current    Available  Status
                                        version    version
Routing Engine 1 RE i40e-NVM    7     4.26      6.01         OK
Perform indicated firmware upgrade ? [yes,no] (no) yes

Firmware upgrade initiated, use "show system firmware" after reboot to verify
the firmware version
```

Monitor the upgrade status by using the **show system firmware** command. If the upgrade is initiated the output displays **PROGRAMMING (0%)** as the status. However, note that the status **PROGRAMMING (0%)** does not increment during the process.

`user@host>` **show system firmware**

```
Part              Type           Tag Current   Available Status
                                     version   version
Routing Engine 0 RE BIOS        0           0.53.1             OK
Routing Engine 1 RE BIOS        0           0.43               OK
Routing Engine 1 RE FPGA        1           28.0.0             OK
Routing Engine 1 RE SSD1        3           0.0.0              OK
Routing Engine 1 RE SSD2        3           0.0.0              OK
Routing Engine 1 RE i40e-NVM    7        4.26 6.01          PROGRAMMING   (0%)
```

3. Switch to the backup Routing Engine by using the **request chassis routing-engine master switch** command to switch mastership to other RE (i.e, RE1).

   > **NOTE:** This step is necessary, because in Step 6 you have to power cycle the Routing Engine which is undergoing the NVM upgrade (i.e, RE0) from RE1

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

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

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

4. Reboot the device by using the **request vmhost reboot** command from the Routing Engine which is undergoing the NVM upgrade (i.e, RE0).

   `user@host>` **request vmhost reboot**

5. Monitor the console output. You may have to perform power recycle of the Routing Engine multiple times. When you are prompted for a power cycle on your console, use external power cycle for power cycling the Routing Engine. Or, you can use the command **request chassis cb slot *slot* offline** as described in Step 6.

   The following message is displayed on the console prompting you to perform a power cycle:

```
“Please Power Cycle your system now and run the NVM update utility again to
complete the update. Failure to do so will result in an incomplete NVM update.
Upgrade complete please power reboot
You may notify to  power reboot again after reboot if required”
```

6. From RE1, power cycle the RE0 using following command **request chassis cb slot *slot* offline**.

   To power off RE0, use the command **request chassis cb slot 0 offline** and to power on RE0, use the command **request chassis cb slot 0 online**.

   `user@host>` **request chassis cb slot 0 offline**

   ```
   Offline initiated, use "show chassis environment cb" to verify
   ```

   `user@host>` **request chassis cb slot 0 online**

   ```
   Online initiated, use "show chassis environment cb" to verify
   ```

7. After a successful upgrade, verify the version of the firmware.

   NOTE: The **Current version** is displayed as 6.1 instead of 6.01.

   `user@host>` **show system firmware**

   ```
   Part              Type           Tag Current   Available Status
                                        version   version
   Routing Engine 0 RE BIOS            0         0.53.1          OK
   Routing Engine 1 RE BIOS            0         0.43            OK
   Routing Engine 1 RE FPGA            1         28.0.0          OK
   Routing Engine 1 RE SSD1            3        0.0.0           OK
   Routing Engine 1 RE SSD2            3        0.0.0           OK
   Routing Engine 1 RE i40e-NVM  7   6.1        6.01            OK
   ```

> **NOTE:** In case, you have run the **request vmhost snapshot** command with a Junos OS image which does not support i40e NVM firmware upgrade, (if the SSD recovery snapshot has a Junos OS version older than the Junos OS versions mentioned in Table 20 on page 342) we recommend you to take a snapshot using the **request vmhost snapshot** command again. Hence, in case of a recovery process, the SSD recovery snapshot will have a Junos OS image which supports NVM 6.01.

8. Similarly, while upgrading i40e NVM on RE1, power cycle RE1 from RE0.

   `user@host>` **request chassis cb slot 1 offline**

   ```
   Offline initiated, use "show chassis environment cb" to verify
   ```

   `user@host>` **request chassis cb slot 1online**

   ```
   Online initiated, use "show chassis environment cb" to verify
   ```

Release History Table

Release	Description
19.3R1	Starting in Junos OS Release 19.3R1, in order to install VM Host image based on Linux WRL9, you have to upgrade the i40e NVM firmware to version 6.01.

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 | 523

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 in-service-upgrade**

  NOTE: This command is not supported on the QFX10002-60C and PTX10002-60C devices.

- **request vmhost software add**
- **request vmhost software in-service-upgrade**

  NOTE: This command is not supported on the QFX10002-60C and PTX10002-60C devices.

- **request vmhost software rollback**
- **request vmhost zeroize**

RELATED DOCUMENTATION

Routing Engines with VM Host Support | 318

5

CHAPTER

Installing and Upgrading the BIOS and Firmware

Upgrading BIOS and Firmware

IN THIS SECTION

You can upgrade BIOS, back up the BIOS, and upgrade automatically on your SRX Series devices.

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, SRX345, and SRX380 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 21 on page 354 Lists the CLI commands used for manual BIOS upgrade.

Table 21: CLI Commands for Manual BIOS Upgrade

Active BIOS	Backup BIOS
request system firmware upgrade re bios	**request system firmware upgrade re bios backup**

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 upgade 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 upgade 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.

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 configuration 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. The **set chassis routing-engine bios uninterrupt** command is introduced in Junos OS Release 20.1R1 for SRX380 Series devices.

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. The **set chassis routing-engine bios uninterrupt** command is introduced in Junos OS Release 20.1R1 on SRX380 Series devices.

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 in case 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

Installing Software on SRX Series Devices | 252

Installing and Upgrading Firmware

IN THIS SECTION

To get the optimal network performance, and to fix a vulnerability, you can upgrade the firmware on your device.

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]
```

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

```
Hostname: YYY

Model: ACX6360-OR
Junos: 18.3I20180430_1917_XXX
JUNOS OS Kernel 64-bit (WITNESS) [20180413.173511_fbsd-builder_stable_11]
JUNOS OS libs [20180413.173511_fbsd-builder_stable_11]
JUNOS OS runtime [20180413.173511_fbsd-builder_stable_11]
JUNOS OS time zone information [20180413.173511_fbsd-builder_stable_11]
...
JUNOS jfirmware [20180430.191738_XXX_dev_common]
JUNOS Online Documentation [20180430.191738_XXX_dev_common]
...
```

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 software 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 364 or "Installing Firmware on the 5-Port 100-Gigabit DWDM OTN PIC (PTX-5-100G-WDM)" on page 361. For information about how to install the firmware package on ACX6360 router, see "Installing Firmware on ACX6360 Router" on page 367.

   `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]
   ```

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

   ```
   Hostname: YYY

   Model: ACX6360-OR
   Junos: 18.3I20180430_1917_XXX
   JUNOS OS Kernel 64-bit (WITNESS) [20180413.173511_fbsd-builder_stable_11]
   JUNOS OS libs [20180413.173511_fbsd-builder_stable_11]
   JUNOS OS runtime [20180413.173511_fbsd-builder_stable_11]
   JUNOS OS time zone information [20180413.173511_fbsd-builder_stable_11]
   ```

```
...

JUNOS IDP Services [20180430.191738_XXX_dev_common]
...
JUNOS Online Documentation [20180430.191738_XXX_dev_common]
...
```

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.

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 358.

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 Device with Redundant Routing Engines" on page 155 or "Installing the Software Package on a Router with a Single Routing Engine" on page 153 for more information.

2. Download the firmware package from https://support.juniper.net/support/downloads/. For information about downloading software packages, see "Downloading Software" on page 140.

   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 software add *path*/*package-name*** command. For example, to install the **jfirmware-15.1F6.9.tgz** package:

   ```
   user@host> request system software 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.

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 358.

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> 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   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.

4. Restart the FPC that the PIC is installed in by using the **request chassis fpc *fpc-slot* restart** command.

5. (Optional) After the firmware upgrade is successfully completed, uninstall the firmware package from the router by using the **request system software delete** command.

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 358.

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 Device with Redundant Routing Engines" on page 155 or "Installing the Software Package on a Router with a Single Routing Engine" on page 153 for more information.

2. Download the firmware package from https://support.juniper.net/support/downloads/. For information about downloading software packages, see "Downloading Software" on page 140.

   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 software add /var/*path*/*package-name*** command. For example, to install the **jfirmware-x86-32-15.1F6.9.tgz** package:

```
user@host> request system software 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.

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 358.

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> **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
```

> **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.

4. Restart the MPC that the MIC is installed in by using the **request chassis fpc *fpc-slot* restart** command.

5. (Optional) After the firmware upgrade is successfully completed, uninstall the firmware package from the router by using the **request system software delete** command.

Installing Firmware on ACX6360 Router

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 358.

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 Device with Redundant Routing Engines" on page 155 or "Installing the Software Package on a Router with a Single Routing Engine" on page 153 for more information.

2. Download the firmware package from https://support.juniper.net/support/downloads/. For information about downloading software packages, see "Downloading Software" on page 140.

   NOTE: Download the firmware package specific to your router. The firmware package for ACX Series routers is different from the firmware package for the MX or 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 software add /var/*path*/*package-name*** command. For example, to install the **jfirmware-x86-32-15.1F6.9.tgz** package:

```
user@host> request system software 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: YYY

Model: ACX6360-OR
Junos: 18.3I20180430_1917_XXX
```

```
JUNOS OS Kernel 64-bit (WITNESS) [20180413.173511_fbsd-builder_stable_11]
JUNOS OS libs [20180413.173511_fbsd-builder_stable_11]
JUNOS OS runtime [20180413.173511_fbsd-builder_stable_11]
JUNOS OS time zone information [20180413.173511_fbsd-builder_stable_11]
...
JUNOS jfirmware [20180430.191738_XXX_dev_common]
JUNOS Online Documentation [20180430.191738_XXX_dev_common]
JUNOS jail runtime [20180413.173511_fbsd-builder_stable_11]
....
```

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.

Upgrading Firmware on the ACX6360 Router

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 358.

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
Pseudo CB 0       CB FPGA           0   2.12.0    2.12.0    OK
Pseudo CB 0       PORT FPGA         9   1.14.0    1.15.0    OK
Pseudo CB 0       TIC FPGA          11  4101.5.0  4101.5.0  OK
FPC 0                               0   0.0.0     71.63d    OK
PIC 1             DWDM DCO-0/1/0    20  38.1.9    38.2.6    OK
PIC 1             DWDM DCO-0/1/1    21  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/2    22  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/3    23  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/4    24  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/5    25  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/6    26  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/7    27  1.0.0     38.2.6    OK
Routing Engine 0 RE BIOS            7   0.24.1    0.24.01   OK
Routing Engine 0 RE FPGA            2   9.6.0     9.9.0     OK
Routing Engine 0 RE SSD1            3   12028               OK
Routing Engine 0 RE SSD2            4   12028               OK
Power Supply 0                      0   0.0.0               OK
Power Supply 1                      0   0.0.0               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, for ACX6360 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 fpc-slot 0 pic-slot 1**

```
Part              Type              Tag Current   Available Status
                                        version   version
FPC 0
PIC 1             DWDM DCO-0/1/0    20  38.2.9    38.2.6    OK
Perform indicated firmware upgrade ? [yes,no] (no)
```

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>` **show system firmware**

```
Part              Type              Tag Current   Available Status
                                        version   version
Pseudo CB 0       CB FPGA           0   2.12.0    2.12.0    OK
Pseudo CB 0       PORT FPGA         9   1.14.0    1.15.0    OK
Pseudo CB 0       TIC FPGA          11  4101.5.0  4101.5.0  OK
FPC 0                               0   0.0.0     71.63d    OK
PIC 1             DWDM DCO-0/1/0    20  38.2.6    38.2.6    OK
PIC 1             DWDM DCO-0/1/1    21  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/2    22  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/3    23  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/4    24  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/5    25  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/6    26  1.0.0     38.2.6    OK
PIC 1             DWDM DCO-0/1/7    27  1.0.0     38.2.6    OK
Routing Engine 0 RE BIOS            7   0.24.1    0.24.01   OK
Routing Engine 0 RE FPGA            2   9.6.0     9.9.0     OK
Routing Engine 0 RE SSD1            3   12028               OK
Routing Engine 0 RE SSD2            4   12028               OK
Power Supply 0                      0   0.0.0               OK
Power Supply 1                      0   0.0.0               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.

4. Restart the ACX6360 router by using **request chassis fpc restart slot 0** command for the upgrade to take effect.

5. (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

show system firmware

Understanding the MIC3-100G-DWDM MIC

Configuring OTN Interfaces on MIC3-100G-DWDM MIC

Understanding the PTX-5-100G-WDM PIC

Configuring OTN Interfaces on PTX-5-100G-WDM PIC

6

CHAPTER

Configuring Root Partitions

Configuring Dual-Root Partitions

IN THIS SECTION

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)

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 22 on page 374.

Table 22: Resilient Dual-Root Partition Scheme

Slice 1	Slice 2	Slice 3		Slice 4
s1a	s2a	s3e	s3d	s4d
/ (root Junos OS)	/ (root Junos OS)	**/var**	**/var/tmp**	**/config**

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 23 on page 376.

Table 23: 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 22 on page 374.

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 | 495

Configuring Root Partitions on SRX Series Devices

IN THIS SECTION

The dual-root partitions help your SRX Series devices to remain functional even if the file system is corrupted. Also, it helps to recover the file system in case of corruption.

Dual-Root Partitioning Scheme on SRX Series Devices

IN THIS SECTION

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

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 24 on page 378 provides information on the storage media available on SRX Series devices.

Table 24: Storage Media on SRX Series Devices

SRX Series Devices	Storage Media
SRX100, SRX210, and SRX240	• Internal NAND flash (default; always present) • USB storage device (alternate)
SRX110, SRX220	• CompactFlash (default; always present) • USB storage device (alternate)
SRX300, SRX320, and SRX340, and SRX345	• eUSB disk (default; always present) • USB storage device (alternate)

Table 24: Storage Media on SRX Series Devices *(continued)*

SRX Series Devices	Storage Media
SRX380	• Internal SSD (default, always present). • USB storage device (alternate)
SRX550	• Internal CF (default; always present) • USB storage device (alternate)
SRX550M	• Internal CF (default; always present) • USB storage device (alternate)
SRX650	• 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, SRX380, 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 270
- 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 273
- 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.

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.

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

Installing Software on SRX Series Devices | 252

Configuring Root Partitions on ACX Series Routers

IN THIS SECTION

The dual-root partitions help your ACX Series routers to remain functional even if the file system is corrupted. Also, it helps to recover the file system in case of corruption.

Dual-Root Partitioning ACX Series Routers Overview

IN THIS SECTION

Dual-root partitioning allows the ACX Series router to remain functional even if there is file system corruption and to facilitate easy recovery of the file system. Dual-root partitioning means that the primary and backup Junos OS images are kept 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.

NOTE: ACX5048 and ACX5096 routers do not support dual-root partitioning. All other ACX routers run with dual-root partitioning.

This section contains the following topics:

Boot Media and Boot Partition on the ACX Series Routers

With dual-root partitioning, the ACX Series router first tries to boot the Junos OS from the primary root partition and then from the backup root partition on the internal NAND flash. If both primary and backup root partitions of the internal NAND flash fail to boot, you must insert a USB storage media with a copy of the Junos OS from which to boot.

The following is the storage media available on the ACX Series router:

- USB media emergency boot device

  NOTE: The USB media device is not dual-root partitioned.

- Dual, internal NAND flash device (first daOs1, then daOs2)

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.

Understanding How the Primary Junos OS Image with Dual-Root Partitioning Recovers on the ACX Series Router

If the ACX Series Universal Metro router 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.

NOTE: ACX5048 and ACX5096 routers do not support dual-root partitioning.

```
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 should immediately restore the primary Junos OS image using one of the following methods:

- Install a new image using the CLI. When you install the new image, the new image is installed on only one partition–the alternate partition, meaning the router is now running two images. When you reboot, the router boots from the newly installed image, which becomes the primary image. So now there are two different images running on the router. Run the installation process again to update the other partition.
- Use a snapshot of the backup root partition by entering the **request system snapshot slice alternate** command. After 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.

  You can use this command to:

  - Save an image of the primary root partition in the backup root partition when the system boots from the primary root partition.
  - Save an image of the backup root partition in the primary root partition when the 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.

Junos OS Release 12.2 or Later Upgrades with Dual-Root Partitioning on ACX Series Routers

NOTE: If you are upgrading to Junos OS Release 12.2 without transitioning to dual-root partitioning, use the conventional CLI installation method.

To format the media with dual-root partitioning while upgrading to Junos OS Release 12.2 or later, use either of the following installation methods:

NOTE: ACX5048 and ACX5096 routers do not support dual-root partitioning. All other ACX routers run with dual-root partitioning.

- Installation 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 Using a USB Storage Device on ACX Series Routers*.
- Installation from the CLI. We recommend this method only if console access is not available. This installation can be performed remotely. See *Installing Junos OS Upgrades from a Remote Server on ACX Series Routers*.

Example: Installing Junos OS and Configuring a Dual-Root Partition on ACX Series Routers Using the CLI

IN THIS SECTION

This example shows how to install Junos OS Release 12.2 or later and configure a dual-root partition on ACX Series routers with the CLI.

Requirements

This example requires an ACX Series router. Before you begin, back up any important data.

Overview

This example formats the NAND Flash device and installs the new Junos OS image on the media with dual-root partitioning. Install the Junos OS Release 12.2 or later image from the CLI by using the **request system software add** command. Partitions are automatically created on ACX Series routers and no option needs to be manually entered for creating partitions. This command copies the image to the device, and then reboots the device for installation. The device boots with the Release 12.2 or later image installed with the dual-root partitioning scheme. The formatting and installation 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.

> CAUTION: Using the **request system software add** command erases the existing contents of the media. Only the current configuration is preserved. You should back up any important data before starting the process.

NOTE: Dual, internal NAND Flash device (first daOs1, then daOs2) and USB storage device are the storage media available on the ACX Series router. The USB storage device is not dual-root partitioned.

In this example, add the software package junos-juniper-12.2R1.9-domestic.tgz with the following options:

- **no-copy** option to install the software package. However, do not save the copies of the package files. You should 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 reboot the device after installation is completed.

Configuration

CLI Quick Configuration

To install Junos OS Release 12.2 or later and configure dual-root partitioning on ACX Series routers, copy the following command, paste it in a text file, remove any line break, and then copy and paste the command into the CLI.

From operational mode, enter:

```
user@host>request system software add junos-juniper-12.2R1.9-domestic.tgz no-copy no-validate reboot
```

Step-by-Step Procedure

To install Junos OS Release 12.2 or later and configure a dual-root partition:

1. Upgrade the ACX Series router to Junos OS Release 12.2 or later using the CLI.

2. Install Junos OS Release 12.2 or later and configure the dual-root partition.

```
user@host>request system software add junos-juniper-12.2R1.9-domestic.tgz no-copy no-validate reboot
Copying package junos-juniper-12.2R1.9-domestic.tgz to var/tmp/install
Rebooting ...
```

Results

In operational mode, confirm your configuration by entering the **show system storage** 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 dual-root partitioning that displays information about the root partition that is mounted (only one root partition is mounted at a point in time):

```
user@host> show system storage
```

```
Filesystem              Size       Used      Avail  Capacity   Mounted on
/dev/da0s1a             872M       150M       713M       17%  /
devfs                   1.0K       1.0K         0B      100%  /dev
/dev/md0                 41M        41M         0B      100%  /packages/mnt/jbase
/dev/md1                183M       183M         0B      100%
/packages/mnt/jkernel-ppc-12.2I20121026_1217_sranjan
/dev/md2                 30M        30M         0B      100%
/packages/mnt/jpfe-ACX-12.2I20121026_1217_sranjan
/dev/md3                9.1M       9.1M         0B      100%
/packages/mnt/jdocs-12.2I20121026_1217_sranjan
/dev/md4                 55M        55M         0B      100%
/packages/mnt/jroute-ppc-12.2I20121026_1217_sranjan
/dev/md5                 12M        12M         0B      100%
/packages/mnt/jcrypto-ppc-12.2I20121026_1217_sranjan
/dev/md6                1.0G       8.0K       951M        0%  /tmp
/dev/md7                1.0G       448K       950M        0%  /mfs
/dev/da0s1e              92M        18K        91M        0%  /config
procfs                  4.0K       4.0K         0B      100%  /proc
/dev/da0s3f             3.9G       3.6G        30M       99%  /var
/dev/da0s3d             447M       2.8M       409M        1%  /var/log
```

If you are done configuring the device, enter **commit** in configuration mode.

You can issue the **fdisk** command from the Junos prompt to display information about the entire partition format on the NAND Flash device. All ACX Series routers run with dual-root partitioning. The following example displays the partition details on an ACX Series router with dual-root partitions:

```
user@host% fdisk
```

```
******* Working on device /dev/da0 *******
parameters extracted from in-core disklabel are:
cylinders=487 heads=255 sectors/track=63 (16065 blks/cyl)

parameters to be used for BIOS calculations are:
cylinders=487 heads=255 sectors/track=63 (16065 blks/cyl)
```

```
Media sector size is 512
Warning: BIOS sector numbering starts with sector 1
Information from DOS bootblock is:
The data for partition 1 is:
sysid 165 (0xa5),(FreeBSD/NetBSD/386BSD)
    start 567, size 1011528 (493 Meg), flag 80 (active)
        beg: cyl 0/ head 9/ sector 1;
        end: cyl 62/ head 254/ sector 63
The data for partition 2 is:
sysid 165 (0xa5),(FreeBSD/NetBSD/386BSD)
    start 1012662, size 1011528 (493 Meg), flag 0
        beg: cyl 63/ head 9/ sector 1;
        end: cyl 125/ head 254/ sector 63
The data for partition 3 is:
sysid 165 (0xa5),(FreeBSD/NetBSD/386BSD)
    start 2024757, size 3581928 (1748 Meg), flag 0
        beg: cyl 126/ head 9/ sector 1;
        end: cyl 348/ head 254/ sector 63
The data for partition 4 is:
sysid 165 (0xa5),(FreeBSD/NetBSD/386BSD)
    start 5607252, size 2200338 (1074 Meg), flag 0
        beg: cyl 349/ head 9/ sector 1;
        end: cyl 485/ head 254/ sector 63
```

In the preceding example, partition 1 and 2 contain two partitions each internally, a root partition and a configuration partition.

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the Partitioning Scheme Details

Purpose

Verify that the partitioning scheme details on the ACX Series router were configured.

Action

In operational mode, enter the **show system storage** command. For details about the output of this command and the descriptions of the output fields, see *show system storage*.

RELATED DOCUMENTATION

Installing Software on ACX Series Routers | 293

7

CHAPTER

Storage Media and Routing Engines

Storage Media and Routing Engines

IN THIS SECTION

Routing Engine and Packet Forwarding Engine (PFE) are the two primary components of Juniper Networks platforms. Junos OS software is installed on the routing engine and it is stored in storage media.

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 Universal Routing Platforms; 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 User Guide for Routing Devices*.

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.

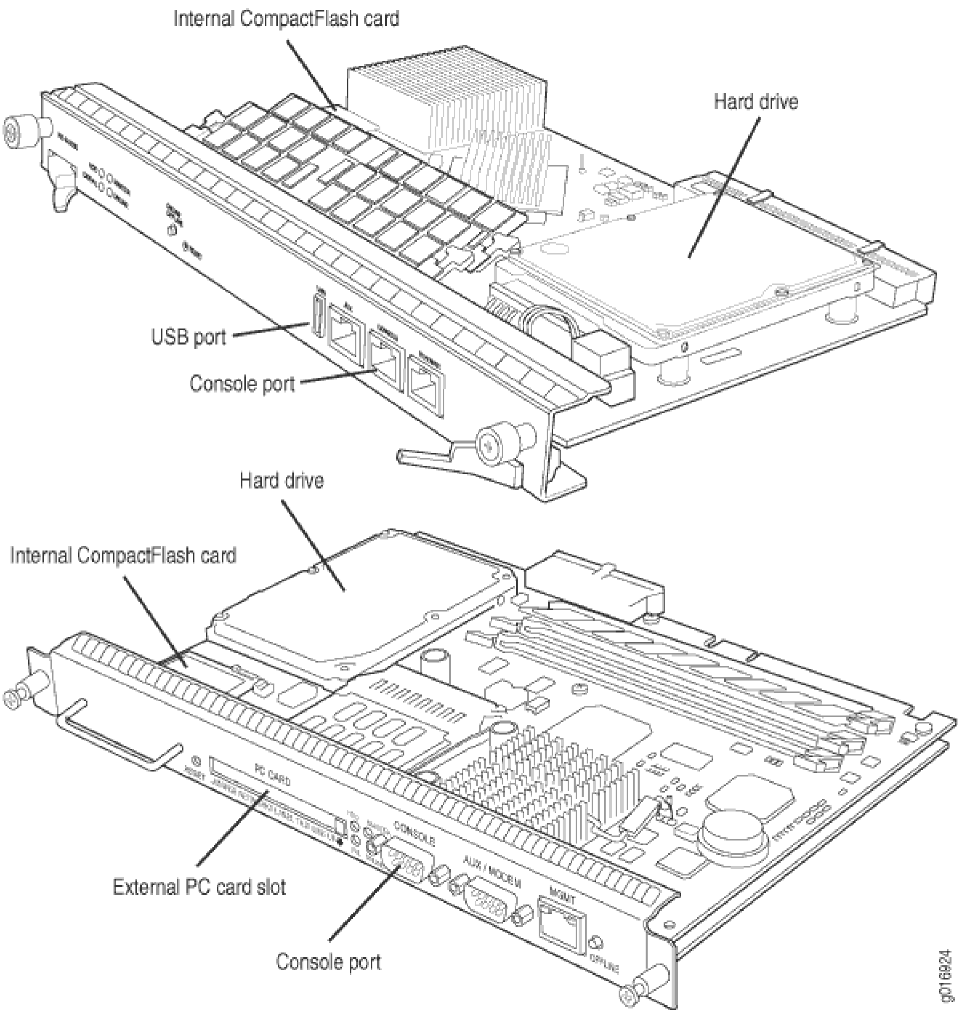
CAUTION: Repartitioning the drive re-creates the **/config** and **/var** directories in the router file system. Although the contents of **/config** and **/var/db** are preserved, the remaining contents of **/var** are lost. For this reason, we recommend that you back up the **/var** directory before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

System Memory and Storage Media on Routers

IN THIS SECTION

Figure 10 on page 398 shows examples of Routing Engines.

Figure 10: 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

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 318 section.

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 25 on page 400 specifies the storage media names by Routing Engine. The storage media device names are displayed when the router boots.

Table 25: 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	Type of Junos OS	CompactFlash Card	Hard Disk	Solid-State Drive	Removable Media Emergency Boot Device
RE-400-768 (RE5)	FreeBSD 6.x	ad0	ad1	No	ad3
RE-600-2048 (RE3)	FreeBSD 6.x	ad0	ad1	No	ad3
RE-850-1536 (RE-850)	FreeBSD 6.x	ad0	ad1	No	ad3
RE-A-1000-2048 (RE-A-1000)	FreeBSD 6.x	ad0	ad2	No	da0

Table 25: 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	Type of Junos OS	CompactFlash Card	Hard Disk	Solid-State Drive	Removable Media Emergency Boot Device
RE-A-1800x2 (RE-A-1800)	FreeBSD 6.x	ad0	No	Yes SSD1: ad1 SSD2: ad2	da0
RE-S-1300-2048 (RE-S-1300)	FreeBSD 6.x	ad0	ad2	No	da0
RE-S-1800x2 RE-S-1800x4 (RE-S-1800)	FreeBSD 6.x	ad0	No	Yes SSD1: ad1 SSD2: ad2	da0
	FreeBSD 10.x/11.x				
RE-B-1800X1-4G-S	FreeBSD 6.x	ad0	No	Yes SSD1: ad1	da0
RE-1600-2048 (RE4)	FreeBSD 6.x	ad0	ad1	No	ad3 and ad4
RE-A-2000-4096 (RE-A-2000)	FreeBSD 6.x	ad0	ad2	No	da0
RE-S-2000-4096 (RE-S-2000)	FreeBSD 6.x	ad0	ad2	No	da0
RE-MX-104	FreeBSD 6.x	No	da0	No	da1 and da2
RE-DUO-C2600-16G (RE-DUO-2600)	FreeBSD 6.x	ad0	No	ad1	da0
RE-DUO-C1800-8G- (RE-DUO-1800)	FreeBSD 6.x	ad0	No	ad1	da0
RE-DUO-C1800-16G	FreeBSD 6.x	ad0	No	ad1	da0

Table 25: 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	Type of Junos OS	CompactFlash Card	Hard Disk	Solid-State Drive	Removable Media Emergency Boot Device
RE-JCS1200-1x2330	FreeBSD 6.x	da0	da1	No	da2
RE-PTX-X8-64G	FreeBSD 6.x	No	No	Yes SSD1: sda SSD2: sdb	da0
RE-S-X6-64G	FreeBSD 6.x	No	No	Yes SSD1: sda SSD2: sdb	da0
REMX2K-X8-64G	FreeBSD 6.x	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.

SEE ALSO

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

JCS1200 Routing Engine Description

System Memory and Storage Media for SRX Series Services Gateways

SRX Series Device Overview

Figure 11 on page 403 shows an example of SRX240 device.

Figure 11: SRX240 Device Front Panel

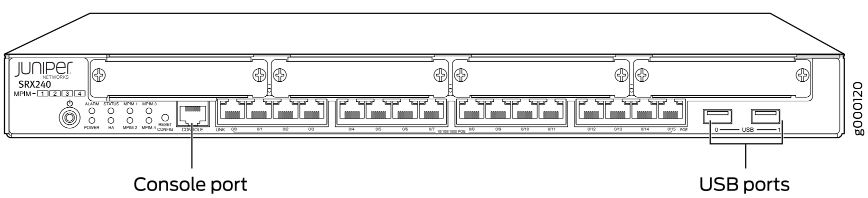


Figure 12 on page 403 shows an example of SRX650 device.

Figure 12: SRX650 Device System Routing Engine

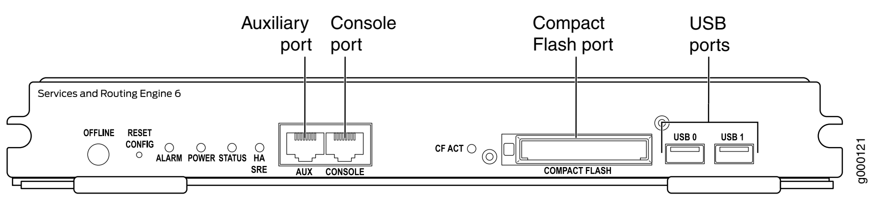


Figure 13 on page 403 shows the front panel of an SRX345 device.

Figure 13: SRX345 Device Front Panel

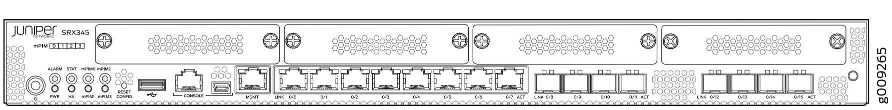


Figure 14 on page 404 shows an example of an SRX1500 device.

Figure 14: SRX1500 Device Front Panel

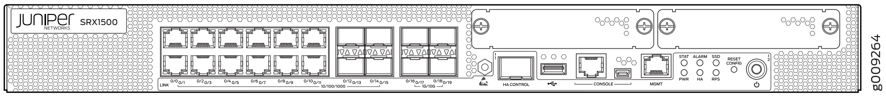


Figure 15 on page 404 shows an example of an SRX4200 device.

Figure 15: SRX4200 Services Gateway Front Panel

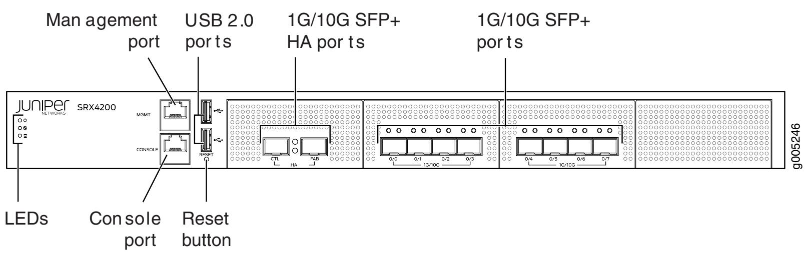


Figure 16 on page 404 shows an example of an SRX4600 device.

Figure 16: SRX4600 Services Gateway Front Panel

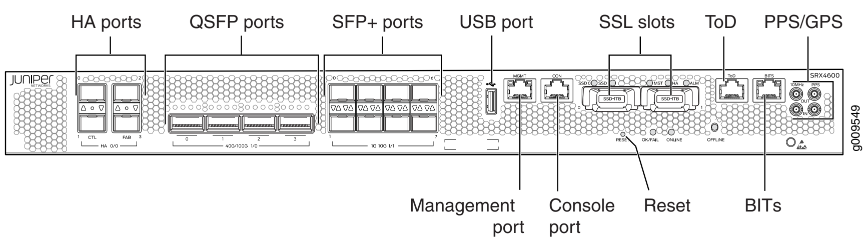
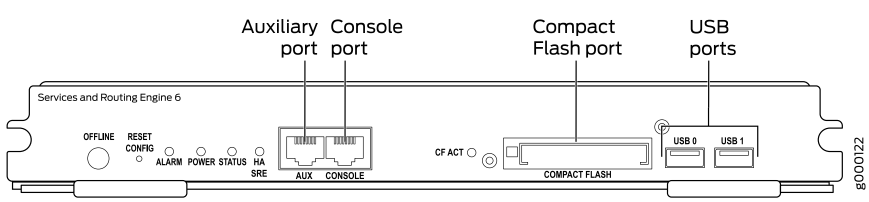


Figure 17 on page 404 shows an example of an SRX5800 device Routing Engine.

Figure 17: 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, and SRX345 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 SRX380 Services Gateway can boot from the following storage media (in the order of priority):

- Internal SSD (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

Table 26 on page 406 specifies the storage media names used by the SRX Series devices. The storage media device names are displayed as the device boots.

Table 26: 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.

SEE ALSO

Verifying PIC Combinations | 487

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.

SEE ALSO

RELATED DOCUMENTATION

8

CHAPTER

Zero Touch Provisioning

Zero Touch Provisioning

IN THIS SECTION

Zero Touch Provisioning installs or upgrades the software automatically on your new Juniper Networks devices with minimal manual intervention.

Zero Touch Provisioning Overview

IN THIS SECTION

NOTE: IPv4 is supported. IPv6 is not supported at this time.

Zero Touch Provisioning (ZTP) allows you to provision new Juniper Networks devices in your network automatically, with minimal manual intervention. You can use either management ports or network ports on your switch to connect to the network. When you physically connect a device to the network and boot it with a default factory configuration, the device upgrades (or downgrades) the Junos OS release and autoinstalls a configuration file from the network. The configuration file can be a Junos OS configuration or a script. Using scripts, you can create device-specific configuration files and perform HTTP request operations to web servers to download specific configuration files or Junos OS releases.

To locate the necessary software image and configuration files on the network, the device uses information that you have configured on a Dynamic Host Configuration Protocol (DHCP) server. If you do not configure the DHCP server to provide this information, the device boots with the preinstalled software and default factory configuration. On switches running Enhanced Layer 2 Software, Junos Extended Dynamic Host Configuration Protocol (JDHCP) is used instead of legacy DHCP. JDHCP supports the same functionality as DHCP, and all configuration options remain the same. JDHCP is an enhanced version of legacy DHCP software.

NOTE: For Junos OS Evolved, there is no JDHCP. Junos OS Evolved uses the Linux DHCP client.

Originally (as of Junos OS release 12.2), the only devices that supported ZTP (or EZ Touchless Provisioning as it was previously known) were EX Series switches and only configuration files could be used to provision configuration.

Over subsequent Junos OS releases, ZTP support has expanded:

- Starting in Junos OS Release 16.1R1, you can provision supported devices by using either a script to be executed or a configuration file to be loaded.
- Starting in Junos OS Release 17.2R1, ZTP can automate the provisioning of the device configuration and software image on VM host platforms that use PTX1000 routers.
- Starting in Junos OS Release 18.1R1, ZTP can automate the provisioning of the device configuration and software image on VM host platforms that use QFX10002-60C switches.
- Starting in Junos OS Release 18.2R1, ZTP can automate the provisioning of the device configuration and software image on VM host platforms that use PTX5000, PTX3000, PTX10008, PTX10016, PTX10002-60C routers.
- Starting in Junos OS Release 18.2R1, ZTP can automate the provisioning of the device configuration and software image on VM host platforms that use QFX10008 and QFX10016 switches.
- Starting in Junos OS Release 18.3R1, ZTP, which automates the provisioning of the device configuration and software image with minimal manual intervention, is supported on MX Series VM hosts.
- Starting in Junos OS Release 19.2R1, ZTP can automate the provisioning of the device configuration and software image on management interface emo for ACX5448 switches.

  Starting in Junos OS Evolved Release 19.1R1, ZTP can automate the provisioning of the device configuration and software image on the management interface for QFX5220 and PTX10003 devices.

The management interfaces for PTX10003 devices are vmb0 and vmb1. The management interface for QFX5220 devices is vmb0.

- Starting in Junos OS Release 19.4R1, ZTP can automate the provisioning of the device configuration and software image on Juniper Route Reflector (JRR). ZTP supports self image upgrades and automatic configuration updates using ZTP DHCP options. In this release, ZTP supports revenue ports em2 thru em9, in addition to management port em0 which is supported in Junos OS Releases before 19.4R1.

> NOTE: To see which platforms support ZTP, 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.

See the following subsections for more information on the ZTP feature:

Provisioning a Device Using a Script

During the ZTP process, when you connect and boot a new networking device, the device requests an IP address from the DHCP server. The server provides the IP address, and if configured, the filenames and locations for the software image and configuration file for the device. The configuration file can be a Junos OS configuration or a script.

If a configuration file is provided, Junos OS determines if the file is a script based on the first line of the file. If the first line contains the characters **#!** followed by an interpreter path, Junos OS treats the file as a script and executes it with the specified interpreter. If the script returns an error (that is, a nonzero value), the ZTP state machine refetches the script and attempts to execute it again. This continues until the script executes successfully.

Table 27 on page 412 outlines the supported script types, the corresponding interpreter path, and the platforms that support that script type during the ZTP process.

Table 27: Scripts Supported During ZTP

Script Type	Interpreter Path	Platform Support
Shell script	**#!/bin/sh**	All devices
SLAX script	**#!/usr/libexec/ui/cscript**	All devices
Python script	**#!/usr/bin/python**	Devices running Junos OS with Enhanced Automation Devices running Junos OS Evolved

NOTE: For security reasons, Junos OS has strict requirements for running unsigned Python scripts on devices running Junos OS. Only devices running Junos OS with Enhanced Automation and devices running Junos OS Evolved support using unsigned Python scripts in DHCP option 43 suboption 01.

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 configuration on the device.

Zero Touch Provisioning Restart Process Triggers

ZTP restarts when any of the following events occur:

- Request for configuration file, script file, or image file fails.
- Configuration file is incorrect, and commit fails.
- No configuration file and no image file is available.
- Image file is corrupted, and installation fails.
- No file server information is available.
- DHCP client does not have valid ZTP parameters configured.
- When none of the DHCP client interfaces goes to a bound state.
- ZTP transaction fails after six attempts to fetch configuration file or image file.

When any of these events occur, ZTP resets the DHCP client state machine on all of the DHCP client-configured interfaces (management and network) and then restarts the state machine. Restarting the state machine enables the DHCP client to get the latest DHCP server-configured parameters.

Before ZTP restarts, approximately 15 to 30 seconds must elapse to allow enough time to build a list of bound and unbound DHCP client interfaces.

The list of bound and unbound DHCP client interfaces can contain:

- No entries.
- Multiple DHCP client interfaces.

  Priority is given to the DHCP client interfaces that have received all ZTP parameters (software image file, configuration file, and file server information) from the DHCP server.

After the lists of bound and unbound client interfaces are created, and a DHCP client gets selected for ZTP activity, any existing default route is deleted and the DHCP client interface that was selected adds a new default route. In order to add a new default route, only one ZTP instance can be active.

After ZTP restarts, the DHCP client attempts fetching files from the DHCP server for up to six times, with ten to fifteen seconds elapsing between attempts. Every attempt, whether successful or not, is logged and can be seen on the console.

If there is a failure, or the number of attempts exceeds the limit, ZTP stops. ZTP then clears the DHCP client bindings and restarts state machine on the DHCP-configured interfaces.

The ZTP restart process continues until there is either a successful software upgrade, or an operator manually commits a user configuration and deletes the ZTP configuration.

Caveats Relating to ZTP

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 ZTP 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 "Configuring Dual-Root Partitions" on page 373.

The following are caveats for QFX Series switches:

- 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.
- QFX5200 switches only work with HTTP in 15.1X53-D30. FTP and TFTP protocols are not supported.
- 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.
- If you are performing Zero Touch Provisioning (ZTP) with a Junos OS image that contains enhanced automation for the QFX5100 switch, configure root authentication, and the provider name, license type, and deployment scope for Chef and Puppet at the **[edit system]** hierarchy in the configuration file that is fetched from the server:

  `{ master:0}`
  `root#` **set root-authentication (encrypted-password *password* | plain-text-password *password* | ssh-dsa *public-key* | ssh-rsa *public-key*)**
  `root#` **set extensions providers juniper license-type customer deployment-scope commercial**
  `root#` **set extensions providers chef license-type customer deployment-scope commercial**

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 no caveats for Junos OS Evolved platforms.

Zero Touch Provisioning Using WAN Interfaces on PTX1000 Routers

Zero Touch Provisioning (ZTP) allows you to provision your router in your network automatically, with minimal manual intervention. Starting in Junos OS Release 19.3R1, you can use either WAN interfaces or management interfaces, to automatically download and install the appropriate software and the configuration file on your router during the ZTP bootstrap process.

When you connect the router to the network at the first time, you can choose any available WAN port on the router to connect the optics. The ZTP automatically configures WAN interfaces based on the optics type, and then connects your device to the Dynamic Host Configuration Protocol (DHCP) server to perform the bootstrap process.

The WAN interfaces created based on the optics type you connected to the device and the WAN interface speed auto-transitions through all possible supported port speeds until the ZTP gets completed successfully. The speed auto-transition ensures to establish physical link of the WAN port with the optics you connected and the peer end device connectivity to the DHCP server.

PTX1000 Port Mapping shows the available combinations for the ports on the PTX1000 routers.

Configuring Zero Touch Provisioning

Configuring Zero Touch Provisioning (ZTP) allows for automatic provisioning of Juniper Network devices that you add to your network. You can provision any supported device by using either a script to be executed or a configuration file to be loaded.

To use ZTP, you configure a DHCP server to provide the required information. 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 device, issue the **request system zeroize** command on the device you want to provision.

> **NOTE:** The **request system zeroize** command is not supported on PTX1000, PTX10001-20C, QFX10002-60C, PTX10002-60C devices. You must issue the **request vmhost zeroize** command (instead of **request system zeroize**) for factory default configuration on PTX1000 routers.

> **NOTE:** On PTX10001-20C devices, after you issue the the **request vmhost zeroize** command, you will see the following message twice: **VMHost Zeroization : Erase all data, including configuration and log files ? [yes,no] (no) yes**
> **warning: Vmhost will reboot and may not boot without configuration**
> **Erase all data, including configuration and log files? [yes,no] (no) yes**

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.

    CAUTION: HTTP URLs are limited to 256 characters in length.
  - (Does not apply to Junos OS Evolved) A Domain Name System (DNS) server to perform reverse DNS lookup (not supported).
  - (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: 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. Boot the device.

2. 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.

Starting in Junos OS Evolved Release 19.3R1, on QFX5220-128C device, in Zero Touch Provisioning (ZTP), you can use either WAN interfaces or management interfaces, to automatically download and install the appropriate software and the configuration file on your device during the bootstrap process. ZTP automatically configures on WAN port that has default port speed of 100-Gbps, and then connects your device to the Dynamic Host Configuration Protocol (DHCP) server to perform the bootstrap process:

- If multiple DHCP replies arrive, the ZTP choose the best set of arguments.
- If multiple interfaces provide the same arguments, ZTP choose one of the equal interfaces.
- If there is an error while connecting to DHCP server, ZTP retry to connect DHCP server, and if multiple interfaces again provide same arguments, ZTP choose one of the interfaces.

We recommend you to provision the DHCP server, which specifies the best set of arguments, and save the software and configuration file in the specified DHCP server path on the file server.

3. 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.

   > **NOTE:** If you are performing zero touch provisioning with a Junos OS image that contains enhanced automation for the QFX5100 device, configure root authentication and the provider name, license type, and deployment scope for Chef and Puppet at the **[edit system]** hierarchy in the configuration file that is fetched from the server:
   >
   > `{ master:0}`
   >
   > `root#` **set root-authentication (encrypted-password *password* | plain-text-password *password* | ssh-dsa *public-key* | ssh-rsa *public-key*)**
   >
   > `root#` **set extensions providers juniper license-type customer deployment-scope commercial**
   >
   > `root#` **set extensions providers chef license-type customer deployment-scope commercial**

4. 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.

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

   Here is an example of an ISC DHCP 4.2 server dhcpd.conf file:

   ```
   option space NEW_OP; option;
     option NEW_OP.image-file-name code 0 = text;
     option NEW_OP.config-file-name code 1 = text;
     option NEW_OP.image-file-type code 2 = text;
     option NEW_OP.transfer-mode code 3 = text;
     option NEW_OP.alt-image-file-name code 4= text;
     option NEW_OP.http-port code 5= text;
     option NEW_OP-encapsulation code 43 = encapsulate NEW_OP;
   ```

   NOTE: Starting in Junos OS Release 18.2R1, a new DHCP option is introduced to set the timeout value for the file downloads over FTP. If the **transfer-mode** is set as FTP, the default value for the timeout is automatically set as 120 minutes, that is, in case the FTP session gets interrupted due to loss of connectivity in the middle of a file transfer, it will timeout after 120 minutes and ZTP will attempt to retry the file fetching process. This value can be overridden using the DHCP option as follows:

   ```
   option NEW_OP.ftp-timeout code 7 = text;
   option NEW_OP.ftp-timeout “
   val”;
   ```

   where **“val”** is the user configurable timeout value in seconds and must be provided within quotes (like, "val”).

6. Configure the following DHCP option 43 suboptions:

   NOTE: DHCP option 43 suboptions 05 through 255 are reserved.

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

NOTE: When the DHCP server cannot use suboption 00, configure the software image filename 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";
```

NOTE: ZTP determines if the file is a script file based on the first line of the file. If the first line contains the characters **#!** followed by an interpreter path, ZTP treats the file as a script and executes it with the specified interpreter path. In order for a script to execute, the script file must provide the ability to fetch and load a valid configuration file on the device during the ZTP process.

The following list provides the types of scripts and their associated interpreter paths:

- Shell script interpreter path: **#!/bin/sh**
- SLAX script interpreter path: **#!/usr/libexec/ui/cscript**
- Python script interpreter path: **#!/usr/bin/python**

  For security reasons, Junos OS has strict requirements for running unsigned Python scripts on devices running Junos OS. Only devices running Junos OS with Enhanced Automation and devices running Junos OS Evolved support running unsigned Python scripts as part of the ZTP process.

If the file does not contain special characters (**#!**) , ZTP determines that the file is a configuration file and loads the configuration file.

- 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 ZTP 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";
```

- Suboption 05: The HTTP port that the device uses to download either the image or configuration file or both instead of the default HTTP port.

```
option NEW_OP.http-port code 5= 8080;
```

7. (Mandatory) Configure either option 150 or option 66.

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;
```

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

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

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";
   option NEW_OP.jloader-file "jloader-qfx-5-14.1X53-D26-signed.tgz";
}
```

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.

> **NOTE:** When SLAX scripts are executed, the **op-script.log** and **event-script.log** files are produced.

- /var/log/dhcp_logfile
- /var/log/event-script.log
- /var/log/image_load_log
- /var/log/messages
- /var/log/op-script.log
- /var/log/script_output

You can also monitor the ZTP process by looking at error messages and issuing operational commands. See "Monitoring Zero Touch Provisioning" on page 426 for more information.

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, SRX380, 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, SRX380, 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:

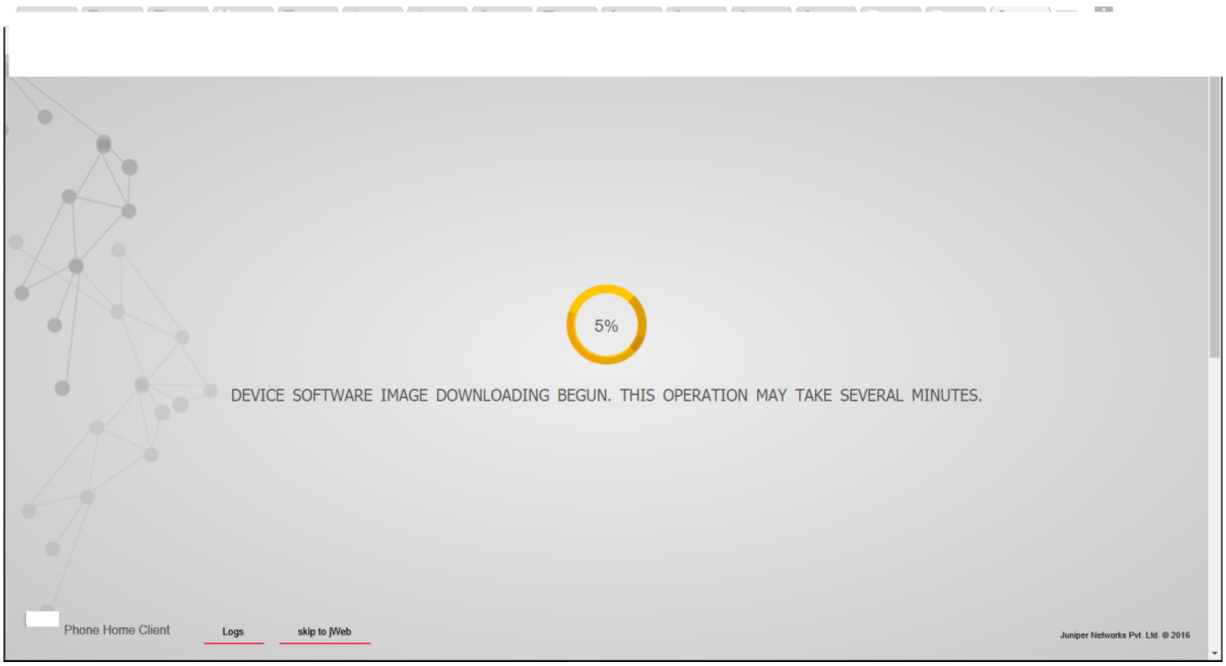
1. Connect a management device (PC or laptop) to any front panel Ethernet port (WAN port) of the SRX Series device.

2. Launch a Web browser from the management device and enter the authentication code in the webpage as shown in Figure 18 on page 423.

Figure 18: 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 19 on page 424.

Figure 19: Initiating ZTP Process (Software Image Downloading)

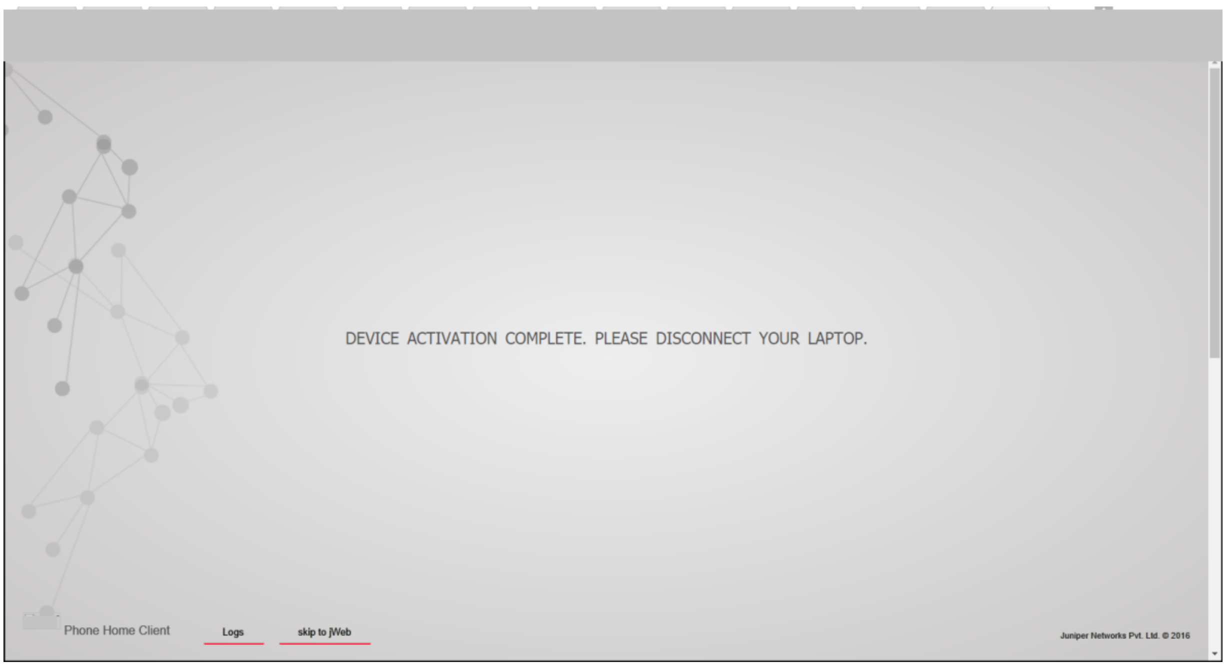


At this step:

- 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 pushes 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 20 on page 425.

Figure 20: 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.

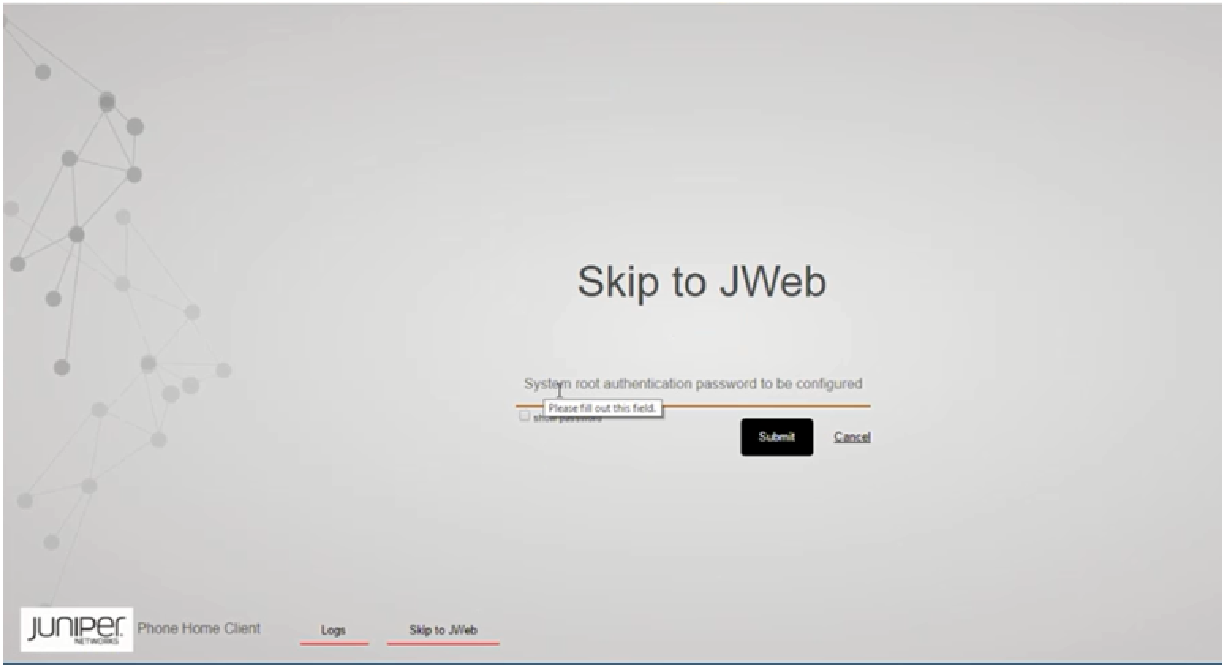
> **NOTE:** In case of failure at any stage, the procedure is started all over again.

> **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, SRX380, 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 21 on page 426.

Figure 21: 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.

Monitoring Zero Touch Provisioning

Starting in Junos OS Release 12.2, you can use the console and operational commands to monitor Zero Touch Provisioning.

Starting in Junos OS Evolved Release 19.1R1, to monitor zero touch provisioning on Junos OS Evolved, use the **show system ztp** command.

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.
- Filenames 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
19.4R1	Starting in Junos OS Release 19.4R1, ZTP can automate the provisioning of the device configuration and software image on Juniper Route Reflector (JRR). ZTP supports self image upgrades and automatic configuration updates using ZTP DHCP options. In this release, ZTP supports revenue ports em2 thru em9, in addition to management port em0 which is supported in Junos OS Releases before 19.4R1.
19.3R1-Evo	Starting in Junos OS Evolved Release 19.3R1, on QFX5220-128C device, in Zero Touch Provisioning (ZTP), you can use either WAN interfaces or management interfaces, to automatically download and install the appropriate software and the configuration file on your device during the bootstrap process.
19.3R1	Starting in Junos OS Release 19.3R1, you can use either WAN interfaces or management interfaces, to automatically download and install the appropriate software and the configuration file on your router during the ZTP bootstrap process.
19.2R1	Starting in Junos OS Release 19.2R1, ZTP can automate the provisioning of the device configuration and software image on management interface emo for ACX5448 switches.
19.1-Evo	Starting in Junos OS Evolved Release 19.1R1, ZTP can automate the provisioning of the device configuration and software image on the management interface for QFX5220 and PTX10003 devices.
19.1-Evo	Starting in Junos OS Evolved Release 19.1R1, to monitor zero touch provisioning on Junos OS Evolved, use the **show system ztp** command.
18.3R1	Starting in Junos OS Release 18.3R1, ZTP, which automates the provisioning of the device configuration and software image with minimal manual intervention, is supported on MX Series VM hosts.
18.2R1	Starting in Junos OS Release 18.2R1, ZTP can automate the provisioning of the device configuration and software image on VM host platforms that use PTX5000, PTX3000, PTX10008, PTX10016, PTX10002-60C routers.
18.2R1	Starting in Junos OS Release 18.2R1, ZTP can automate the provisioning of the device configuration and software image on VM host platforms that use QFX10008 and QFX10016 switches.
18.1R1	Starting in Junos OS Release 18.1R1, ZTP can automate the provisioning of the device configuration and software image on VM host platforms that use QFX10002-60C switches.
17.2R1	Starting in Junos OS Release 17.2R1, ZTP can automate the provisioning of the device configuration and software image on VM host platforms that use PTX1000 routers.

16.1R1	Starting in Junos OS Release 16.1R1, you can provision supported devices by using either a script to be executed or a configuration file to be loaded
12.2	Starting in Junos OS Release 12.2, you can use the console and operational commands to monitor Zero Touch Provisioning.

9

CHAPTER

Phone-Home Client

Obtaining Configurations and Software Image Without User Intervention Using Phone-Home Client

IN THIS SECTION

The phone-home client (PHC) enables the device to securely obtain bootstrapping data, such as a configuration or software image, with no user intervention other than having to physically connect the device to the network.

Prerequisites

PHC depends on the following software and utilities to operate:

- Connectivity to redirect server and phone-home server (PHS)
- DHCP client
- SLAX support for configuration commits
- Python support
- Curl support
- Factory default configuration
- Mechanism to retrieve device serial number
- SHA1/MD5 utilities to verify software image
- Basic utilities like GREP and AWK

Understanding the Phone-Home Client

PHC enables the device to securely obtain bootstrapping data, such as a configuration or software image, with no user intervention other than having to physically connect the device to the network. When the device first boots, PHC connects to a redirect server, which then redirects to PHS to get the configuration or software image.

Similar to DHCP-based ZTP, the device must be in factory default state in order for PHC to provision the device. If the device is not in factory default state, you can issue the **request system zeroize** command to bring the device back to the factory default state.

Understanding the Redirect Server Configuration

By default, the factory default configuration includes the redirect server URL, which is https://redirect.juniper.net.

Understanding Interoperability Between the Phone-Home Client and DHCP-Based ZTP

Both PHC and the DHCP-based ZTP methods are available. To avoid conflicts between these two provisioning methods, the following steps are taken when the device boots up:

> **NOTE:** Provisioning does not start if the device is not in factory default mode. If the device is not in factory default mode, issue the **request system zeroize** command.

1. If the DHCP client receives either partial or complete DHCP options, PHC is aborted, and DHCP-based ZTP attempts to provision the device until it is successful.
2. If the DHCP client does not receive DHCP options, PHC attempts to provision the device until it is successful.

   If PHC fails to connect to the redirect server, however, DHCP-based ZTP attempts to provision the device. Both provisioning methods attempt to provision the device until one method is successful.

Understanding the Phone-Home Client Process

The following steps take place when PHC is launched:

1. PHC connects to the redirect server.
2. The device downloads and installs the software image from PHS.

   If the software upgrade fails, the process starts over.
3. The device reboots, and PHC validates the installed software image when the device comes back online.
4. The device downloads the configuration.
5. If a script (either pre-configuration script, post-configuration scripts, or both) was received as part of the configuration, the following happens:

   > **NOTE:** PHC supports both Python and shell scripts.

   a. The pre-configuration script is executed.

   b. The configuration received from the redirect server is committed.

   c. The post-configuration script is executed.
6. PHC sends a bootstrap-complete message to the PHS.
7. PHC cleans up the downloaded resources.
8. The phone-home configuration, along with any supporting configuration, is deleted from the device.
9. If any of the above steps fail, the phone-home process starts over again from the beginning, and a bootstrap failure error message is sent to PHS

Understanding the Configuration File Format for the Phone-Home Client

PHC supports XML as the file format for the configuration file.

For example, the configuration file format looks like this:

```
<configuration>
[ Configuration in XML format ]
</configuration>
```

Currently, only the **merge** and **override** CLI commands are supported on configurations received by the PHC.

Understanding Pre-Configuration and Post-Configuration Scripts

You can include pre-configuration and post-configuration scripts on PHS in addition to, or instead of, using the Junos OS CLI. Embed the scripts in base64 encoded format. PHC extracts the encoded scripts from the bootstrap information received from PHS, decodes, and then runs the decoded scripts at the appropriate stages of provisioning.

Verifying that the Phone-Home Client Downloaded the Configuration and Software Image

To verify the progress of the phone-home process, you can view the **notification.xml** file on PHS.

10

CHAPTER

Automatic Installation of Configuration Files

Understanding Autoinstallation of Configuration Files

IN THIS SECTION

Autoinstallation is an automated process and does not require any specific configuration on a device. To simplify the process, you can specify one or more interfaces, protocols, and configuration servers to be used for autoinstallation.

Autoinstallation Overview

IN THIS SECTION

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 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 28 on page 442 lists the protocols that the device can use on these interfaces for IP address acquisition.

Table 28: Interfaces and Protocols for IP Address Acquisition During Autoinstallation

Interface and Encapsulation Type	Protocol for Autoinstallation
Ethernet LAN interface with High-Level Data Link Control (HDLC)	DHCP, BOOTP, or Reverse Address Resolution Protocol (RARP)
Serial WAN interface with HDLC	Serial Line Address Resolution Protocol (SLARP)
Serial WAN interface with Frame Relay	BOOTP

If the server with the autoinstallation configuration file is not on the same LAN segment as the new device, or if a specific device is required by the network, you must configure an intermediate device directly attached to the new device through which the new device can send Trivial File Transfer Protocol (TFTP), BOOTP, and Domain Name System (DNS) requests. In this case, you specify the IP address of the intermediate device as the location to receive TFTP requests for autoinstallation.

Typical Autoinstallation Process on a New Device

When a device is powered on for the first time, it performs the following autoinstallation tasks:

1. The new device sends out DHCP, BOOTP, RARP, or SLARP requests on each connected interface simultaneously to obtain an IP address.

   If a DHCP server responds, it provides the device with some or all of the following information:

   - An IP address and subnet mask for the autoinstallation interface.
   - The location of the TFTP (typically), Hypertext Transfer Protocol (HTTP), or FTP server on which the configuration file is stored.
   - The name of the configuration file to be requested from the TFTP server.
   - The IP address or hostname of the TFTP server.

     If the DHCP server provides only the hostname, a DNS server must be available on the network to resolve the name to an IP address.
   - The IP address of an intermediate device if the configuration server is on a different LAN segment from the new device.

2. After the new device acquires an IP address, the autoinstallation process on the device attempts to download a configuration file in the following ways:

   a. If the DHCP server specifies the host-specific configuration file (boot file) ***hostname*.conf**, the device uses that filename in the TFTP server request. (In the filename, ***hostname*** is the hostname of the new device.) The autoinstallation process on the new device makes three unicast TFTP requests for ***hostname*.conf**. If these attempts fail, the device broadcasts three requests to any available TFTP server for the file.

   b. If the new device cannot locate ***hostname*.conf**, the autoinstallation process unicasts or broadcasts TFTP requests for a default device configuration file called **network.conf**, which contains hostname-to-IP address mapping information, to attempt to find its hostname.

   c. If **network.conf** contains no hostname entry for the new device, the autoinstallation process sends out a DNS request and attempts to resolve the new device's IP address to a hostname.

   d. If the new device can determine its hostname, it sends a TFTP request for the ***hostname*.conf** file.

   e. If the new device is unable to map its IP address to a hostname, it sends TFTP requests for the default configuration file **router.conf**.

3. After the new device locates a configuration file on a TFTP server, autoinstallation downloads the file, installs the file on the device, and commits the configuration.

NOTE:

- If you configure the DHCP server to provide only the TFTP server hostname, add an IP address-to-hostname mapping entry for the TFTP server to the DNS database file on the DNS server in the network.
- If the new device is not on the same network segment as the DHCP server (or other device providing IP address resolution), configure an existing device as an intermediate to receive TFTP and DNS requests and forward them to the TFTP server and the DNS server. You must configure the LAN or serial interface on the intermediate device with the IP addresses of the hosts providing TFTP and DNS service. Connect this interface to the new device.

NOTE: Starting in Junos OS Release 15.1X49-D60 and in Junos OS Release 17.3R1, on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices, some of the factory-default configurations are changed.

- The **name-server** statement, used to configure one or more Domain Name System (DNS) name servers, is changed to 8.8.8.8 and 8.8.8.4. Previously, it was 208.67.222.222 and 208.67.220.220.
- A new system service, NETCONF service over SSH, is introduced at the **[edit system services]** hierarchy:

```
edit system services netconf ssh
```

- The following configuration setting for HTTPS (secure management) access using the J-Web interface is changed. Now, there is no need to specify the interface details for J-Web management. With this configuration, you can manage the device from any interface through HTTPS.

```
edit system services web-management https interface [irb.0]
```

- A license autoupdate URL (**https://ae1.juniper.net/junos/key_retrieval**) is now supported under the **[edit system]** hierarchy:

```
license {
    autoupdate {
        url https://ae1.juniper.net/junos/key_retrieval;
    }
}
```

- A new system log configuration is introduced to configure system log messages to record all commands entered by users and all authentication or authorization attempts under the **[edit system]** hierarchy:

```
syslog {
    archive size 100k files 3;
    user * {
        any emergency;
    }
    file messages {
        any notice;
        authorization info;
```

```
    }
    file interactive-commands {
        interactive-commands any;
    }
}
```

The above factory-default configurations are also applicable on SRX380 Series devices.

NOTE: On SRX300, SRX320, SRX340, SRX345, SRX380, 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.

Layer 2 Ethernet switching is not supported on the same interface for SRX300, SRX320, SRX340, SRX345, SRX380, 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.

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.

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.

NOTE: Please refer to the product Data Sheets for details, or contact your Juniper Account Team or Juniper Partner. Please refer to the Juniper Licensing Guide for general information about License Management.

Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group

IN THIS SECTION

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

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.

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 Metro 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).

SEE ALSO

autoinstallation | **498**
delete-after-commit (JNU Satellites) | **506**

Configuring Autoinstallation of Configuration Files

IN THIS SECTION

Autoinstallation is an automated process and does not require any specific configuration on a device. To simplify the process, you can specify one or more interfaces, protocols, and configuration servers to be used for autoinstallation.

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 device 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 device, 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 device.
- Configure a DHCP server on your network to meet your network requirements. You can configure a switch to operate as a DHCP server.

- 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 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 **device.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 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 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 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 device.
- 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 device. Each device 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 device, the device 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 device to a DNS database file.

    The device 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@host# 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@host# set autoinstallation interfaces ge-0/0/0 bootp
   ```

To verify autoinstallation, from the CLI, enter the **show system autoinstallation status** command.

```
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: device-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
```

Example: Configuring Autoinstallation on SRX Series Devices

IN THIS SECTION

This example shows how to configure a device for autoinstallation.

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:
  - 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 HLDC 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

IN THIS SECTION

Confirm that the configuration is working properly.

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.

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.

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.

SEE ALSO

show system autoinstallation status | 718
Installing Software on EX Series Switches | 163

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 Metro 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: retype 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";

    }
}
```

SEE ALSO

autoinstallation | 498
delete-after-commit (JNU Satellites) | 506

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.

SEE ALSO

autoinstallation | 498

delete-after-commit (JNU Satellites) | 506

show system autoinstallation status | 718

Configuring Autoinstallation of Configuration Files on ACX Series

IN THIS SECTION

Autoinstallation is an automated process and does not require any specific configuration on a device. To simplify the process, you can specify one or more interfaces, protocols, and configuration servers to be used for autoinstallation.

ACX Series Autoinstallation Overview

IN THIS SECTION

Autoinstallation provides automatic configuration for a new router that you connect to the network and turn on, or for a router configured for autoinstallation. The autoinstallation process begins anytime a router is powered on and cannot locate a valid configuration file in the CompactFlash (CF) 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 CF 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 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

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 ACX Series 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), 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 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, autoinstallation downloads the file, installs the file on the router, and commits the configuration.

Before You Begin Autoinstallation on an ACX Series Universal Metro Router

To configure a router for autoinstallation, complete the following tasks:

- Make sure you have a DHCP server on your network to meet your network requirements.
- Create one of the following configuration files and store it on an HTTP, FTP, or TFTP server in the network:
  - A host-specific file with the name ***hostname*.conf** for each router undergoing autoinstallation. Replace ***hostname*** with the name of a router. The ***hostname*.conf** file typically contains all the configuration information necessary for the router with this hostname.
  - A default configuration file named **router.conf** with the minimum configuration necessary to enable you to telnet into the new router for further configuration.
- Physically attach the router to the network using a Gigabit Ethernet interface.
- If you configure the DHCP server to provide only the HTTP, FTP, or TFTP server hostname, add an IP address-to-hostname mapping entry for the HTTP, FTP, or TFTP server to the DNS database file on the DNS server in the network.
- If the new router is not on the same network segment as the DHCP server (or other router providing IP address resolution), configure an existing router as an intermediate to receive HTTP, FTP, or TFTP and DNS requests and forward them to the HTTP, FTP, or TFTP and DNS servers. You must configure the LAN on the intermediate router with the IP addresses of the hosts providing HTTP, FTP, or TFTP and DNS service. Connect this interface to the new router.
- If you are using ***hostname*.conf** files for autoinstallation of host-specific configuration files, you must also complete the following tasks:
  - Configure the DHCP server to provide a ***hostname*.conf** filename to each new router. Each router 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 router, the router 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 new router to a DNS database file.

The router uses the hostname to request a ***hostname*.conf** file from the server.

Autoinstallation Configuration of ACX Series Universal Metro Routers

No configuration is required on a router 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.

To configure autoinstallation:

1. 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**.

2. 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
   ```

Verifying Autoinstallation on ACX Series Universal Metro Routers

Purpose

After you have configured autoinstallation, display the status of autoinstallation on an ACX Series router.

Action

From the CLI, enter the **show system autoinstallation status** command.

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.

SEE ALSO

autoinstallation

show system autoinstallation status | 718

USB Autoinstallation on ACX Series Routers

If you have a new ACX Series router, you can use a Disk-on-Key USB memory stick (“USB key”) to configure the router.

This configuration method has the following requirements:

- A management device (PC or laptop).
- A Disk-on-Key device with one of the following 16-bit or 32-bit file allocation table (FAT) file systems:
  - DOS 3.0+ 16-bit FAT (up to 32 MB)
  - DOS 3.31+ 16-bit FAT (over 32 MB)
  - FAT32
  - FAT32, LBA-mapped
  - 16-bit FAT, LBA-mapped
- An ACX Series router with the factory configuration. If other Junos OS configuration files exist on the router, the router cannot read the **juniper-config.txt** file from the Disk-on-Key device.

  > **NOTE:** The USB-based autoinstallation process overrides the network-based autoinstallation process. If the ACX Series router detects a USB Disk-on-Key device containing a valid configuration file during autoinstallation, it configures the router using the configuration file on Disk-on-Key instead of fetching the configuration from the network.

To configure an ACX Series router using Disk-on-Key:

1. Using a text editor on a PC or laptop, create the configuration file, named *juniper-config.txt*, as a sequence of configuration commands (“set” commands). To reuse configuration from another ACX Series router, the configuration can be saved in configuration mode as a sequence of configuration commands on the router using the **“show | display set | save <filename>”** command and then copying the <filename> to the PC or router as *juniper-config.txt*.

2. Copy the *juniper-config.txt* file to a Disk-on-Key device.

3. Plug the Disk-on-Key device into the USB port on the new ACX Series router.

4. Power on the router by pressing the POWER button on the front panel. Wait for the router to start and access the Disk-on-Key device (observe the LEDs on the Disk-on-Key device).

   The router reads the *juniper-config.txt* file from the Disk-on-Key device and commits the configuration.

5. Remove the Disk-on-Key device from the router.

6. The configuration of the router is complete.

Autoinstallation on ACX Series Routers in Hybrid Mode Overview

The ACX Series router has an autoinstallation mechanism that allows the router to configure itself out-of-the-box with no manual intervention, using the configuration available either on the network, locally through a removable media, or a combination of both.

Autoinstallation process delivers the following benefits:

- The router can be sent from the warehouse to the deployment site without any pre-configuration 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.

ACX Series routers support the retrieval of partial configuration from an external USB storage device plugged into the router's USB port during the autoinstallation process. This partial configuration in turn facilitates the network mode of autoinstallation to retrieve the complete configuration file from the network. This method is called hybrid mode of autoinstallation.

Autoinstallation process operates in three modes:

- USB mode—Autoinstallation obtains the required configuration from the configuration file saved in an external USB storage device plugged into the router.
- Network Mode—Autoinstallation triggers IP address acquisition mechanism (the router sends out DHCP or RARP requests on each connected interface simultaneously) to obtain an IP address. Once the router has an IP address, it sends a request to the specified configuration server and downloads and installs the configuration.
- Hybrid mode—Autoinstallation obtains partial configuration from an external USB storage device and uses that configuration to obtain the complete configuration file in network mode. This mode is a combination of USB mode and Network mode.

On the different ACX Series routers, autoinstallation is supported on the following Gigabit Ethernet (**ge**) and 10- Gigabit Ethernet (**xe**) interfaces:

- On ACX1000 routers, interfaces ge-0/1/0 through ge-0/1/7, and ge-0/2/0 through ge-0/2/3
- On ACX1100 routers, interfaces ge-0/0/0 through ge-0/0/7, and ge-0/1/0 through ge-0/1/3
- On ACX2000 routers, interfaces ge-0/1/0 through ge-0/1/7, ge-0/2/0 through ge-0/2/1, and xe-0/3/0 through xe-0/3/1

- On ACX2100 routers, interfaces ge-1/0/0 through ge-1/0/3, ge-1/1/0 through ge-1/1/3, ge-1/2/0 through ge-1/2/1, and xe-1/3/0 through xe-1/3/1
- On ACX2200 routers, interfaces ge-0/0/0 through ge-0/0/3, ge-0/1/0 through ge-0/1/3, ge-0/2/0 through ge-0/2/1, and xe-0/3/0 through xe-0/3/1
- On ACX4000 routers, interfaces ge-0/0/0 through ge-0/0/7, ge-0/1/0 through ge-0/1/1, ge-1/0/0 through ge-1/0/5, ge-1/1/0 through ge-1/1/5 , and xe-0/2/0 through xe-0/2/1

Prerequisites for Autoinstallation on ACX Series Routers in Hybrid Mode

Before you perform autoinstallation on a router in hybrid mode, complete the following tasks:

Using a text editor on a PC or laptop, create the configuration file, named *juniper-config.txt*, as a sequence of configuration commands (“set” commands). To reuse configuration from another ACX Series router, the configuration can be saved in configuration mode as a sequence of configuration commands on the router using the **“show | display set | save <filename>”** command and then copying the <filename> to the PC or router as *juniper-config.txt*.

You must copy the *juniper-config.txt* file to an external USB storage device. Plug the USB device into the USB port on the new ACX Series router. When you power on the router, the router first attempts to access the external USB storage device. The router reads the *juniper-config.txt* file from the external USB storage device and commits the configuration.

> **NOTE:** For autoinstallation process to switch to the network mode, the **continue-network-mode** statement must be present in the autoinstallation stanza at the **[edit system autoinstallation]** hierarchy level of the **juniper-config.txt** configuration file. The presence of the **continue-network-mode** statement in the **juniper-config.txt** file causes the router to consider it as a partial configuration. Otherwise, if the **continue-network-mode** statement is not present in the **juniper-config.txt** file, the router considers the configuration on the external USB storage device as the complete configuration and it will not switch to the network mode.

Perform all of the steps described in the “Before You Begin Autoinstallation on an ACX Series Universal Metro Router” on page 469 section, which prepares the router for network-based autoinstallation.

Autoinstallation Process on a New ACX Series Router in Hybrid Mode

You can perform autoinstallation on a new ACX Series router in hybrid mode, which is a combination of the USB-based autoinstallation process and the network-based autoinstallation process.

This configuration method has the following requirements:

- A management device (PC or laptop).
- An external USB storage device with one of the following 16-bit or 32-bit file allocation table (FAT) file systems:
  - DOS 3.0+ 16-bit FAT (up to 32 MB)
  - DOS 3.31+ 16-bit FAT (over 32 MB)
  - FAT32
  - FAT32, LBA-mapped
  - 16-bit FAT, LBA-mapped

BOOTP, RARP and DHCP are the supported protocols for acquisition of IP address of the router and TFTP, FTP, and HTTP are the supported protocols for downloading the configuration file from an external server URL on which the configuration file is stored.

The following operations occur during autoinstallation in hybrid mode on ACX Series routers:

1. When a new ACX Series router is powered on for the first time, the router performs the following autoinstallation tasks: The router boots the Junos OS image. The management process (mgd) is invoked and it determines whether a valid configuration exists on the router's Flash memory. If a valid configuration is not present on the router, it loads and commits the factory-default configuration.
2. If the factory-default configuration contains the **autoinstallation configuration** stanza at the **[edit system]** hierarchy level, the autoinstallation process is triggered.
3. The autoinstallation process detects whether an external USB storage device is connected to the router and examines whether the USB device contains a valid configuration file. If the USB storage device contains a configuration file named **juniper-config.txt**, the router reads the **juniper-config.txt** file and commits the configuration.
4. If the **juniper-config.txt** file on the external USB storage device contains **continue-network-mode** statement, the configuration is treated as partial configuration. The autoinstallation process uses this partial configuration to obtain the complete configuration file from a server on the network. At this stage, the router completes the USB mode of the autoinstallation procedure and switches to the network mode of the autoinstallation procedure.

   > **NOTE:** The **continue-network-mode** statement must be present in the autoinstallation stanza at the **[edit system autoinstallation]** hierarchy level of the **juniper-config.txt** file.

5. After acquiring the partial configuration from the **juniper-config.txt** file, the configuration discovery procedure is initiated. For all physical Ethernet interfaces that transition to the up state, the autoinstallation process verifies whether autoinstallation is configured on that Ethernet interface. The

autoinstallation process starts IP address acquisition mechanism to obtain IP address of the server followed by the configuration file retrieval mechanism.

6. For the interfaces that take part in the autoinstallation process, the IPv4 address discovery procedure is triggered. The new ACX Series router sends out DHCP, or BOOTP, or RARP requests on each connected interface simultaneously to obtain an IP address. The interfaces statement in the **autoinstallation configuration** stanza at the **[edit system]** hierarchy level in the factory-default configuration also specify the protocols to be used for IPv4 address discovery. If the interfaces statement is not configured, all the applicable protocols for an interface are used to send out requests on each connected Ethernet interface.

7. If an IPv4 address cannot be retrieved, the autoinstallation process starts the DHCP server on all participating interfaces (assigns static IP address in the form of 192.168.x.1 to allow a management station to connect to the router for manual configuration) and terminates the autoinstallation procedure.

8. 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 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 configuration server 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.

   NOTE: To use HTTP or FTP server, you need to specify the URL of the configuration server under the **[edit system autoinstallation configuration-servers]** hierarchy level.

9. After an IPv4 address is retrieved for an interface, the interface is configured with that address and the autoinstallation process starts the configuration file discovery procedure. The autoinstallation process on the router attempts to download a configuration file in the following methods:

   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 (either through file field option or boot file option or host name), the router uses that filename in the TFTP server request. In case of host name, the configuration filename is hostname.conf. The autoinstallation process on the new router makes unicast TFTP request for hostname.conf. If this attempt fails, the router broadcasts the request to any available TFTP server for the configuration file.

c. If the new router is unable locate the configuration file, 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.

> **NOTE:** The autoinstallation process makes a maximum of three attempts to retrieve the configuration file by repeating the methods listed above (b to f). In case the autoinstallation process fails to retrieve the configuration file after three attempts, the autoinstallation process goes to start state.

g. After the new router locates a configuration file on a TFTP server, autoinstallation downloads the file, installs the file on the router, and commits the configuration.

Configuring Autoinstallation of ACX Series Routers in Hybrid Mode

To configure the router for autoinstallation in hybrid mode, perform the following tasks:

Create a configuration file as *juniper-config.txt*.

1. Using a text editor on a PC or laptop, create the configuration file, named *juniper-config.txt*. This configuration file must contain a sequence of configuration commands ("set" commands).

   > **NOTE:** To reuse a configuration from another ACX Series router, save the configuration in configuration mode as a sequence of configuration commands on the router using the **"show | display set | save <filename>"** command and then copying the <filename> to the PC or router as *juniper-config.txt*.

2. Include the **continue-network-mode** statement at the **[edit system autoinstallation]** hierarchy level in the *juniper-config.txt* configuration file. The presence of the **continue-network-mode** statement causes the router to consider it as a partial configuration and the autoinstallation process switches to network mode to retrieve the complete configuration from a network server.

```
[edit system]
user@host# set autoinstallation continue-network-mode
```

3. Specify the URL address of one or more network servers from which to obtain the complete configuration.

```
[edit system]
user@host# set autoinstallation configuration-servers
tftp://username:password@tftpconfig.sp.com/filename.conf
```

> **NOTE:** You can also use an HTTP or FTP address—for example, **http://*user:password*@httpconfig.sp.com/filename.conf** or **ftp://*user:password*@sftpconfig.sp.com/filename.conf**.

4. Specify the root authentication password.

```
[edit system]
user@host# set root-authentication encrypted-password “password”;
```

5. 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
```

> **NOTE:** Configuring an interface is optional. If an interface is configured, then autoinstallation process is triggered on the configured interface only. If an interface is not configured, then autoinstallation process is triggered on all the interfaces that are physically in link up state.

6. Copy the *juniper-config.txt* file to an external USB storage device.

7. Plug the external USB storage device to the router’s USB port.

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.

```
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
```

SEE ALSO

autoinstallation

show system autoinstallation status | 718

11

CHAPTER

Troubleshooting Software Installation

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

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 180 to get an image on the system and boot the switch.

To perform a recovery installation:

1. Power on the switch.

   The loader script starts.

   After the message **Loading /boot/defaults/loader.conf** displays, you are prompted with the following:

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

2. Press the space bar to enter the manual loader.

   The **loader>** prompt displays.

3. Enter the following command:

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

where:

- **format**—Use this option to wipe the installation media before installing the software package. If you do not include this option, the system installs the new Junos OS package in a different partition from the partition used by the most recently installed Junos OS package.
- **external**—Use this option to install the software package on an external medium.
- ***source***—Represents the name and location of the Junos OS package either on a server on the network or as a file on the USB flash drive:
  - Network address of the server and the path on the server; for example, **tftp://192.168.1.28/junos/jinstall-ex-4200-9.4R1.5-domestic-signed.tgz**
  - The Junos OS package on a USB device is commonly stored in the root drive as the only file; for example, **file:///jinstall-ex-4200-9.4R1.5-domestic-signed.tgz**

The boot process proceeds as normal and ends with a login prompt.

Rebooting from the Inactive Partition

Problem

Description: EX Series switches shipped with Junos OS Release 10.4R2 or earlier have Junos OS loaded on the system disk in partition 1. The first time you upgrade, the new software package is installed in partition 2. When you finish the installation and reboot, partition 2 becomes the active partition. Similarly, subsequent software packages are installed in the inactive partition, which becomes the active partition when you reboot at the end of the installation process.

On switches shipped with Release 10.4R3 and later, the same Junos OS image is loaded in each of the two root partitions, and you should copy the new software image to the alternate partition each time you upgrade.

If you performed an upgrade and rebooted, the system resets the active partition. You can use this procedure to manually boot from the inactive partition.

> **NOTE:** If you have completed the installation of the software image but have not yet rebooted, issue the **request system software rollback** command to return to the original software installation package.

Solution

Reboot from the inactive partition:

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

NOTE: If you cannot access the CLI, you can reboot from the inactive partition using the following procedure from the loader script prompt:

1. Unload and clear the interrupted boot from the active partition:

   ```
   loader> unload
   loader> unset vfs.root.mountfrom
   ```

2. Select the new (inactive) partition to boot from:

   ```
   loader> set currdev=diskxsy:
   ```

   where *x* is either 0 (internal) or 1 (external), and the y indicates the number of the inactive partition, either 1 or 2.

   You must include the colon (:) at the end of this command.

3. Boot Junos OS from the inactive partition:

   ```
   loader> boot
   ```

Freeing Disk Space for Software Installation

Problem

Description: The software installation process requires a certain amount of unused disk space. If there is not enough space, you might receive an error message such as:

```
fetch: /var/tmp/incoming-package.tgz: No space left on device
```

Solution

Identify and delete unnecessary files by using the request system storage cleanup command.

Installation from the Boot Loader Generates ’cannot open package’ Error

Problem

Description: When installing a Junos OS software image from the loader prompt, a “cannot open package error” is generated:

```
loader> install - -format
```

```
tftp://10.204.33.248/images/Flash_corr/official/jinstall-ex-4200-10.4I2011012-domestic-signed.tgz
Speed: 1000, full duplex
bootp: no reply
No response for RARP request
net_open: RARP failed
cannot open package (error 5)
```

Solution

This might be due to the IP address, gateway IP address, netmask address, or server IP address not being properly set. You can set these values either from the shell or from the u-boot prompt.

To set these values from the shell:

% **nvram setenv ipaddr 10.204.35.235**

% **nvram setenv netmask 255.255.240.0**

% **nvram setenv gatewayip 10.204.47.254**

% **nvram setenv serverip 10.204.33.248**

To set these values from the u-boot prompt, log in to a console connection, reboot, and stop at the u-boot prompt (Cntrl+c):

```
=> setenv ipaddr 10.204.35.235
=> setenv gatewayip 10.204.47.254
=> setenv serverip 10.204.33.248
=> setenv netmask 255.255.240.0
=> saveenv
=> printenv Verify whether variables are set properly or not
=> boot
```

RELATED DOCUMENTATION

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 | 135
Troubleshooting Software Installation | 481
show system storage partitions | 772

Managing Disk Space 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

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

System Memory and Storage Media for SRX Series Services Gateways | **403**

Storage Media Names for SRX Series Devices

12
CHAPTER

Configuration Statements

auto-configuration

Syntax

```
auto-configuration {
    command binary-file-path;
    disable;
}
```

Hierarchy Level

```
[edit system processes]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Configure the autoconfiguration process.

Options

- **command *binary-file-path***—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

Autoinstallation Overview | 440
Example: Configuring Autoinstallation on SRX Series Devices | 455

auto-configuration (System)

Syntax

```
auto-configuration {
    traceoptions {
        file {
        filename;
            files number;
            match reqular-expression;
            size maximum-file-size;
            (world-readable | no-world-readable);
        }
        flag flag;
        level (all | error | info | notice | verbose | warning);
        no-remote-trace;
    }
}
```

Hierarchy Level

```
[edit system]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure the autoconfiguration process.

Options

traceoptions—Set the trace options.

- **file**—Configure the trace file information.
  - **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 **/var/log**. 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 trace-file 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.

    If you specify a maximum number of files, you also must specify a maximum file size with the **size** option and a filename.

Range: 2 through 1000 files

Default: 10 files

- **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

Autoinstallation Overview | 440
Example: Configuring Autoinstallation on SRX Series Devices | 455

auto-image-upgrade

Syntax

```
auto-image-upgrade;
```

Hierarchy Level

```
[edit chassis (EX Series)]
```

Release Information

Statement introduced in Junos OS Release 9.6 for EX Series switches.

Description

Enable automatic software download on an EX Series switch acting as a DHCP client.

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.

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.

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

Upgrading Software by Using Automatic Software Download for Switches | 196
Understanding Software Installation on EX Series Switches | 164

auto-snapshot

Syntax

```
auto-snapshot;
```

Hierarchy Level

```
[edit system]
```

Release Information

Statement introduced in Junos OS Release 12.3 for EX Series switches.
Statement introduced in Junos OS Release 15.1X49-D35 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Statement introduced in Junos OS Release 20.1R1for SRX380 devices.

Description

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.

> **NOTE:** EX9200 switches do not support the automatic snapshot feature.

Default

- 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.
- By default, the auto-snapshot feature is disabled on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550M devices.

Required Privilege Level

system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Dual-Root Partitions | 373

show system auto-snapshot | 732

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 | 455

autoinstallation (JNU Satellite Devices)

Syntax

```
autoinstallation {
  delete-after-commit;
  configuration-servers {
    url;
  }
  interfaces {
    interface-name {
      bootp;
      rarp;
    }
  }
}
```

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

Autoinstallation of Satellite Devices in a Junos Node Unifier Group

Autoinstallation Process on Satellite Devices in a Junos Node Unifier Group | 449

bootp

Syntax

```
bootp {
    command binary-file-path;
    disable;
    failover (alternate-media | other-routing-engine);
}
```

Hierarchy Level

```
[edit system processes]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Specify the booting process.

Options

- **command *binary-file-path***—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.
    - **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

commit (System)

Syntax

```
commit {
    commit-synchronize-server;
    delta-export;
    fast-synchronize;
    notification;
    peers;
    peers-synchronize;
    persist-groups-inheritance | no-persist-groups-inheritance;
    server;
    synchronize;
}
```

The parameters for fast-synchronize and synchronize do not apply for the SRX Series.

Hierarchy Level

```
[edit system]
```

Release Information

Statement introduced in Junos OS Release 7.4.
Option **persist-groups-inheritance** added in Junos OS Release 13.2.
Option **delta-export** added in Junos OS Release 14.2.
Option **peers** added in Junos OS Release 14.2R6.
Option **peers-synchronize** added in Junos OS Release 14.2R6.
Option **no-persist-groups-inheritance** added in Junos OS Evolved Release 19.2R1 and Junos OS Release 19.4R1.

Description

Configure options for Junos OS commit.

Options

commit-synchronize-server—(Optional) Specify traceoptions for commit synchronize server actions.

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. This helps to reduce the time taken to commit the configuration changes..

fast-synchronize—(Optional) Configure commits to run in parallel (simultaneously) on both the master and backup Routing Engines to reduce the time required for commit synchronization. The fast-synchronize configuration is valid only on systems with two Routing Engines. (Option not available for SRX Series.)

notification—(Optional) Notify applications upon commit completion.

peers—(Optional) Specify the host names or IP addresses of the MC-LAG peers and the user authentication details for the users administering the MC-LAG peers that are participating in commit synchronization.

> **NOTE:** Starting in Junos OS Release 17.1R1, the **peers** option at the [**edit system commit**] hierarchy level is not supported in batch configuration mode.

peers-synchronize—(Optional) Configure a commit synchronization on MC-LAG peers.

persist-group-inheritance—(Optional) Configure this option to improve commit performance for systems that use many configuration groups that use wildcards. This option causes the full inheritance paths of the configuration groups to be built in the database instead of in the process memory. Starting in Junos OS Evolved Release 19.2R1 and Junos OS Release 19.4R1, this option is enabled by default. To disable this option, use **no-persist-groups-inheritance**.

server—(Optional) Configure a default batch commit.

synchronize—(Optional) For devices with multiple Routing Engines only. Configure the commit command to automatically perform a commit synchronize action between dual Routing Engines within the same chassis. The Routing Engine on which you execute the commit command (the requesting Routing Engine) copies and loads its candidate configuration to the other (the responding) Routing Engine. Each Routing Engine then performs a syntax check on the candidate configuration file being committed. If no errors are found, the configuration is activated and becomes the current operational configuration on both Routing Engines. (Option not available for SRX Series.)

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level
maintenance—To view this statement in the configuration.
maintenance-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Improving Commit Time When Using Configuration Groups

server

synchronize

commit-synchronize-server

Syntax

```
commit-synchronize-server {
    traceoptions {
        file {
            filename;
            files number;
            microsecond-stamp;
            size maximum-file-size;
            (world-readable | no-world-readable);
        }
        flag (all | debug | ephemeral-commit | operational-command);
        no-remote-trace;
    }
}
```

Hierarchy Level

```
[edit system commit]
```

Release Information

Statement introduced in Junos OS Release 16.1R3.

Description

For commit synchronize server actions, configure tracing operations.

Options

filename—Name of the file to receive the output of the tracing operation.

> **NOTE:** If you configure traceoptions and do not explicitly specify a filename for logging the events, the events are logged in the file **/var/log/commitd** by default.

files *number*—Maximum number of trace files.

microsecond-stamp—Include microsecond in the timestamp.

size—Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB).

world-readable | no-world-readable—Grant all users permission to read archived log files, or restrict the permission only to the root user and users who have the Junos OS maintenance permission.

flag ***flag***—Tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements. You can include the following flags:

- **all**—All tracing operations flags.
- **debug**—Trace operations for debug events.
- **ephemeral-commit**—Trace operations for ephemeral database commit synchronize events.
- **operational-command**—Trace operations for operational command events.

world-readable | no-world-readable—Grant all users permission to read archived log files, or restrict the permission only to the root user and users who have the Junos OS maintenance permission.

no-remote-trace—Disable remote tracing.

Required Privilege Level
system—To view this statement in the configuration.
system-control—To add this statement to the configuration.

RELATED DOCUMENTATION

commit (System) | **501**

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://usenamer: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

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

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.

Required Privilege Level

system—To view this statement in the configuration.
system-control—To add this statement to the configuration.

RELATED DOCUMENTATION

file (App Engine Virtual Machine Management Service)

Syntax

```
file {filename <files number> | match | no-world-readable | size size | 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

filename—Name of the file in which the trace information is stored. By default, the file is created in the **/var/log** directory.

files *number*—(Optional) Maximum number of trace files. When a trace file reaches the size specified by the **size** option, the filename is appended with 0 and compressed. For example, when a trace file named trace-file-log reaches size *size*, it is compressed and renamed as trace-file-log.0.gz. When trace-file-log reaches size *size* 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 sheme ensures that the older logs have a greater index number. When number of trace files reaches *number*, the oldest file is deleted.

If you specify a maximum number of files, you also must specify a maximum file size with the **size** option and a filename.

Range: 2 through 1000

Default: 10

match—Refine the output to include only those lines that match the given regular expression.

no-world-readable—Restrict file access to the user who created the trace files.

size*size*—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.

Range: 10 KB through 1 GB

Default: 128 KB

world-readable—Enable unrestricted file access.

Required Privilege Level

system-control

RELATED DOCUMENTATION

traceoptions (App Engine Virtual Machine Management Service) | **518**

level (App Engine Virtual Machine Management Service) | **513**

flag (App Engine Virtual Machine Management Service) | **509**

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

all—Trace all events.

ccif—Trace compute node interface events. This is the default option.

configuration—Trace configuration events.

heartbeat—Trace compute node heartbeat-related events.

init—Trace initialization events.

miscellaneous—Trace miscellaneous events.

platform—Trace platform-related events.

pxe—Trace events related to Preboot Execution Environment (PXE).

routing-instances—Trace events related to routing instances.

snmp—Trace SNMP events.

Required Privilege Level

system-control

RELATED DOCUMENTATION

traceoptions (App Engine Virtual Machine Management Service) | **518**

file (App Engine Virtual Machine Management Service) | **507**

level (App Engine Virtual Machine Management Service) | **513**

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

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

all—Match all levels.

error—Match error conditions.

info—Match informational messages.

notice—Match conditions that must be handled specially.

verbose—Match verbose messages.

warning—Match warning messages.

Required Privilege Level

system-control

RELATED DOCUMENTATION

traceoptions (App Engine Virtual Machine Management Service) | 518
flag (App Engine Virtual Machine Management Service) | 509
file (App Engine Virtual Machine Management Service) | 507

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.

notification (Commit)

Syntax

```
notification;
```

Hierarchy Level

```
[edit system commit]
```

Release Information

Statement introduced in Junos OS Release 16.1R3.

Description

Notify applications upon commit completion.

Options

There are no options for this configuration statement.

Required Privilege Level

system—To view this statement in the configuration.
system-control—To add this statement to the configuration.

RELATED DOCUMENTATION

commit (System) | 501

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

processes

level (App Engine Virtual Machine Management Service) | **513**

flag (App Engine Virtual Machine Management Service) | **509**

file (App Engine Virtual Machine Management Service) | **507**

traceoptions (System License)

Syntax

```
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 license]
```

Release Information

Statement introduced in Junos OS Release 8.5 for SRX Series and vSRX.
Statement introduced in Junos OS Release 14.1X53-D10 for EX Series and QFX Series.
Statement introduced in Junos OS Release 15.1 for M Series, MX Series, and T Series.

Description

Set trace options for licenses.

Options

file—Configure the trace file information.

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 **/var/log**. 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 **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.

If you specify a maximum number of files, you also must specify a maximum file size with the **size *maximum file-size*** option.

Range: 2 through 1000 files

Default: 10 files

match *regular-expression*—Refine the output to include lines that contain the regular expression.

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.

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 *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.

usb

Syntax

```
usb {
    disable;
}
```

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

Example: Configuring Autoinstallation on SRX Series Devices | 455

vmhost

Syntax

```
vmhost {
    interfaces {
    (management-if0|management-if1)
        {
        family (inet |inet6) {
        address;
        {
        master-only;
        }
        gateway;
        }
    }
    }
    management-if {
    disable;
    link-mode;
    speed;
    }
    no-auto-recovery;
    services {
    ssh {
        root-login (allow|deny);
    }
    }
    syslog {
    host <hostname>;
    }
}
```

Hierarchy Level

```
[edit]
```

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.

Statement introduced in Junos OS Release 18.1R1 for the QFX10002-60C switches.

Statement introduced in Junos OS Release 18.2R1 for the PTX10002-60C routers.

NOTE: PTX3000 router supports the Routing and Control Board, RCBPTX.

Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

Description

Configure VM host management properties. Set values in the **edit vmhost** hierarchy of the configuration.

Options

interfaces—Configure interface properties of the host.

- management-if0 | management-if1—Configure the host's side management interface0 or interface1.

  family (inet|inet6) address— Configure IPv4 or IPv6 parameters.

  master-only— Configure the IP address to be used when the Routing Engine is the current master. The configured IP address is assigned to master RE host when the Routing Engine is the current master. It is recommended to set this option for platforms with dual Routing Engine architecture with VM host support.

  gateway— Configure gateway IP address.

management-if—Configure management interface properties of the host.

- disable—Disable the host interface eth0, which serves as the management port
- link-mode (automatic | half-duplex | full-duplex)—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.
- speed (automatic | 10m | 100m | 1g)—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.

no-auto-recovery— Disable the automatic guest recovery by the host.

services— Enable ssh access to the host and enable or disable root-login to the host from guest.

- ssh—Allow ssh access

  root-login—Configure host root access through ssh

  allow | deny—Allow or deny root access through ssh.

syslog—Enable the remote syslog configuration from guest to host OS. Based on the severity configured on guest, the syslog information is logged onto the **/etc/syslog.conf** file on the host.

- host <host-name>—Host notified for remote syslog configuration.

Required Privilege Level

system—To view this statement in the configuration.
system-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Routing Engines with VM Host Support | 318

Disabling Autorecovery on Routing Engines with VM Host Support | 350

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.

Statement introduced in Junos OS Release 18.1R1 for the QFX10002-60C switches.
Statement introduced in Junos OS Release 18.2R1 for the PTX10002-60C routers.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 | 523
vmhost management-if speed | 531
vmhost management-if link-mode | 529

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.

> **NOTE:** The command is supported on the routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines only.

Statement introduced in Junos OS Release 18.1R1 for the QFX10002-60C switches.
Statement introduced in Junos OS Release 18.2R1 for the PTX10002-60C routers.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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

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**.

half-duplex—Set the link mode of the management interface to half-duplex.

full-duplex—Set the link mode of the management interface to full-duplex.

Required Privilege Level

system—To view this statement in the configuration.
system-control—To add this statement to the configuration.

RELATED DOCUMENTATION

vmhost | 523
vmhost management-if disable | 527
vmhost management-if speed | 531
show vmhost management-if | 842

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.

Statement introduced in Junos OS Release 18.1R1 for the QFX10002-60C switches.
Statement introduced in Junos OS Release 18.2R1 for the PTX10002-60C routers.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 | 523

vmhost management-if disable | 527

vmhost management-if link-mode | 529

show vmhost management-if | 842

upgrade-mode

Syntax

```
upgrade-mode (3d-fabric | default | t4000);
```

Hierarchy Level

```
[edit chassis fabric],
[edit chassis member name fabric]
```

Release Information

Statement introduced in Junos OS Release 7.5.
Statement introduced in Junos OS Release 15.1F3 for PTX Series Routers with third-generation FPCs.

Description

Configures upgrade mode for SIBs and forces them to operate in the same mode until the upgrade is complete. Upgrade mode is used so that two different types of SIBs can be installed in an operational router or routing matrix. After you upgrade the SIBs delete the **upgrade-mode** statement from the configuration. See the hardware installation guide for your router for more information about upgrading SIBs in an operational router or routing matrix.

Options

3d-fabric—Enables the TX Matrix Plus router to upgrade to a TX Matrix Plus router with 3D SIBs. On the SFC, enables setting proper support for mixed SIBs (TXP-F13 SIB and TXP-F13-3D SIB). On the T640 or T1600 or T4000 routers in a routing matrix enables support for mixed SIBs (TXP-T1600 SIB and TXP-3D-LCC SIBs on the T1600 router and SIB-I-T4000 and TXP-3D-LCC SIBs on the T4000 router).

default—Enables support for mixed SIBs when upgrading SIBs in the PTX3000 and PTX5000 routers.

t4000—Enables support for mixed SIBs when upgrading to SIB-I-T4000 SIBs in a T640 or T1600.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

TX Matrix Router and T640 Router Configuration Overview

Configuring Junos OS to Upgrade a T4000 Router to LCC 0 of a TX Matrix Plus Router with 3D SIBs

Configuring Junos OS to Upgrade the T1600 Router Chassis to LCC 0 of a TX Matrix Plus Router with 3D SIBs

Upgrading an Operational Standalone Router and Integrating It into a TX Matrix Plus Routing Matrix with 3D SIBs

Upgrading to SIB3-SFF-PTX SIBs in an Operational PTX3000

Upgrading the FPCs in an Operational PTX5000

13

CHAPTER

Operational Commands

clear system login lockout

Syntax

```
clear system login lockout
<all>
<user username>
```

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 *username*—Clear the specified locked user account.

Required Privilege Level

clear

RELATED DOCUMENTATION

retry-options

show system login lockout | 752

Output Fields

This command produces no output.

request node (offline | online)

Syntax

```
request node (offline | online) node-name
```

Release Information

Statement introduced in Junos OS Evolved Release 18.3R1.

Description

Offline or online a specified node. Online means to add the node into the cluster. Offline means to remove the node from the cluster. Use the **offline** option to stop all applications on the node (and move them to other nodes if applicable). The node will not be allowed to join the cluster until the node is made online by using the **request node online** command.

> **NOTE:** When you use the **request node offline** for fpc or spmb nodes, the node will be powered off. When used for an re node, the node will just reboot.

Options

node-name—Specify the name of the node to offline or online.

(offline | online)—Specify online or offline.

Required Privilege Level

view

RELATED DOCUMENTATION

request node reboot (re0 | re1) | 540
request system application | 541

request node reboot (re0 | re1)

Syntax

```
request node reboot (re0 |re1)
<(at time | in minutes)>
<message message>
```

Release Information

Command introduced in Junos OS Release Evolved Release 18.3R1.

Description

Reboot a specified Routing Engine. To manage all nodes at once, use the **request system shutdown** command.

Options

(at *time* | in *minutes*)—(Optional) Specify when the action is performed, either in time, in *hh*:*mm* format, or in number of minutes.

message *message*—(Optional) Message to display to all users.

(re0 |re1)—Specify the Routing Engine to reboot.

Required Privilege Level

view

RELATED DOCUMENTATION

request system application | 541

request system application

Syntax

```
request system application app application-name node node-name restart
```

Release Information

Statement introduced in Junos OS Evolved Release 18.3R1.

Description

Stop and then start (restart) an application on the specified node. (Use the **show system applications** command to verify if an application is started or stopped.)

Options

app *application-name*—Specify the application you want started or stopped.

node *node-name*—Specify the node on which to start or stop the application.

restart—Restart the application.

Required Privilege Level

view

RELATED DOCUMENTATION

request node reboot (re0 | re1) | 540

request node (offline | online) | 539

List of Sample Output

request system application app application-name node node-name restart on page 541

Sample Output

request system application app application-name node node-name restart

user@host> **request system application app cmdd node fpc0 restart**

```
This may affect traffic in the system. Proceed ? [yes,no] (no) yes

App cmdd on node fpc0 restart request is submitted
```

request system autorecovery state

Syntax

```
request system autorecovery state (save | recover | clear)
```

Release Information

Command introduced in Junos OS Release 15.1X49-D35 for SRX300, SRX320, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 20.1R1for SRX380 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 | 720

List of Sample Output

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 bsdlabel 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 bsdlabel recovery information
```

request system download abort

Syntax

```
request system download abort <download-id>
```

Release Information

Command introduced in Junos OS Release 15.1X49-D35 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 20.1R1for SRX380 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 | 552
request system download pause | 548
request system download resume | 550
request system download clear | 547

List of Sample Output

request system download abort on page 546

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 15.1X49-D35 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 20.1R1for SRX380 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

request system download start | 552
request system download pause | 548
request system download resume | 550
request system download abort | 545

List of Sample Output

request system download clear on page 547

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 15.1X49-D35 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 20.1R1for SRX380 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 | 552
request system download resume | 550
request system download abort | 545
request system download clear | 547

List of Sample Output

request system download pause on page 549

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 15.1X49-D35 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 20.1R1for SRX380 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 | 552
request system download pause | 548
request system download abort | 545
request system download clear | 547

List of Sample Output

request system download resume on page 551

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

```
request system download start (sftp-url | delay | identity-file | login | max-rate | passphrase | save as )
```

Release Information

Command introduced in Junos OS Release 15.1X49-D35 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 20.1R1for SRX380 devices.

Description

Create a download instance and identify it with a unique integer called the download ID.

Options

sftp-url—(Required) The FTP or HTTP URL location of the file to be downloaded securely.

delay—(Optional) The number of hours after which the download should start (range from 1 through 48 hours).

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.

login—(Optional) The username and password for the server in the format **username:password**.

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.

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.

save-as—(Optional) The filename to be used for saving the file in the **/var/tmp** location.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system download pause | 548
request system download resume | 550
request system download abort | 545
request system download clear | 547

List of Sample Output

request system download start on page 553

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

```
request system firmware upgrade
```

Release Information

Command introduced in Junos OS Release 10.2.

Description

Upgrade firmware on a system.

Options

fpc—Upgrade FPC ROM monitor.

pic—Upgrade PIC firmware.

re—Upgrade baseboard BIOS/FPGA. There is an active BIOS image and a backup BIOS image.

- **bios**—(Optional) Upgrade BIOS.
- **fpga**—(Optional)Upgrade baseboard FPGA.
- **i40nvm**—(Optional) Upgrade baseboard i40.
- **ssd**—(Optional)Upgrade Routing Engine solid-state drive (SSD) firmware.

  disk1—Upgrade SSD disk1 firmware.

  disk2—Upgrade SSD disk2 firmware.

> **NOTE:** Starting in Junos OS Release 17.2R1, you can upgrade the SSD firmware on routers with the VM Host support.
>
> Starting in Junos OS Release 19.3R1, you can upgrade the i40e NVM firmware on routers with VM Host support.

vcpu—Upgrade VCPU ROM monitor.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system halt

List of Sample Output

request system firmware upgrade on page 555

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   Available Status
                                     version   version
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   Available Status
                                     version   version
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   Available    Status
                                   version   version
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

```
request system halt
at <time>
in <minutes>
media (compact-flash | disk | usb)
messages <message>
```

Release Information

Command introduced in Junos OS Release 11.4.

Description

Stop the system.

Options

at *time*— Time at which to stop the system.

in *minutes*— Number of minutes to delay before halting the system.

media —Boot media for the next boot.

- **compact-flash**— Standard boot from a flash device.
- **disk**— Boot from a hard disk.
- **usb**— Boot from a USB device.

message *message*— Message that is displayed to all system users before stopping the system.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system power-off | 567

List of Sample Output

request system halt on page 557

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 2 1 1 1 1 1 1 1 1 0 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

```
request system license add (filename | terminal)
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 9.5 for SRX Series devices.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Added additional information section on XML RPC in Junos OS Release 17.4.

Description

Adding a license key to the Junos OS devices to activate the feature.

Starting in Junos OS Release 18.3R1, the **display xml rpc** CLI option is supported for **request system license add** and **request system license save** commands while installing licenses on Juniper Networks devices.

Options

filename—License key from a file or URL. Specify the filename or the URL where the key is located.

terminal—License key from the terminal.

Additional Information

The **| display xml rpc** filter returns “xml rpc equivalent of this command is not available,” the following RPC is supported for license installation:

The following RPC is supported for license installation:

```
<rpc>
<request-license-add>
<key-data> key </key-data>
</request-license-add>
</rpc>
```

Where ***key-data*** is the license key data.

```
<rpc>
<request-license-add>
<filename> key-file </filename>
```

```
</request-license-add>
</rpc>
```

Where ***source*** is the URL of the source license key file.

Required Privilege Level

maintenance

List of Sample Output

request system license add on page 559

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**

```
XXXXXXXXXX xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
         xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx xxxxxx
         xxxxxx xxxxxx xxx
XXXXXXXXXX: successfully added
add license complete (no errors)
```

request system license delete

Syntax

```
request system license delete ( license-identifier | license-identifier-list [ licenseid001 licenseid002 licenseid003 ] | all
  )
```

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

license-identifier—Text string that uniquely identifies a license key.

license-identifier-list [*licenseid001 licenseid002 licenseid003....*]—Delete multiple license identifiers as a list enclosed in brackets.

all—Delete all licenses on the device.

Required Privilege Level

maintenance

request system license save

Syntax

```
request system license save (filename | terminal)
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 9.5 for SRX Series devices.
Added additional information section on XML RPC in Junos OS Release 17.4.

Description

Save installed license keys to a file or URL.

Starting in Junos OS Release 18.3R1, the **display xml rpc** CLI option is supported for **request system license add** and **request system license save** commands while installing licenses on Juniper Networks devices.

Options

filename—License key from a file or URL. Specify the filename or the URL where the key is located.

terminal—License key from the terminal.

Additional Information

The following RPC is supported for saving installed license keys to a file or URL:

```
<rpc>
<request-license-save>
<filename>destination</filename>
</request-license-save>
</rpc>
```

Where ***destination*** is the URL of the destination license key file.

Required Privilege Level

maintenance

List of Sample Output

request system license save on page 562

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

```
request system license update
```

Release Information

Command introduced in Junos OS Release 9.5.

Description

Starts autoupdating license keys from the license portal.

- The **request system license update** command always uses the default Juniper license server: **https://ae1.juniper.net/**.
- The **request system license update** command is supported only on SRX, vSRX, and QFX Series devices.

The products supported by the Juniper Agile Licensing (JAL) portal includes: QFX series, SRX Series, EX Series, NFX, vBNG, vMX, vSRX, and ACX. For other Juniper products (SPACE, JSA, SBR Carrier, Screen OS and so on) access the License Management System (LMS).

Options

trial—Immediately updates trial license keys from the license portal.

Required Privilege Level

maintenance

List of Sample Output

request system license update on page 563
request system license update trial on page 564

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.

> **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 35.

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

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

request system partition compact-flash (If Yes)

`user@host>` **request system partition compact-flash**

```
Are you sure you want to reboot
and partition the compact-flash ? [yes,no] yes
Initiating repartition operation.
The operation may take several minutes to complete.
```

```
System will reboot now...
<System reboots>
<Repartition operation is performed>
<System reboots and starts up normally>
```

Sample Output

request system partition compact-flash (If No)

`user@host>` **request system partition compact-flash**

```
Are you sure you want to reboot
and partition the compact-flash ? [yes,no] no
```

request system power-off

Syntax

```
request system power-off
at <time>
in <minutes>
junos
message <message>
network
oam
power-off
usb
media (compact-flash | disk | usb | internal)
```

Release Information

Command introduced in Junos OS Release 11.4.

Description

Power off the system.

Options

at *time*— Time at which to power off the system.

in *minutes*— Number of minutes to delay before powering off the system.

media —Boot media for the next boot.

- **compact-flash**— Standard boot from a flash device.
- **disk**— Boot from a hard disk.
- **usb**— Boot from a USB device.
- **internal**— Boot from internal flash.

message *message*— Message that is displayed to all system users before powering off the system.

junos—(SRX1500, SRX5400, SRX5600, and SRX5800) Boot off Junos volume.

network—(SRX1500, SRX5400, SRX5600, and SRX5800) Network boot through PXE.

oam—(SRX1500, SRX5400, SRX5600, and SRX5800) Boot off OAM volume.

usb—(SRX1500, SRX5400, SRX5600, and SRX5800) Boot off USB device.

power-off—(SRX1500) Power off the software on RE.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system halt | 556

List of Sample Output

request system power-off on page 568

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 2 1 1 1 1 1 1 1 1 1 0 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

```
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 and EX Series Virtual Chassis)

```
request system reboot
<all-members | local | member member-id>
<at time>
<in minutes>
<media (external | internal)> | <media (compact-flash | disk | removable-compact-flash | usb)>
<message "text">
<slice slice>
```

Syntax (MX Series Routers and MX Series Virtual Chassis, EX9200 Switches and EX9200 Virtual Chassis)

```
request system reboot
<all-members | local | member member-id>
<at time>
<both-routing-engines>
<in minutes>
<media (external | internal)> | <media (compact-flash | disk | usb)> | <junos | network | oam | usb>
<message "text">
<other-routing-engine>
```

Syntax (QFabric Systems)

```
request system reboot
<all <graceful>>
<at time>
<director-device name>
<director-group <graceful>>
<fabric <graceful>>
<in minutes>
<in-service>
<media>
<message “text”>
<node-group name>
<slice slice>
```

Syntax (QFX Series Switches and QFX Series Virtual Chassis, Virtual Chassis Fabric)

```
request system reboot
<all-members | local | member member-id>
<at time>
<in minutes>
<in-service>
<hypervisor>
<junos | network | oam | usb>
<message “text”>
<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)>
```

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 **partition** changed to **slice** in Junos OS Release 10.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Option **both-routing-engines** introduced in Junos OS Release 12.1.

Description

Reboot the software.

This command can be used on standalone devices and on devices supported in a Virtual Chassis, Virtual Chassis Fabric, or QFabric system.

> **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. Starting from Junos OS Release 17.4R1, PTX10016 routers do not support the **request system reboot** command. Use the **request vmhost reboot** command instead of the **request system reboot** command on the PTX10008 and PTX10016 routers to reboot the Junos OS software package or bundle on the router. See request vmhost reboot.

> **NOTE:** On a QFabric system, to avoid traffic loss on the network Node group, switch mastership of the Routing Engine to the backup Routing Engine, and then reboot.

Options

The options described here are not all supported on every platform or release of Junos OS. Refer to the Syntax sections for the options commonly available on each type of platform.

none—Reboot the software immediately.

all-chassis—(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—(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 | local | member *member-id*—(Optional) Specify which member of the Virtual Chassis to reboot:

- **all-members**—Reboots each switch that is a member of the Virtual Chassis.
- **local**—Reboots only the local switch (switch where you are logged in).
- **member *member-id***—Reboots the specified member switch of the Virtual Chassis

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.

hypervisor—(Optional) Reboot Junos OS, host OS, and any installed guest VMs.

in *minutes*—(Optional) Number of minutes from now to reboot the software. The minimum value is 1. This option is an alias for the **at *+minutes*** option.

in-service—(Optional) Enables you to reset the software state (no software version change) of the system with minimal disruption in data and control traffic.

junos—(Optional) Reboot from the Junos OS (main) volume.

lcc *number*——(Optional) Line-card chassis (LLC) 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.

media (compact-flash | disk | removable-compact-flash | usb)—(Optional) Use the indicated boot medium for the next boot.

media (external | internal)—(Optional) Use the indicated boot medium for the next boot:

- **external**—Reboot the device using a software package stored on an external boot source, such as a USB flash drive.
- **internal**—Reboot the device using a software package stored in an internal memory source.

message "*text*"—(Optional) Message to display to all system users before stopping or rebooting the software.

network—(Optional) Reboot using the Preboot Execution Environment (PXE) boot method over the network.

oam—(Optional) Reboot from the maintenance volume (OAM volume, usually the compact flash drive).

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 *partition*—(Optional) Reboot using the specified partition on the boot media. This option is equivalent to the **slice** option that is supported on some devices. Specify one of the following *partition* values:

- **1**—Reboot from partition 1.
- **2**—Reboot from partition 2.
- **alternate**—Reboot from the alternate partition.

scc—(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*—(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*—(Optional) Reboot using the specified partition on the boot media. This option was originally the **partitiion** option but was renamed to **slice** on EX Series and QFX Series switches. Specify one of the following *slice* values:

- **1**—Reboot from partition 1.

- **2**—Reboot from partition 2.
- **alternate**—Reboot from the alternate partition (which did not boot the switch at the last bootup).

> **NOTE:** The **slice** option is not supported on QFX Series switches that have no alternate slice when Junos OS boots as a Virtual Machine (VM). To switch to the previous version of Junos OS, issue the **request system software rollback** command.

usb—(Optional) Reboot from a USB device.

The following options are available only on QFabric Systems:

all—(Optional) Reboots the software on the Director group, fabric control Routing Engines, fabric manager Routing Engines, Interconnect devices, and network and server Node groups.

director-device ***name***—(Optional) Reboots the software on the Director device and the default partition (QFabric CLI).

director-group—(Optional) Reboots the software on the Director group and the default partition (QFabric CLI).

fabric—(Optional) Reboots the fabric control Routing Engines and the Interconnect devices.

node-group ***name***—(Optional) Reboots the software on a server Node group or a network Node group.

graceful—(Optional) Enables the QFabric component to reboot with minimal impact to network traffic. This sub-option is only available for the **all**, **fabric**, and**director-group** options.

Additional Information

Reboot requests are recorded in the system log files, which you can view with the **show log** command (see *show log*). Also, the names of any running processes that are scheduled to be shut down are changed. You can view the process names with the **show system processes** command (see *show system processes*).

On a TX Matrix or TX Matrix Plus router, if you issue the **request system reboot** command on the master Routing Engine, all the master Routing Engines connected to the routing matrix are rebooted. If you issue this command on the backup Routing Engine, all the backup Routing Engines connected to the routing matrix are rebooted.

NOTE: Before issuing the **request system reboot** command on a TX Matrix Plus router with no options or the **all-chassis**, **all-lcc**, **lcc *number***, or **sfc** options, verify that master Routing Engine for all routers in the routing matrix are in the same slot number. If the master Routing Engine for a line-card chassis is in a different slot number than the master Routing Engine for a TX Matrix Plus router, the line-card chassis might become logically disconnected from the routing matrix after the **request system reboot** command.

NOTE: To reboot a router that has two Routing Engines, reboot the backup Routing Engine (if you have upgraded it) first, and then reboot the master Routing Engine.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

clear system reboot

request system halt

Routing Matrix with a TX Matrix Plus Router Solutions Page

request vmhost reboot | 799

List of Sample Output

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 in-service

user@switch> **request system reboot in-service**

```
Reboot the system ? [yes,no]
[Feb 22 02:37:04]:ISSU: Validating Image

 PRE ISSR CHECK:
 ---------------
 PFE Status                            : Online
 Member Id zero                        : Valid
 VC not in mixed or fabric mode        : Valid
 Member is single node vc              : Valid
 BFD minimum-interval check done       : Valid
 GRES enabled                          : Valid
 NSR enabled                           : Valid
 drop-all-tcp not configured           : Valid
 Ready for ISSR                        : Valid

warning: Do NOT use /user during ISSR. Changes to /user during ISSR may get lost!
Current image is jinstall-jcp-i386-flex-18.1.img
[Feb 22 02:37:14]:ISSU: Preparing Backup RE
Prepare for ISSR
[Feb 22 02:37:19]:ISSU: Backup RE Prepare Done
Spawning the backup RE
Spawn backup RE, index 1 successful
Starting secondary dataplane
Second dataplane container started
GRES in progress
Waiting for backup RE switchover ready
GRES operational
Copying home directories
Copying home directories successful
Initiating Chassis In-Service-Upgrade for ISSR
Chassis ISSU Started
[Feb 22 02:42:55]:ISSU: Preparing Daemons
[Feb 22 02:43:00]:ISSU: Daemons Ready for ISSU
[Feb 22 02:43:05]:ISSU: Starting Upgrade for FRUs
[Feb 22 02:43:15]:ISSU: FPC Warm Booting
[Feb 22 02:44:16]:ISSU: FPC Warm Booted
[Feb 22 02:44:27]:ISSU: Preparing for Switchover
[Feb 22 02:44:31]:ISSU: Ready for Switchover
Checking In-Service-Upgrade status
  Item           Status                  Reason
  FPC 0          Online (ISSU)
Send ISSR done to chassisd on backup RE
Chassis ISSU Completed
Removing dcpfe0 eth1 128.168.0.16 IP
```

```
Bringing down bme00
Post Chassis ISSU processing done
[Feb 22 02:44:33]:ISSU: IDLE
Stopping primary dataplane
Clearing ISSU states
Console and management sessions will be disconnected. Please login again.
device_handoff successful ret: 0
Shutdown NOW!
[pid 14305]

*** FINAL System shutdown message from root@sw-duckhorn-01 ***

System going down IMMEDIATELY
```

request system reboot (Junos OS with Upgraded FreeBSD)

Syntax

```
request system reboot
<at time>
<both-routing-engines>
<in minutes>
<junos>
<message "text">
<network>
<oam>
<other-routing-engine>
<usb>
```

Release Information

Command introduced in Junos OS Release 15.1 for MX240, MX480, MX960, MX2010, and MX2020 routers and EX9200 switches.
Command introduced in Junos OS Release 15.1X53-D30 for QFX5200 switches.

Description

Reboot the software.

Options

none—Reboot the software immediately.

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. Omitting a value will default to the current date for that value.
- ***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.

junos—(Optional) Reboot from the **junos** volume.

message "*text*"—(Optional) Message to display to all system users before stopping or rebooting the software.

network—(Optional) Reboot from the network.

oam—(Optional) Reboot from the **oam** volume.

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.

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) | 603
show system snapshot (Junos OS with Upgraded FreeBSD) | 760
clear system reboot
request system halt
Release Information for Junos OS with Upgraded FreeBSD | 35

List of Sample Output

request system reboot on page 582
request system reboot (at 2300) on page 582
request system reboot (in 2 Hours) on page 582
request system reboot (Immediately) on page 582
request system reboot (at 1:20 AM) on page 582

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

```
request system recover (junos-volume | oam-volume)
```

Release Information

Command introduced in Junos OS Release 15.1 for MX240, MX480, MX960, MX2010, and MX2020 routers and EX9200 switches only.
Command introduced in Junos OS Release 15.1X53-D30 for QFX5200 switches.
Command introduced in Junos OS Release 15.1X53-D60 for QFX10000 switches.
Command introduced in Junos OS Release 16.1R1 for VM host on MX240, MX480, MX960, MX2010, and MX2020 routers and PTX5000 routers.
Command introduced in Junos OS Release 17.3 for SRX5400, SRX5600, and SRX5800 devices.

Description

Recover a specified volume of the system.

> **NOTE:** Guest network functions (GNFs) do not support the **recover** option under the **request system** command. See *Components of Junos Node Slicing* for more details on GNF.

Options

junos-volume—Specify the **/junos** volume to be recovered. The **/junos** 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**/junos** 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.

oam-volume—Specify the **/oam** volume to be recovered. The compact flash drive is the **/oam** volume and stores recovery snapshot backup information. In case of failure of the **/junos** volume, the **/oam** volume can be used to boot the system. The **/oam** 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.

Additional Information

If you try to recover the junos volume while you are booted on the junos volume, you will get an error message.

To recover the junos volume, do the following:

1. Recover the oam volume.

```
user@host> request system recover oam
```

2. Reboot on the oam volume.

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

Required Privilege Level
view

RELATED DOCUMENTATION

Release Information for Junos OS with Upgraded FreeBSD | 35
Changes in Disk Volumes for Junos OS with Upgraded FreeBSD | 43
Changes in Use of Snapshots for Junos OS with Upgraded FreeBSD | 44

List of Sample Output
request system recover junos-volume (While booted on the junos volume) on page 585
request system recover junos-volume (While booted on the oam volume) on page 585
request system recover oam-volume on page 586

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 | 588
request system scripts rollback | 589
request system scripts event-scripts reload

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 | 587

request system scripts rollback | 589

request system scripts event-scripts reload

request system scripts rollback

Syntax

```
request system scripts rollback
```

Release Information

Command introduced before Junos OS Release 9.0.

Description

Attempt to roll back to most recent installation of AI-Scripts (jais) package.

Options

No options are available.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system scripts add | 587
request system scripts delete | 588
request system scripts event-scripts reload

request system snapshot

List of Syntax

Syntax (ACX Series Routers)

```
request system snapshot
<media type>
<partition>
```

Syntax (EX Series Switches; for EX4600, see QFX Series Syntax)

```
request system snapshot
<all-members | local | member member-id>
<media type>
<partition>
<re0 | re1 | routing-engine routing-engine-id>
<slice alternate>
```

Syntax (MX Series Routers)

```
request system snapshot
<all-members>
<config-partition>
<local>
<member member-id>
<media usb-port-number>
<partition>
<root-partition>
```

Syntax (PTX Series)

```
request system snapshot
<partition>
```

Syntax (QFX Series, OCX1100, and EX4600)

```
request system snapshot
<all-members | local | member member-id>
<config-partition>
<partition>
<root-partition>
<slice alternate>
```

Syntax (TX Matrix Routers)

```
request system snapshot
<all-chassis | all-lcc | lcc number |  scc>
<config-partition>
<partition>
<root-partition>
```

Syntax (TX Matrix Plus Routers)

```
request system snapshot
<all-chassis | all-lcc | lcc number |  sfc number>
<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 11.3 for the QFX Series.
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.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
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 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**.

Description

- On routers running Junos OS, 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 switches running Junos OS, 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 version of Junos OS, the active configuration, and log files.
- On devices running Junos OS Evolved, take a snapshot of the contents of the **/soft** directory, which includes the running version of Junos OS Evolved.

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: System snapshot is not supported on QFX10000 switches.

NOTE: Starting with Junos OS Release 15.1F3, the command **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 command **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 "Configuring Dual-Root Partitions" on page 373.

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 Virtual Chassis, MX Series routers, QFX Series switches, QFabric System, and OCX1100 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—(EX Series Virtual Chassis, MX Series routers, QFX Series switches, QFabric System, OCX1100, and T and 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 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**.

media *type*—(ACX Series, M320, T640, and MX Series routers) (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, and, except for MX104, MX Series routers) 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.

(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, and TX Series routers; EX Series Virtual Chassis; QFX Series switches; QFabric System; and OCX1100 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 35.

slice alternate—(EX Series switches, EX Series Virtual Chassis, QFX Series switches, QFabric System, and OCX1100 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 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**.

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.
- (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

view

RELATED DOCUMENTATION

request system snapshot (Junos OS with Upgraded FreeBSD) | **603**

show system snapshot | **756**

show system auto-snapshot | **732**

List of Sample Output

request system snapshot (Routers) on page 597
request system snapshot (EX Series Switches) on page 597
request system snapshot partition (EX4600, QFX Series, QFabric System, and OCX1100) on page 597
request system snapshot (When the Partition Flag Is On) on page 598
request system snapshot (MX104 Routers When Media Device is Missing) on page 598
request system snapshot (When Mirroring Is Enabled) on page 598
request system snapshot all-lcc (Routing Matrix) on page 598
request system snapshot all-members (Virtual Chassis) on page 599
request system snapshot (Junos OS Evolved) on page 599

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 partition (EX4600, QFX Series, QFabric System, and OCX1100)

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

```
Clearing current label...
Partitioning external media (da1) ...
Verifying compatibility of destination media partitions...
Running newfs (334MB) on external media  / partition ...
Running newfs (404MB) on external media  /config partition ...
Running newfs (222MB) on external media  /var partition ...
```

```
Copying '/dev/da0s2a' to '/dev/da1s1a' .. (this may take a few minutes)
Copying '/dev/da0s3e' to '/dev/da1s3e' .. (this may take a few minutes)
Copying '/dev/da0s2f' to '/dev/da1s1f' .. (this may take a few minutes)
The following filesystems were archived: / /config /var
```

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 Evolved)

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

```
                  -------------------------------
                  node: re0
```

```
                  ------------------------------
                  Snapshot device found - /dev/sdb
                  System is up for 29 minutes, proceed and initiate snapshot
                  Creating snapshot mount dirs in /tmp/snapshot
                  Creating snapshot disk filesystems and mounts
                  sfdisk: failed to dump partition table: Success
                Found 186 gig (195360984 kbytes) Vendor ATA, Model XXXXXXXXXXXXXXX-

                  Writing new partitioning table to disk sdb -
                      boot - 204800K
                      soft - 16384M
                      swap - 4096M
                      data - 3072M
                      conf - 1024M
                      var - 166006M
                      user - 0M
                  Done
                  New Partitions  - rebuilding filesystems
                  Error: Could not stat device sdb - No such file or directory.
                  Processing /dev/sdb2 for mount on /tmp/snapshot/soft
...[creating]..ok [mounting]..done
                  Processing /dev/sdb5 for mount on /tmp/snapshot/data
...[creating]..ok [mounting]..done
                  Processing /dev/sdb6 for mount on /tmp/snapshot/data/config
...[creating]..ok [mounting]..done
                  Processing /dev/sdb7 for mount on /tmp/snapshot/data/var
...[creating]..ok [mounting]..done
                  Setting up swapspace version 1, size = 4 GiB (4294963200 bytes)
                  no label, UUID=xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                   Processing /dev/sdb1 for mount on /tmp/snapshot/boot
...[creating]..ok [mounting]..mkswap: /dev/sdb3: warning: wiping old swap signature.

                  done
                  Done with local filesystems setup.
                  ........
                  Space available 14822042 K, proceeding...
                  Copying junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2...
                Copying /soft/junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2
 to /tmp/snapshot//soft/junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2
                  Installing BOM...
                      bom.xml...done
                      bom.sh...done
                      bom.py...done
                  Done.
```

```
                 Copying files to
/tmp/snapshot//soft/junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2...
                 Copying files for node re...
                 Boot objects...
                     bzImage-re-64b.bin...done
                     bzImage-re-64b.bin.psig...done
                     .bzImage-re-64b.bin.hmac...done
                     bzImage-ptx1000-rcb.dtb...done
                     bzImage-ptx1000-rcb.dtb.psig...done
                 FSF files...
                     ptx-fixed-js64_Yocto_2.2_x86_64.fs...done
                     java_Yocto_2.2_x86_64.fs...done
                 ........
                     modules_Yocto_2.2_x86_64.fs...done
                     zookeeper_Yocto_2.2_x86_64.fs...done
                     python-2.7_Yocto_2.2_x86_64.fs...done
                     dev_Yocto_2.2_x86_64.fs...done
                     perl-5.22.1_Yocto_2.2_x86_64.fs...done
                     initrd_Yocto_2.2_x86_64.fs...done
                     initrd_Yocto_2.2_x86_64.fs.psig...done
                     .initrd_Yocto_2.2_x86_64.fs.hmac...done
                     python-3.5_Yocto_2.2_x86_64.fs...done
                     jimbase_Yocto_2.2_x86_64.fs...done
                     osbase_Yocto_2.2_x86_64.fs...done
                 done.
                 Done installing modules for re.
               Copying /data/junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2

                 Copying
/data/config/junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2
                 Copying
/data/etc/junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2
                 Copying
/data/var_db/junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2
                 ........
                 Copying
/data/usr_conf/junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2
                 Copying
/data/var_rundb/junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2
                 Copying
/data/usr_evo_share/junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2
                 Copying /data/var
                 Copying /boot
```

```
          Unmounting snapshot mounts
          Snapshot commpleted for /dev/sdb
```

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. Recovery snapshots are the ones taken with the **request system snapshot recovery** command. Recovery snapshots, shown as **recovery.ufs** files, are full copies of the packages and configuration taken at the time the snapshot command is issued. QFX with TVP platforms do not support these commands.

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 | 44

show system snapshot | 756

List of Sample Output

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

user@host> **request system snapshot**

```
NOTICE: Snapshot snap.20190412.124526 created successfully
```

show system snapshot

user@host> **show system snapshot**

```
Non-recovery snapshots:
Snapshot snap.20190412.124526:
Location: /packages/sets/snap.20190412.124526
Creation date: Apr 12 12:45:26 2019
Junos version: 18.3R1.8

Total non-recovery snapshots: 1

Recovery Snapshots:
Snapshots available on the OAM volume:
recovery.ufs
Date created: Thu Mar 28 07:44:25 PDT 2019
Junos version: 18.3R1.8

Total recovery snapshots: 1
```

request system snapshot delete

`user@host>` **request system snapshot delete snap.20150112.122106**

```
NOTICE: Snapshot ’
snap.20150112.122106’
deleted 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**— Creates a snapshot of the configuration partition only and stores it onto the default **/altconfig** on the hard disk device or an **/altconfig** on a USB device.
- **root-partition**— Creates a snapshot of the root partition only and stores it onto the default **/altroot** on the hard disk device or an **/altroot** on a USB device.
- **factory**— (Optional) Specifies that only the files shipped from the factory are included in the snapshot.
- **media**—(Optional) Specify the boot device the software is copied to:
  - **compact-flash**—Copy software to the primary compact flash drive.
  - **hard-disk**— Copy software to the hard disk.
  - **usb**— Copy software to the device connected to the USB port.
  - **internal**— Copy software to an internal flash drive. This is the default option.

    > **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, SRX380, and SRX550M devices.

  - external— Copies software to an external storage device. This option is available for the compact flash on the SRX650 Services Gateway.

- **node**—(Optional) Specify the archive data and executable areas of a specific node. If you do not specify the node option, the device considers the current node as default option.
  - **node-id**—Specify for node (0, 1).
  - **all**—Specify for all nodes.
  - **local**—Specify for local nodes.
  - **primary**— Specify for primary nodes.
- **partition**—(Default) Specify that the target media should be repartitioned before the backup is saved to it.

  NOTE: The target media is partitioned whether or not it is specified in the command, because this is a mandatory option.

  Example: **request system snapshot media usb partition**

  Example: **request system snapshot media usb partition factory**

- **slice**—(Optional) Take a snapshot of the root partition the system has currently booted from to another slice in the same media.
  - **alternate**—(Optional) Store the snapshot on the other root partition in the system.

  NOTE: The slice option cannot be used along with the other **request system snapshot** options, because the options are mutually exclusive. If you use the factory, media, or partition option, you cannot use the slice option; if you use the slice option, you cannot use any of the other options.

Required Privilege Level

maintenance

List of Sample Output

request system snapshot config-partition on page 609
request system snapshot root-partition on page 609
request system snapshot media hard-disk on page 609
request system snapshot media usb (when usb device is missing on page 609
request system snapshot media compact-flash on page 610
request system snapshot partition on page 610

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 (da1s1a)...
Running newfs (47MB) on usb media  /config partition (da1s1e)...
```

```
Copying '/dev/da0s2a' to '/dev/da1s1a' .. (this may take a few minutes)
Copying '/dev/da0s2e' to '/dev/da1s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

request system snapshot media compact-flash

`user@host>` **request system snapshot media compact-flash**

```
error: cannot snapshot to current boot device
```

request system snapshot partition

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

```
Verifying compatibility of destination media partitions...
Running newfs (439MB) on internal media  / partition (da0s1a)...
Running newfs (46MB) on internal media  /config partition (da0s1e)...
Copying '/dev/da1s1a' to '/dev/da0s1a' .. (this may take a few minutes)
Copying '/dev/da1s1e' to '/dev/da0s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

request system software abort in-service-upgrade (ICU)

Syntax

```
request system software abort in-service-upgrade
```

Release Information

Command introduced in Junos OS Release 15.1X49-D35 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 20.1R1for SRX380 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

Upgrading Devices in a Chassis Cluster Using ICU

List of Sample Output

request system software abort in-service-upgrade on page 612

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

```
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>
```

Syntax (Junos OS Evolved)

```
request system software add package-name
<force>
<no-validate>
<reboot>
<restart>
```

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.
The following options are deprecated in Junos OS Evolved Release 18.3R1: **best-effort-load**, **delay-restart**, **no-copy**, **on-primary**, (**re0** | **re1**), **set**, **unlink**, **validate**, **validate-on-host**, and **validate-on-routing-engine**.

Description

For Junos OS Evolved, the **request system software add** command has a built-in feature not to start the upgrade if a reboot is pending after an upgrade or rollback.

> **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:** When graceful Routing Engine switchover (GRES) is enabled on a device, you must perform a unified in-service software upgrade (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.

NOTE: In Junos OS, ***package-name*** can be either the URL of a remote location or the pathname of a local package. But Junos OS Evolved does not support a remote iso for upgrade, so “URL” is removed from the help string in the CLI.

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 the 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).

NOTE: For Junos OS Evolved, if you are trying to reinstall an already installed application, use the **force** option. The **force** option will cause the program to remove the existing application before reinstalling it.

force-host—(Optional) Force the addition of the 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.

NOTE: The **reboot** command is not needed to install third-party applications on devices running Junos OS Evolved.

restart—(Optional) (For Junos OS Evolved only) This option allows you to upgrade the system using an application-level restart, that is, no system reboot. First the system determines how many applications need to restart (start with a new version), and then, after sending output to the CLI session, it restarts those applications. Restarted applications resync their state from the system. You can perform a dry run by using the **request system software validate-restart** command before using the **request system software add restart** command.

set [*package-name1 package-name2*]—(Mixed EX4200 and EX4500 Virtual Chassis, M Series, MX Series, and T Series routers only) (Optional) Install multiple packages at same time:

- In the case of mixed EX4200 and EX4500 Virtual Chassis, install two software packages—a package for an EX4200 switch and the same release of the package for an EX4500 switch—to upgrade all member switches in a mixed EX4200 and EX4500 Virtual Chassis.
- In the case of M Series, MX Series, and T Series routers, install multiple (two or more) software packages and software add-on packages at the same time. The variable ***package-name*** can either be a list of installation packages, each separated by a blank space, or the full URL to the directory or tar file containing the list of installation packages.

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

Use the **request system software add set** command to retain any SDK configuration by installing the SDK add-on packages along with the core Junos OS installation package.

unlink—(Optional) On M Series, T Series, and MX Series routers, use the unlink option to remove the software package from this directory after a successful upgrade is completed.

upgrade-group [all |*upgrade-group-name*]—(Junos Fusion only) (Required to configure a Junos Fusion using autoconversion or manual conversion) Associate a satellite software image with a satellite software upgrade group. The satellite software package is associated with the specified satellite software upgrade group using the *upgrade-group-name*, or for all satellite software upgrade groups in a Junos Fusion when the all keyword is specified.

A satellite software upgrade group is a group of satellite devices in a Junos Fusion that are designated to upgrade to the same satellite software version using the same satellite software package. See *Understanding Software in a Junos Fusion Provider Edge*, *Understanding Software in a Junos Fusion Enterprise*, and *Managing Satellite Software Upgrade Groups in a Junos Fusion*.

upgrade-with-config—(Optional) Install one or more configuration files.

NOTE: Configuration files specified with this option must have the extension **.text** or **.xml** and have the extension specified. Using the extension **.txt** will not work.

validate—(Optional) Validate the software package or bundle against the current configuration as a prerequisite to adding the software package or bundle. This is the default behavior when the software package or bundle being added is a different release.

NOTE: The **validate** option only works on systems that do not have **graceful-switchover** (GRES) enabled. To use the **validate** option on a system with GRES, either disable GRES for the duration of the installation, or install using the command **request system software in-service-upgrade**, which requires nonstop active routing (NSR) to be enabled when using GRES.

validate-on-host *hostname*—(Optional) Validate the software package by comparing it to the running configuration on a remote Junos OS host. Specify a host, replacing ***hostname*** with the remote hostname. You can optionally provide the username that will be used to log in to the remote host by specifying the hostname in the format **user@hostname**.

validate-on-routing-engine *routing-engine*—(Optional) Validate the software bundle or package by comparing it to the running configuration on a Junos OS Routing Engine on the same chassis. Specify a Routing Engine, replacing ***routing-engine*** with the routing engine name.

Additional Information

Before upgrading the software on the router or switch, when you have a known stable system, issue the **request system snapshot** command to back up the software, including the configuration, to the **/altroot** and **/altconfig** file systems. After you have upgraded the software on the router or switch and are satisfied that the new package or bundle is successfully installed and running, issue the **request system snapshot** command again to back up the new software to the **/altroot** and **/altconfig** file systems.

NOTE: The **request system snapshot** command is currently not supported on the QFabric system. Also, you cannot add or install multiple packages on a QFabric system.

After you run the **request system snapshot** command, you cannot return to the previous version of the software because the running and backup copies of the software are identical.

If you are upgrading more than one package at the same time, delete the operating system package, jkernel, last. Add the operating system package, jkernel, first and the routing software package, jroute, last. If you are upgrading all packages at once, delete and add them in the following order:

```
user@host> request system software add /var/tmp/jbase
user@host> request system software add /var/tmp/jkernel
user@host> request system software add /var/tmp/jpfe
user@host> request system software add /var/tmp/jdocs
user@host> request system software add /var/tmp/jroute
user@host> request system software add /var/tmp/jcrypto
```

By default, when you issue the **request system software add *package-name*** command on a TX Matrix master Routing Engine, all the T640 master Routing Engines that are connected to it are upgraded to the same version of software. If you issue the same command on the TX Matrix backup Routing Engine, all the T640 backup Routing Engines that are connected to it are upgraded to the same version of software.

Likewise, when you issue the **request system software add *package-name*** command on a TX Matrix Plus master Routing Engine, all the T1600 or T4000 master Routing Engines that are connected to it are upgraded to the same version of software. If you issue the same command on the TX Matrix Plus backup Routing Engine, all the T1600 or T4000 backup Routing Engines that are connected to it are upgraded to the same version of software.

Before installing software on a device that has one or more custom YANG data models added to it, back up and remove the configuration data corresponding to the custom YANG data models from the active configuration. For more information see "Managing YANG Packages and Configurations During a Software Upgrade or Downgrade" on page 150.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

Format for Specifying Filenames and URLs in Junos OS CLI Commands

request system software delete | **641**

request system software rollback | **654**

request system storage cleanup | **679**

Installing Software Packages on QFX Series Devices | **182**

Upgrading Software on a QFabric System

Managing Satellite Software Upgrade Groups in a Junos Fusion

request system software add (Maintenance) | **636**

Routing Matrix with a TX Matrix Plus Router Solutions Page

List of Sample Output

request system software add validate on page 624

request system software add /var/tmp/ no-validate on page 625
request system software add no-copy no-validate reboot on page 626
request system software add validate-on-host on page 626
request system software add (Mixed EX4200 and EX4500 Virtual Chassis) on page 628
request system software add component all (QFabric Systems) on page 628
request system software add upgrade-group (Junos Fusion) on page 628
request system software add restart (Junos OS Evolved for fixed form platforms) on page 628
request system software add restart (Junos OS Evolved for chassis-based platforms) on page 632

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 ...
```

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 restart (Junos OS Evolved for fixed form platforms)

`user@device>` **request system software add restart /var/tmp/qfx-ms-fixed-1-target1-ifmanarpcos.iso**

```
Adding software images. This process can take several minutes. Please be patient...
Download and Validate in Progress
re0: Starting upgrade : /var/tmp/qfx-ms-fixed-1-target1-ifmanarpcos.iso
re0: Single RE upgrade detected.
re0: Installing IMA keys of the incoming ISO image...
re0: Validating existing configs. See /var/log/validation_config.log for config
validation logs.
```

```
re0: Validation Passed. Going ahead with Installation
re0: Starting the installation...
re0: Copying files to
//soft/junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190523235333-1...
re0: Running post install commands...
re0: Post install sequence was successful.
re0: Installation of image
junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190523235333-1 done.
re0: Boot version is now
'junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190523235333-1'
Image validation and installation succeeded. Restarting Applications.

*** Restart Apps list ***
arpd
ifmand
cosd
Activating active instance of app arpd on node re0
Activating active instance of app ifmand on node re0
Activating active instance of app cosd on node re0
Image activation succeeded for arpd on node re0
Stopping active instance of app arpd on node re0
Image activation succeeded for ifmand on node re0
Stopping active instance of app ifmand on node re0
Image activation succeeded for cosd on node re0
Stopping active instance of app cosd on node re0
App arpd stopped on node re0
Starting active instance of app arpd on node re0
App ifmand stopped on node re0
Starting active instance of app ifmand on node re0
App cosd stopped on node re0
Starting active instance of app cosd on node re0
App cosd started/restarted on node re0
App ifmand started/restarted on node re0
App arpd started/restarted on node re0
*** Restart Summary ***
   *** Restart Success ***
   arpd
   ifmand
   cosd
Please check the status of applications using 'show system alarms'
```

`user@device>` **request system software add restart /var/tmp/qfx-ms-fixed-1-target2-ifmanarpcossysman.iso**

```
Adding software images. This process can take several minutes. Please be patient...
Download and Validate in Progress
re0: Starting upgrade : /var/tmp/qfx-ms-fixed-1-target2-ifmanarpcossysman.iso
re0: Single RE upgrade detected.
re0: Installing IMA keys of the incoming ISO image...
re0: Validating existing configs. See /var/log/validation_config.log for config
validation logs.
re0: Validation Passed. Going ahead with Installation
re0: Starting the installation...
re0: Copying files to
//soft/junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190523235731-1...
re0: Running post install commands...
re0: Post install sequence was successful.
re0: Installation of image
junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190523235731-1 done.
re0: Boot version is now
'junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190523235731-1'
Image validation and installation succeeded. Restarting Applications.

*** Restart Apps list ***
sysman
arpd
ifmand
cosd
Activating active instance of app sysman on node re0
Image activation succeeded for sysman on node re0
Restarting active instance of app sysman on node re0
App sysman started/restarted on node re0
Activating active instance of app arpd on node re0
Activating active instance of app ifmand on node re0
Activating active instance of app cosd on node re0
Image activation succeeded for arpd on node re0
Stopping active instance of app arpd on node re0
Image activation succeeded for ifmand on node re0
Stopping active instance of app ifmand on node re0
Image activation succeeded for cosd on node re0
Stopping active instance of app cosd on node re0
App cosd stopped on node re0
Starting active instance of app cosd on node re0
App ifmand stopped on node re0
Starting active instance of app ifmand on node re0
App arpd stopped on node re0
Starting active instance of app arpd on node re0
App cosd started/restarted on node re0
```

```
App ifmand started/restarted on node re0
App arpd started/restarted on node re0
*** Restart Summary ***
   *** Restart Success ***
   sysman
   arpd
   ifmand
   cosd
Please check the status of applications using 'show system alarms'
```

user@device> **request system software add restart /var/tmp/qfx-ms-fixed-1-target3-ifmanarpcossysmanimgdorchd.iso**

```
ifmanarpcossysmanimgdor
Adding software images. This process can take several minutes. Please be patient...
Download and Validate in Progress
re0: Starting upgrade :
/var/tmp/qfx-ms-fixed-1-target3-ifmanarpcossysmanimgdorchd.iso
re0: Single RE upgrade detected.
re0: Installing IMA keys of the incoming ISO image...
re0: Validating existing configs. See /var/log/validation_config.log for config
validation logs.
re0: Validation Passed. Going ahead with Installation
re0: Starting the installation...
re0: Copying files to
//soft/junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190524000025...
re0: Running post install commands...
re0: Post install sequence was successful.
re0: Installation of image
junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190524000025-gsanka-1
 done.
re0: Boot version is now
'junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190524000025'
Image validation and installation succeeded. Restarting Applications.

*** Restart Apps list ***
sysman
arpd
ifmand
cosd
imgd
orchestratord
Activating active instance of app sysman on node re0
```

```
Image activation succeeded for sysman on node re0
Restarting active instance of app sysman on node re0
App sysman started/restarted on node re0
Activating active instance of app arpd on node re0
Activating active instance of app ifmand on node re0
Activating active instance of app cosd on node re0
Image activation succeeded for arpd on node re0
Stopping active instance of app arpd on node re0
Image activation succeeded for ifmand on node re0
Stopping active instance of app ifmand on node re0
Image activation succeeded for cosd on node re0
Stopping active instance of app cosd on node re0
App arpd stopped on node re0
Starting active instance of app arpd on node re0
App ifmand stopped on node re0
Starting active instance of app ifmand on node re0
App cosd stopped on node re0
Starting active instance of app cosd on node re0
App cosd started/restarted on node re0
App arpd started/restarted on node re0
App ifmand started/restarted on node re0
Activating active instance of app imgd on node re0
Activating active instance of app orchestratord on node re0
Image activation succeeded for imgd on node re0
Image activation succeeded for orchestratord on node re0
Restarting active instance of app imgd on node re0
Restarting active instance of app orchestratord on node re0
```

request system software add restart (Junos OS Evolved for chassis-based platforms)

user@host> **request system software add a.iso restart**

```
Adding software images. This process can take several minutes. Please be patient...



*** List of Offlined FPCs ***
FPC0
FPC1
FPC2

Perform online for above FPCs (y/n) ? y

*** Incompatible FPCs ***
```

```
FPC0
FPC1

Warning: Perform offline for above incompatible FPCs

Perform offline for above incompatible FPCs (y/n) ? n

Aborting Software Upgrade

user@host>
```

```
Adding software images. This process can take several minutes. Please be patient...


*** Unsupported frus list ***
fpc0

Perform offline for above unsupported FPCs

Offline the incompatible FRUs before proceeding for upgrade?
Enter yes to proceed with offline for incompatible frus or no to abort the upgrade..
 Proceed? [yes,no] (no) yes


*** Restart Apps list ***
sysman
arpd
mgd
orchestratord

*** Offlining of Unsupported frus may take few mins ***
fpc0 has been successfully offlined

*** Offlining of Unsupported frus are done ***
Activating active instance of app sysman on node re0
Activating active instance of app sysman on node re1
Activating active instance of app sysman on node fpc0
Image activation succeeded for sysman on node re0
Restarting active instance of app sysman on node re0
Image activation succeeded for sysman on node re1
Restarting active instance of app sysman on node re1
```

```
Image activation failed on node fpc0
App does not exist sysman on node fpc0
App sysman started/restarted on node re0
App sysman started/restarted on node re1
Activating active instance of app arpd on node re0
Image activation succeeded for arpd on node re0
Stopping active instance of app arpd on node re0
App arpd stopped on node re0
Starting active instance of app arpd on node re0
App arpd started/restarted on node re0
Activating active instance of app mgd on node re0
Activating active instance of app mgd on node re1
Activating active instance of app orchestratord on node re0
Image activation succeeded for mgd on node re0
Image activation succeeded for mgd on node re1
Image activation succeeded for orchestratord on node re0
Activating new version of the software on node fpc0
Activating new version of the software on node re0
Activating new version of the software on node re1
Image activation failed on node fpc0
Image activation succeeded on node re0
Image activation succeeded on node re1

*** Onlining of Unsupported frus may take few mins ***
fpc0 could not  to be onlined

*** Onlining of Unsupported frus are done ***
Restarting active instance of app mgd on node re0
Restarting active instance of app mgd on node re1
Restarting active instance of app orchestratord on node re0
*** Restart Summary ***
   *** Restart Success ***
   sysman
   sysman
   arpd
   *** Restart Failure ***
   sysman
Please restart the failed applications
Please check the status of applications using 'show system alarms'

WARNING: cli has been replaced by an updated version:
CLI release 20190916.173330_rbu-builder.r1055817 built by rbu-builder on 2019-09-16
 18:02:02 UTC
Restart cli using the new version ? [yes,no] (yes) yes
```

```
Restarting cli ...
{master}
root@scapa-infra-01-re0>
```

request system software add (Maintenance)

Syntax

```
request system software add package-name
```

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

- **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

request system reboot (SRX Series)

request system software configuration-backup

Syntax

```
request system software configuration-backup (path)
```

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

configure—To enter configuration mode, but other required privilege levels depend on where the statement is located in the configuration hierarchy.

RELATED DOCUMENTATION

request system software configuration-restore | 639

List of Sample Output

request system software configuration-backup on page 637

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

```
request system software configuration-restore (path)
```

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

request system software configuration-backup | 637

Performing a QFabric System Recovery Installation on the Director Group

List of Sample Output

request system software configuration-restore on page 639

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

```
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 (Junos OS Evolved)

```
request system software delete
<force>
<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

package-name—(Only for Junos OS Evolved) Name of the Junos OS Evolved package running on the device. You can see this package name by using the **request system software list** command. Type the package-name explicitly and do not use the tab key to auto-complete the command.

software-package—(Not available on Junos OS Evolved) Software package or bundle name. You can see this software package name by using the **show system software** command. Type the software package name explicitly and do not use the tab key to auto-complete the command.

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*]—(M Series, MX Series, and T Series routers only) (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 | 613
request system software rollback | 654
request system software validate | 668
Routing Matrix with a TX Matrix Plus Router Solutions Page

List of Sample Output

request system software delete jdocs on page 644
request system software delete (Junos OS Evolved) on page 645

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> **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 delete (Junos OS Evolved)

user@host> **request system software delete junos-evo-install-qfx-fixed-x86-64-18.3I20180911102422**

```
Removing version 'junos-evo-install-qfx-fixed-x86-64-18.3I20180911102422'.
Software ... done.
Data ... done.
Version 'junos-evo-evo-qfx-fixed-x86-64-18.3I20180911102422' removed successfully.
```

request system software delete-backup

Syntax

```
request system software delete-backup
```

Release Information

Command introduced in Junos OS Release 9.6R1 for SRX Series devices.

Description

Junos OS keeps a backup image of the software that was previously installed so that you can downgrade to that version of the software if necessary. You can delete this backup image. If you delete this image, you cannot downgrade to this particular version of the software. Note that this command does not perform any function on SRX Series devices with dual-root partitioning scheme (units that shipped with Junos OS 10.0 version or later).

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system software delete | 641

List of Sample Output

request system software system-backup on page 646

Sample Output

request system software system-backup

`user@host>` **request system software delete-backup**

```
Delete backup system software package [yes,no]
```

request system software download

Syntax (QFabric System)

```
request system software download path package-name
```

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

path—Location where the software package is located. For example:

- **/pbdata/packages/*package-name***—For a software package that is being installed from a local directory on the 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.
  - **scp**—Secure copy (available only for Canada and U.S. version).
    Use **scp://*hostname/pathname/package-name***. To specify authentication credentials, use **scp://<*username*>:<*password*>@*hostname/pathname/package-name***.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system software add | 613
request system software delete | 641
request system software rollback | 654
request system storage cleanup | 679
Installing Software Packages on QFX Series Devices | 182

Upgrading Software on a QFabric System

List of Sample Output

request system software download on page 648

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.

> **NOTE:** Rolling back to a previously created restore-point might disrupt traffic, as both Director devices reboot from the restore-point partition.

Required Privilege Level

configure

RELATED DOCUMENTATION

request system software restore-point | 651

List of Sample Output

request system software recover-from-restore-point on page 649

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
 LogVol00 has the root filesystem
 Found Restore-Point:  Fri Aug 15 07:42:39 UTC 2014 /dev/VolGroup00/LogVol03
 Mounting restore-volume LogVol03
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

```
request system software restore-point
```

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

request system software recover-from-restore-point | 649

List of Sample Output

request system software restore-point on page 651

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/VolGroup00/LogVol03
Restore volume selected is /dev/VolGroup00/LogVol03
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/VolGroup00/LogVol03: 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/VolGroup00/LogVol03
waiting 10 mins for for peer dg to finish
Restore point creation success on both DGs
```

request system software rollback

List of Syntax

Syntax

```
request system software rollback
```

Syntax (EX Series Switches)

```
request system software rollback
<all-members>
<local>
<member member-id>
<reboot>
```

Syntax (TX Matrix Router)

```
request system software rollback
<lcc number | scc>
<reboot>
```

Syntax (TX Matrix Plus Router)

```
request system software rollback
<lcc number | sfc number>
<reboot>
```

Syntax (MX Series Router)

```
request system software rollback
<all-members>
<device-alias alias-name>
<local>
<member member-id
```

```
<reboot>
<satellite slot-id>
<upgrade-group [all |upgrade-group-name]>
```

Syntax (Junos OS Evolved)

```
request system software rollback
<no-validate>
<package-name>
<reboot>
<validate>
<with-old-snapshot-config>
```

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 35.

validate and **no-validate** options introduced for Junos OS Evolved Release 18.3R1.
package-name ***version*** option introduced for Junos OS Evolved Release 18.3R1.
with-old-snapshot-config option introduced for Junos OS Evolved Release 18.3R1.

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.
- On Junos OS Evolved, the **reboot** command is required in order to complete the rollback.

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.

no-validate | validate—(Only for Junos OS Evolved) Check compatibility with current configuration, yes or no.

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.

package-name *version*—(Junos OS Evolved only) Select any installed version for the rollback. The **request system software rollback** command uses the version instead of the package-name. you can see the available versions by using the **show system software list** command. If a version is not specified, the system rolls back to the default rollback version (the one with the '<' before it on the **show system software list** command output). You can specify any previous Junos OS Evolved release as long as it is not the one that is currently running or the rollback version.

reboot—(Optional) For Junos OS 12.3 and later, the system reboots automatically to complete the rollback. However, for Junos OS Evolved, you must explicitly specify the **reboot** option 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.

validate | no-validate—(Junos OS Evolved only).

with-old-snapshot-config—(Optional) (Junos OS Evolved only) Rolls back system to the specified version with the old snapshot of the configuration used in that version. Otherwise, the rollback, by default, takes the current configuration.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system software abort

request system software add | 613

request system software delete | 641

request system software validate | 668

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 658

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

```
request system software rollback <node-id>
```

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 upgraded FreeBSD 11.x (supported in Junos OS Release 17.4R1) 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

node-id—Identification number of the chassis cluster node. It can be 0 or 1.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system reboot (SRX Series)
Upgrading Junos OS with Upgraded FreeBSD | **282**
Release Information for Junos OS with Upgraded FreeBSD | **35**
What Is Junos OS with Upgraded FreeBSD? | **34**

request system software sync

Syntax

```
request system software sync (current | rollback)
```

Release Information

Command introduced in Junos OS Evolved Release 18.3R1.

Description

Sync software from master node to other node and reboot other node.

Options

current | rollback—Specify which software version (current or rollback) to sync to other node:

- For the **current** option, system syncs the current version to the other node and reboots with that version.
- For the **rollback** option, the system syncs the rollback version to the other node.

Additional Information

To see what the software version on the device are, use the **request system software list** command.

Required Privilege Level

view

RELATED DOCUMENTATION

request system software rollback | 654

List of Sample Output

request system software sync current on page 661
request system software sync rollback on page 664

Sample Output

request system software sync current

`user@host>` **request system software sync current**

```
Cleanup old software versions on re1
    Removing version junos-evo-install-qfx-x86-64-16.2I20170625192110...
    Done.
    Transfer software version files for
junos-evo-install-qfx-x86-64-16.2I20170626030132 to node re1...
    Setting up software version files for
junos-evo-install-qfx-x86-64-16.2I20170626030132 on re1
        /soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-re-64b.bin
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_uImage-ptxpmb-p2020.bin
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-jnx_fpc_0bf9.dtb
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-jnx_fpc_0bfa.dtb
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-jnx_spmb_0c2b.dtb
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-ptx21k-rcb.dtb
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-ptx5k-mtrcb.dtb
linked
       /soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-ptx5k-rcb.dtb
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_uImage-ptxpmb_p2020.dtb
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_uImage-ptxpmb_p2020_spmb.dtb
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_ccd-ptx-fpc64_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_ccd-ptx-fpc_Yocto_1.8_p2020.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_ccd-ptx-re64_Yocto_1.8_x86_64.fs
 linked
```

```
/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_dev_Yocto_1.8_p2020.fs
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_dev_Yocto_1.8_x86_64.fs
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_dev_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_initramfs_Yocto_1.8_p2020.fs
 copied

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_initrd_Yocto_1.8_p2020.fs
copied

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_initrd_Yocto_1.8_x86_64.fs
 copied

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_initrd_Yocto_1.8_x86_64_fpc.fs
 copied

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_java_Yocto_1.8_x86_64.fs
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_jimbase_Yocto_1.8_p2020.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_jimbase_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_jimbase_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_modules_Yocto_1.8_p2020.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_modules_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_modules_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_osbase_Yocto_1.8_p2020.fs
```

```
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_osbase_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_osbase_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_perl-5.20.0_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_python-2.7_Yocto_1.8_p2020.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_python-2.7_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_python-2.7_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_python-3.3_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_zookeeper_Yocto_1.8_x86_64.fs
 linked
    Set version junos-evo-install-qfx-x86-64-16.2I20170626030132_tchittar as
nextboot
    Installing for i386-pc platform.
    Installation finished. No error reported.
    Boot version is now 'junos-evo-install-qfx-x86-64-16.2I20170626030132'
    Rebooting re1
    Shutdown at Mon Jun 26 03:24:08 2017
```

request system software sync rollback

user@host> **request system software sync rollback**

```
Cleanup old software versions on re1
    Removing version junos-evo-install-qfx-x86-64-16.2I20170625192110...
    Done.
    Transfer software version files for
junos-evo-install-qfx-x86-64-16.2I20170626030132 to node re1...
    Setting up software version files for
junos-evo-install-qfx-x86-64-16.2I20170626030132 on re1
```

```
       /soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-re-64b.bin
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_uImage-ptxpmb-p2020.bin
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-jnx_fpc_0bf9.dtb
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-jnx_fpc_0bfa.dtb
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-jnx_spmb_0c2b.dtb
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-ptx21k-rcb.dtb
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-ptx5k-mtrcb.dtb
linked
       /soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_bzImage-ptx5k-rcb.dtb
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_uImage-ptxpmb_p2020.dtb
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_uImage-ptxpmb_p2020_spmb.dtb
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_ccd-ptx-fpc64_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_ccd-ptx-fpc_Yocto_1.8_p2020.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_ccd-ptx-re64_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_dev_Yocto_1.8_p2020.fs
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_dev_Yocto_1.8_x86_64.fs
linked
```

```
/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_dev_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_initramfs_Yocto_1.8_p2020.fs
 copied

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_initrd_Yocto_1.8_p2020.fs
copied

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_initrd_Yocto_1.8_x86_64.fs
 copied

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_initrd_Yocto_1.8_x86_64_fpc.fs
 copied

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_java_Yocto_1.8_x86_64.fs
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_jimbase_Yocto_1.8_p2020.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_jimbase_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_jimbase_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_modules_Yocto_1.8_p2020.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_modules_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_modules_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_osbase_Yocto_1.8_p2020.fs
linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_osbase_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_osbase_Yocto_1.8_x86_64_fpc.fs
 linked
```

```
/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_perl-5.20.0_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_python-2.7_Yocto_1.8_p2020.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_python-2.7_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_python-2.7_Yocto_1.8_x86_64_fpc.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_python-3.3_Yocto_1.8_x86_64.fs
 linked

/soft/junos-evo-install-qfx-x86-64-16.2I20170626030132_zookeeper_Yocto_1.8_x86_64.fs
 linked
    Set version junos-evo-install-qfx-x86-64-16.2I20170626030132 as nextboot
    Installing for i386-pc platform.
    Installation finished. No error reported.
    Boot version is now 'junos-evo-install-qfx-x86-64-16.2I20170626030132'
    Rebooting re1
    Shutdown at Mon Jun 26 03:24:08 2017
```

request system software validate

List of Syntax

Syntax

```
request system software validate package-name
<on (host host <username username> | routing-engine routing-engine)>
<set [package-name package-name]>
<upgrade-with-config>
<upgrade-with-config-format format>
```

Syntax (TX Matrix Router)

```
request system software validate package-name
<lcc number | scc>
<on (host host <username username> | routing-engine routing-engine)>
<set [package-name package-name]>
<upgrade-with-config>
<upgrade-with-config-format format>
```

Syntax (TX Matrix Plus Router)

```
request system software validate package-name
<lcc number | sfc number>
<on (host host <username username> | routing-engine routing-engine)>
<set [package-name package-name]>
<upgrade-with-config>
<upgrade-with-config-format format>
```

Syntax (MX Series Router)

```
request system software validate <package-name>
<member member-id>
<on (host host <username username> | routing-engine routing-engine)>
<set [package-name package-name]>
<upgrade-with-config>
```

```
<upgrade-with-config-format format>
```

Syntax (Junos OS Evolved)

```
request system software validate package-name
```

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 [*package-name package-name*] option added in Junos OS Release 12.2 for M Series, MX Series, T Series routers.
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.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
on (host *host* <username *username*> | routing-engine *routing-engine*) 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.
set [*package-name package-name*] option deprecated in Junos OS Evolved 18.3R1.
restart option added in Junos OS Evolved Release 18.3R1.

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.

restart—(For Junos OS Evolved only) (Optional). Verify the new software configuration compatibility. This option verifies the ISSU services impact with new software. It emits the list of services which may get restarted when user issues this command.

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 fomat, **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 validate in-service-upgrade
request system software abort
request system software add | **613**
request system software delete | **641**
request system software rollback | **654**
Routing Matrix with a TX Matrix Plus Router Solutions Page

List of Sample Output
request system software validate (Successful Case) on page 671
request system software validate (Failure Case) on page 672

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

```
request system software validate on
<host host-name [ username user-name ]>
<routing-engine (re0 | re1)>
```

Release Information

Command introduced in Junos OS Release 15.1 for supported platforms. See Feature Explorer.

Description

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.

This command validates the current configuration on a Routing Engine that is not running Junos OS with upgraded FreeBSD or a remote host.

Options

The specific options available are:

host *host-name* [username *user-name*]—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.

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.

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

request system reboot (Junos OS with Upgraded FreeBSD) | 580

List of Sample Output

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
 vlans': not found
mgd: commit complete
Validation succeeded
```

request system software validate-restart

Syntax

```
request system software validate-restart package-name
```

Release Information

Command introduced in Junos OS Evolved Release 19.1R2.

Description

Verify new package with current configuration and get ISSU impact. Use this command before you run **request system software add restart**.

Options

package-name—Path and filename of the software package to be installed.

Required Privilege Level

view

RELATED DOCUMENTATION

request system software add | 613

List of Sample Output

Sample Output

request system software validate-restart (fixed form platforms)

`user@host>` **request system software validate-restart /var/tmp/qfx-ms-fixed-1-target1-ifmanarpcos.iso validate-restart**

```
Validating software image and getting ISSU services impact
/var/tmp/qfx-ms-fixed-1-target1-ifmanarpcos.iso...
Download and Validate in Progress
re0: Starting software validation for
/var/tmp/qfx-ms-fixed-1-target1-ifmanarpcos.iso
re0: This process can take several minutes. Please be patient...
```

```
[ 2631.326645] ICMPv6: RA: ndisc_router_discovery failed to add default route
[ 2648.292906] disk_mgmt_tool.py[7812]: 2019-05-24 18:31:29 INFO: /var  is running
 high on usage,   74% used of  15G
re0: Installing IMA keys of the incoming ISO image...
re0: Validating existing configs. See /var/log/validation_config.log for config
validation logs.
re0: Config validation Success. Getting ISSU impact report. Can take few more
minutes...
[ 2717.155661] ICMPv6: RA: ndisc_router_discovery failed to add default route
re0: Removing version
junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190523235333-gsanka-1...
re0: Validation for version
junos-evo-install-qfx-ms-fixed-x86-64-19.2R1-20190522.4-EVOI20190523235333-gsanka-1
 completed
Image validation succeeded. ISSU impact report:

*** Restart Apps list ***
arpd
ifmand
cosd
Image validation succeeded.
```

request system software validate-restart (chassis-based platforms)

`user@host>` **request system software validate-restart /var/tmp/a.iso validate-restart**

```
*** List of Offlined FPCs ***
FPC0
FPC1
FPC2

Please perform online for above FPCs.


*** List of Incompatible FPCs ***

FPC0
FPC1

Warning: Please perform offline for above FPCs.


*** List of Incompatible configurations ***
```

```
"system services subscriber-management enable"

Warning: Please perform deactivation of the above configurations.


*** Restart Apps list ***
sysman
ifmand
arpd
Image validation succeeded.

user@host>
```

request system storage cleanup

List of Syntax

Syntax

```
request system storage cleanup
<dry-run>
<no-confirm>
<re0 | re1 | routing-engine (backup | both | local | master | other)>
```

Syntax (EX Series Switches)

```
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]>
```

Syntax (MX Series Router)

```
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]>
```

Syntax (QFX Series)

```
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)>
```

Syntax (SRX Series)

```
request system storage cleanup
<dry-run>
<no-confirm>
<re0 | re1 | routing-engine (backup | both | local | master | other)>
```

Syntax (Junos OS Evolved)

```
request system storage cleanup (dry-run | force-deep | no-confirm)
```

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.

The Junos OS Evolved implementation of the **request system storage cleanup** command is slightly different from the implementation on Junos OS:

- The user is prompted to specify the **dry-run** option:

  ```
  Please check the list of files to be deleted using the dry-run option.
  Continue anyway without checking? [yes,no] (yes)
  ```

  The command cleans up any ISO files on the system, rotates syslogs, and clears trace files. It does not remove user-created files.

- In Junos OS Evolved, the system computes the available space and emits o/p on console for reference.
- When **request system storage cleanup** is executed, Junos OS Evolved displays the types of files that are being deleted. See the Sample Output section below for an example.

> **NOTE:** The **request system storage cleanup | display xml rpc** command displays different XML tags for different file types. In Junos OS, only the **file** tag is displayed for all types of files. For more information about the differences between Junos OS and Junos OS Evolved, see *How Junos OS Evolved Differs from Junos OS*.

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).

force-deep—(Junos OS Evolved only) Deep clean all temporary files and rotate logs. This option cleans up all the user-created files under **/tmp** and **/var/tmp** directories.

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

Output Fields

Table 29 on page 683 describes the output fields for the **request system storage cleanup** command. Output fields are listed in the approximate order in which they appear.

Table 29: 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**.

Table 29: request system storage cleanup Output Fields *(continued)*

Field Name	Field Description
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 (Junos OS Evolved)

`user@host>` **request system storage cleanup**

```
Please check the list of files to be deleted using the dry-run option.
Continue anyway without checking? [yes,no] (no) yes

  -------------------------------
  node:  re0
  -------------------------------
  Clearing all core files


  Clearing all local host core files and files from /var/log/watchdog


  Clearing node specific core files


  Clearing FPC log  files
```

```
Clearing logical-systems log files


Clearing journal logs


Clearing all /var/log/* files

  Size      Date              Name
  4.0K  Thu Nov 21 11:25  /var/log/__policy_names_rpdc__
  4.0K  Thu Nov 21 11:27  /var/log/__policy_names_rpdn__
  8.0K  Thu Nov 21 11:27  /var/log/alarm-mgmtd
  4.0K  Thu Nov 21 11:24  /var/log/boot_init.log
  4.0K  Thu Nov 21 11:25  /var/log/charonctl_trace.log
  4.0K  Thu Nov 21 11:23  /var/log/check_restore_recovery_bios.log
  4.0K  Thu Nov 21 11:29  /var/log/clksynced.log
   80K  Thu Nov 21 11:27  /var/log/configd-streamer.log
  4.0K  Thu Nov 21 11:25  /var/log/core_mgr.log
     0  Thu Nov 21 11:24  /var/log/cscript.log
  4.0K  Thu Nov 21 11:29  /var/log/ddosd.log
  4.0K  Thu Nov 21 11:25  /var/log/disk_mgmt
  4.0K  Thu Nov 21 11:29  /var/log/eth_linkmon.log
  960K  Thu Nov 21 11:48  /var/log/evo-cda-zx.log
  2.9M  Thu Nov 21 15:27  /var/log/evo_dns_relay.log
   12K  Thu Nov 21 11:29  /var/log/evoinit.log
  8.0K  Thu Nov 21 15:24  /var/log/fibd-proxy.log
  4.0K  Thu Nov 21 11:25  /var/log/imgd.log
  4.0K  Thu Nov 21 11:28  /var/log/interactive-commands
   16K  Thu Nov 21 11:29  /var/log/kfirewall-agent.log
     0  Thu Nov 21 11:27  /var/log/mcelog.log
   12K  Thu Nov 21 11:29  /var/log/mem_mgmt.log
     0  Thu Nov 21 11:26  /var/log/mem_monitor.dat
   84K  Thu Nov 21 15:27  /var/log/messages
     0  Thu Nov 21 11:26  /var/log/mgd-api
  4.0K  Thu Nov 21 11:25  /var/log/mgmt-ethd-helper.log
   44K  Thu Nov 21 11:28  /var/log/mib2d
  4.0K  Thu Nov 21 11:25  /var/log/mstr.log
   84K  Thu Nov 21 11:28  /var/log/packetio-cout.log
  1.4M  Thu Nov 21 11:48  /var/log/picd.log
  4.0K  Thu Nov 21 11:29  /var/log/platform_mon.log
  4.0K  Thu Nov 21 11:29  /var/log/ptp_fpga.log
  4.0K  Thu Nov 21 15:27  /var/log/security
  4.0K  Thu Nov 21 11:29  /var/log/set_mgmt_mac.log
```

```
     92K  Thu Nov 21 15:25  /var/log/sinet.log
     16K  Thu Nov 21 11:28  /var/log/snmpd
    4.0K  Thu Nov 21 11:29  /var/log/ss.log
    4.0K  Thu Nov 21 11:24  /var/log/ssh-key-utils.log
    4.0K  Thu Nov 21 11:28  /var/log/sshd_lua.log
    4.0K  Thu Nov 21 11:29  /var/log/sysconfig.log
    4.0K  Thu Nov 21 11:29  /var/log/sysstart.log
    4.0K  Thu Nov 21 11:28  /var/log/system-events
     20K  Thu Nov 21 11:24  /var/log/uswitch.log
     24K  Thu Nov 21 11:23  /var/log/uswitch.log.prev
    4.0K  Thu Nov 21 11:26  /var/log/validator_debug.log
    8.0K  Thu Nov 21 15:24  /var/log/wtmp
    4.2M  Thu Nov 21 11:48  /var/log/zookeeper--server-re0.log
    4.0K  Thu Nov 21 11:25  /var/log/zookeeper--server-re0.out
    4.0K  Thu Nov 21 11:27  /var/log/ztp.log

  Clearing all JSON files in /var/log/objmon


  Cleared traces for application all node all pid all


  Clearing SI traces


  Removing any ISO files in /data

  Current list of software versions installed

Active boot device is primary : /dev/vda
List of installed version(s) :

    '-' running version
    '>' next boot version after upgrade/downgrade
    '<' rollback boot version

 -   junos-evo-install-ptx-fixed-x86-64-20.1-201911201458.0-EVO - [2019-11-21
11:23:21]

  Current space available in /soft: 14752420 K and /data: 2788736 K
```

request system storage cleanup dry-run (Junos OS Evolved)

`user@host>` **request system storage cleanup dry-run**

```
  -------------------------------
  node:  re0
  -------------------------------
  List of all core files to be cleared:


  List of local_host core files to be cleared:


  List of core sub directory files to be cleared:


  List of log files from FPCs to be cleared:


  List of log files from logical systems to be cleared:


  Clears all App logs, App traces, App SI traces and App core files from /var/log/*,
 /var/log/traces/*, /var/log/si_traces/* and /var/core/*


  Clears all JSON files in /var/log/objmon/ during cleanup


  List of ISO files from /data partition to be cleared:

  Current list of software versions installed
  Removes older software versions - Minimum two versions would be left around

Active boot device is primary : /dev/vda
List of installed version(s) :

    '-' running version
    '>' next boot version after upgrade/downgrade
    '<' rollback boot version

 -   junos-evo-install-ptx-fixed-x86-64-20.1-201911201458.0-EVO - [2019-11-21
11:23:21]
```

request system storage cleanup force-deep (Junos OS Evolved)

`user@host>` **request system storage cleanup force-deep**

```
Please check the list of files to be deleted using the dry-run option.
 Continue anyway without checking? [yes,no] (no) yes

-------------------------------
node: re0
-------------------------------
........
========= Start cleanup now ==========
=== Start removing other logs, traces, core files  ===
Clearing core files
Clearing FPC logs
Clearing logical-systems logs
=== Clearing journal logs ===
Clearing log: /var/log/RE_journal.log
Clearing log: /var/log/RE_journal_boot.log
Clearing log: /var/log/alarm-mgmtd
Clearing log: /var/log/appDemo_stdout
Clearing log: /var/log/charonctl_trace.log
Clearing log: /var/log/configd-streamer.log
Clearing log: /var/log/core_mgr.log
Clearing log: /var/log/cscript.log
Clearing log: /var/log/eth_linkmon.log
Clearing log: /var/log/evo-cda-zx.log
Clearing log: /var/log/evoinit.log
Clearing log: /var/log/fibd-proxy.log
Clearing log: /var/log/i2ctrace.log
Clearing log: /var/log/i2ctrace_spmb0.log
Clearing log: /var/log/i2ctrace_spmb1.log
Clearing log: /var/log/icmpd.log
Clearing log: /var/log/ifinfo.log
Clearing log: /var/log/imgd_svr.log
Clearing log: /var/log/install
Clearing log: /var/log/interactive-commands
Clearing log: /var/log/jsd
Clearing log: /var/log/lastlog
Clearing log: /var/log/mcelog.log
Clearing log: /var/log/messages
Clearing log: /var/log/mgd-api
Clearing log: /var/log/mgmt-ethd-helper.log
Clearing log: /var/log/mib2d
Clearing log: /var/log/na-grpcd
Clearing log: /var/log/objmon_sync.json
Clearing log: /var/log/packetio-cout.log
Clearing log: /var/log/picd.log
```

```
Clearing log: /var/log/platform_mon.log
Clearing log: /var/log/policerd.log
Clearing log: /var/log/postinstall.log
Clearing log: /var/log/ptp_fpga.log
Clearing log: /var/log/reboot_node.log
Clearing log: /var/log/rollback.log
Clearing log: /var/log/security
Clearing log: /var/log/semctl.log
Clearing log: /var/log/set_mgmt_mac.log
Clearing log: /var/log/shutdown_complete.log
Clearing log: /var/log/sinet.log
Clearing log:
/var/log/smartd-attr-SFSA200GM3AA4TO_C_HC_636_JUN-000060139624B1000020.log
Clearing log:
/var/log/smartd-attr-SFSA200GM3AA4TO_C_HC_636_JUN-000060139624B1000022.log
Clearing log: /var/log/snmpd
Clearing log: /var/log/ss.log
Clearing log: /var/log/ssh-key-utils.log
Clearing log: /var/log/sshd_lua.log
Clearing log: /var/log/sysconfig.log
Clearing log: /var/log/sysman.conf
Clearing log: /var/log/system-events
Clearing log: /var/log/upgrade_master.log
Clearing log: /var/log/uswitch.log
Clearing log: /var/log/uswitch.log.prev
Clearing log: /var/log/validator_debug.log
Clearing log: /var/log/wtmp
Clearing log: /var/log/zookeeper--server-re.log
Clearing log: /var/log/zookeeper--server-re.out
Clearing log: /var/log/ztp.log
=== Clearing all traces ===
=== Clearing SI traces ===
=== Removing other logs, traces, core files completed ===
=== Started removing any ISO files in /data
=== Removing any ISO files in /data completed
=== Start Software versions cleanup ===
Removing older software versions except current and rollback
=== Software versions cleanup completed ===
========= Cleanup done ==========
Current space available in /soft: 12372572 K
Current space available in /data: 2638752 K
Cannot delete junos-evo-install-qfx-fixed-x86-64-18.3I20180906130134_mkamil - It
is the rollback version
Cannot delete junos-evo-install-qfx-fixed-x86-64-18.3-20180906.3 - It is the current
```

```
 version
Removing version junos-evo-install-qfx-x86-64-16.2I20180516093649...
Done.
```

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/rtsdb/if-rtsdb
```

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-rtsdb/env.lck
  12.0K Nov  7 15:41 /var/tmp/if-rtsdb/env.mem
 132.0K Nov  7 15:55 /var/tmp/if-rtsdb/shm_usr1.mem
2688.0K Nov  7 15:41 /var/tmp/if-rtsdb/shm_usr2.mem
2048.0K Nov  7 15:41 /var/tmp/if-rtsdb/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/rtsdb/if-rtsdb

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
  K 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-rtsdb/env.lck
  12.0K Nov  7 15:41 /var/tmp/if-rtsdb/env.mem
 132.0K Nov  7 15:55 /var/tmp/if-rtsdb/shm_usr1.mem
2688.0K Nov  7 15:41 /var/tmp/if-rtsdb/shm_usr2.mem
2048.0K Nov  7 15:41 /var/tmp/if-rtsdb/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/rtsdb/if-rtsdb
```

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-rtsdb/env.lck
  12.0K Nov  8 17:09 /var/tmp/if-rtsdb/env.mem
2688.0K Nov  8 17:09 /var/tmp/if-rtsdb/shm_usr1.mem
 132.0K Nov  8 17:09 /var/tmp/if-rtsdb/shm_usr2.mem
2048.0K Nov  8 17:09 /var/tmp/if-rtsdb/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/rtsdb/if-rtsdb

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-rtsdb/env.lck
  12.0K Nov  7 17:39 /var/tmp/if-rtsdb/env.mem
2688.0K Nov  7 17:39 /var/tmp/if-rtsdb/shm_usr1.mem
 132.0K Nov  7 17:40 /var/tmp/if-rtsdb/shm_usr2.mem
2048.0K Nov  7 17:39 /var/tmp/if-rtsdb/trace.mem
   155B Nov  7 17:40 /var/tmp/krt_gencfg_filter.txt
     0B Nov  7 17:39 /var/tmp/rtsdb/if-rtsdb
```

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/d0afda1e-0710-11e1-a1d0-00e081c5297e/install-11072011125930.log
/pbdata/export/rlogs/d0afda1e-0710-11e1-a1d0-00e081c5297e/install-11072011133211.log
/pbdata/export/rlogs/d0afda1e-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

```
request system storage cleanup <dry-run>
```

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

Cleaning Up Files with the CLI

List of Sample Output

request system storage cleanup dry-run on page 700
request system storage cleanup on page 701

Output Fields

Table 29 on page 683 describes the output fields for the **request system storage cleanup** command. Output fields are listed in the approximate order in which they appear.

Table 30: 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.

Table 30: request system storage cleanup Output Fields *(continued)*

Field Name	Field Description
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 | 590

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 | 80

Reverting to the Default Factory Configuration by Using the request system zeroize Command | 83

List of Sample Output

request system zeroize on page 705

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
...
```

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

Installing Junos OS on SRX Series Devices Using a USB Flash Drive | 268

List of Sample Output

Sample Output

show chassis hardware detail

user@host> **show chassis hardware detail**

```
Hardware inventory:
Item              Version  Part number  Serial number     Description
Chassis                                 BV4911AA0005      SRX240H2-POE
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                                                     FPC
```

```
  PIC 0                                                 16x GE Base PIC
Power Supply 0
```

show chassis usb storage

`user@host>` **show chassis usb storage**

```
USB Disabled
```

show system applications

Syntax

```
show system applications
<app app-name>
<brief>
<detail>
<node node-name>
```

Release Information

Command introduced in Junos OS Evolved Release 18.3R1.

Description

Display applications summary information in one of the following forms:

- Show all applications summary information for all nodes.
- Show the applications summary information for a specific application.
- Show the applications summary information for a specific node.

Options

app *app-name*—(Optional) Specify application name for which you want to display applications summary information.

brief—(Optional) Display brief output. This is the default format of display.

detail—(Optional) Display detailed output.

node *node-name*—(Optional) Specify node name for which you want to display applications summary information.

Required Privilege Level

view

RELATED DOCUMENTATION

request system application | 541

request node reboot (re0 | re1) | 540

request node (offline | online) | 539

List of Sample Output

show system applications on page 712

show system applications app application-name node node-name on page 714
show system applications node node-name detail on page 714

Output Fields

For a description of the output fields, see Table 31 on page 710. Output fields are listed in the approximate order in which they appear.

Table 31: show system applications Output Fields

Field Name	Description	Level
Applications Information	**Application**—Name of the application. **Node**—Name of the node the application is running on. **App State**—State of the application: online, offline, failed, or active. **App Weight**—A relative weight for multiple instances of the app across multiple nodes. The app instance with the higher wright is providing more functionality. **App Zookeeper Session**—Zeekeeper session ID.	all levels
Object Producer details	**Producer ID**—Identifies which production set the object is part of. **Epoch ID**—A number that identifies the current process that owns a production set. There can only be one owning process (active producer) that owns a production set at one time. The current owning process has an Epoch ID that is larger than any previous producer. **Production Topic**—Hierarchical string that represents the production set. **Producer State**—**active** or **standby**: • **active** indicates the application has production rights to modify the state in the production set. • **standy** means that the application is waiting to get the production right for the production set.	all levels
Description	A short description of the application, it also lists the systemd service file used for the application.	**detail**
Loaded	A systemd state that indicates if the application is loaded in the system or not.	**detail**
Run State from OS	A systemd state that indicates if the application is active or not.	**detail**
Main PID	Process identifier (PID) of the application.	**detail**

Table 31: show system applications Output Fields *(continued)*

Field Name	Description	Level
Command	Command line to launch the application.	**detail**
ID	Name of the application.	**detail**
Meta	Meta data for the application includes the following fields: **Bin path**—Path to application execution. **Log file**—Where logs go. **Working Dir**—Working directory. **Production Set**—Global or local production set. Values might be **shared** or **local**.	**detail**
Resource	Resource data for the application includes the following fields: **all nodes**—Is the application run on all nodes, **true** or **false**. **Max instances**—How many instances of the application are there. **Max instances per node**—How many instances of the application per node are there. **Run on startup**—Is the appliction launched at bootup, **true** or **false**. **Node attributes**—Typical node attributes are RE, FPC, MasterRE. You can see the node attributes by using the **show system node-attributes** command. **Node attribute match**—What is the node attribute required to launch this application on a node? For example, if this field has the output **re, Service file: lfmd**, it indicates that the process lfmd will be launched on a node that has the attribute RE.	**detail**
Failure	Failure data for the application includes the following fields: **Alarm color**—Which alarm to be raised on failure, or **none**. **Alarm ID**—The alarm ID. **Restart**—Whether to restart the application, **true** or **false**.	**detail**
Upgrade	**Upgrade parallely**—Options are **true** or **false**. **Upgrade restart node**—Options are **true** or **false**. **Upgrade style**—Option is **stop-start**.	**detail**

Table 31: show system applications Output Fields *(continued)*

Field Name	Description	Level
App-Exit	App-Exit data for the application includes the following fields: **Restart Supported**—True/false. When the apllication exits, should the application be restarted. **Restart Node**—True/false. When the apllication exits, should the node be rebooted. **Mark node spare**—When an application exits, should the node be marked spare.	**detail**

Sample Output

show system applications

user@host> **show system applications**

```
Applications Information:
Application              : bcmd_evo
Node                     : fpc0
App State               : offline
Object Producer details
Producer ID              : 0
  Epoch ID                 : 0

Applications Information:
Application              : ccdpfe
Node                     : fpc0
App State               : online
Object Producer details
Producer ID              : 576
  Epoch ID                 : 65
  Production Topic         : /Root/fpc0/ccdpfe/100143551468101228
  Producer State           : active

Applications Information:
Application              : cmdd
Node                     : fpc0
App State               : online
Object Producer details
Producer ID              : 570
  Epoch ID                 : 66
```

```
  Production Topic        : /Root/fpc0/cmdd/1099227235289688912
  Producer State          : active

...

Applications Information:
Application             : alarm-mgmtd
Node                    : re0
App State             : online
Object Producer details
Producer ID             : 26
  Epoch ID                : 1
  Production Topic        : /Root/alarm-mgmtd/2988563069668674039
  Producer State          : active

Applications Information:
Application             : alarmd
Node                    : re0
App State             : online
Object Producer details
Producer ID             : 377
  Epoch ID                : 30
  Production Topic        : /Root/alarmd/6512784671716237713
  Producer State          : active

Applications Information:
Application             : arpd
Node                    : re0
App State             : online
Object Producer details
Producer ID             : 396
  Epoch ID                : 41
  Production Topic        : /Root/arpd/14284058728950342139
  Producer State          : active

...

Applications Information:
Application             : alarm-mgmtd
Node                    : re1
App State             : online
Object Producer details
Producer ID             : 26
  Epoch ID                : 0
```

```
  Production Topic        : /Root/alarm-mgmtd/2988563069668674039
  Producer State          : standby

Applications Information:
Application             : bcmd_evo
Node                    : re1
App State              : offline
Object Producer details
Producer ID             : 0
  Epoch ID                : 0

Applications Information:
Application             : charonctl
Node                    : re1
App State              : online
Object Producer details
Producer ID             : 25
  Epoch ID                : 4
  Production Topic        : /Root/re1/charonctl/10854553120394604032
  Producer State          : active

...
```

show system applications app application-name node node-name

`user@host>` **show system applications app alarm-mgmtd node re1**

```
Applications Information:
Application             : alarm-mgmtd
Node                    : re1
App State              : online
Object Producer details
Producer ID             : 26
  Epoch ID                : 0
  Production Topic        : /Root/alarm-mgmtd/2988563069668674039
  Producer State          : standby
```

show system applications node node-name detail

`user@host>` **show system applications app cmdd detail**

```
[edit]

regress@dakkar# run show system applications app cmdd detail

Applications Information:

Application             : cmdd

Node                    : re0

App State               : online ready

App Weight              : 1

App Zookeeper Session   : 1000000934d000d

Object Producer details

Producer ID             : 50331736

  Epoch ID                : 47

  Production Topic        : /Root/re0/cmdd/3158206796014561683

  Producer State          : active

Description             : cmdd.service - "Command Daemon"

Loaded                  : loaded (/etc/systemd/system/cmdd.service;static;vendor
preset:enabled)
Run State from OS       : active (running) (Result: success) since Mon 2018-10-29
 05:02:24 PDT
Main PID                : 5814

Command                 : /usr/sbin/cmdd --app-name cmdd -I object_select
--shared-objects-mode 3
App Config Info

 ID                     : cmdd

 Meta

  Bin path              : /usr/sbin/cmdd
```

```
  Log file              : /var/log
  Working Dir           : /usr/sbin

  Production Set        : local

  Sysman Managed        : true

  Type Evo              : true

 Resource

  All nodes             : true

  Max instances         : 1

  Max instances per node: 1

  App Suite             : default,diags_default

  Run on startup        : true
  Node attributes       :

(Node attribute match  : *, Service file  : cmdd)

 Failure

  Alarm color           : red

  Restart Node          : false

  Mark node spare       : false

 Upgrade

  Upgrade parallely           : true

  Upgrade restart node        : false

  Upgrade style               : stop-start

 App-Exit

  Restart Supported           : true
```

```
  Restart Node                 : false
  Mark node spare              : false



[edit]

regress@dakkar#
```

```
Applications Information:
 Upgrade
  Prepare to upgrade notify   : false
  Unprepare to upgrade notify : false
  Node upgrade done notify    : false
  System upgrade done notify  : false
  Upgrade abort capable       : false
  Weight                      : 1
 App-Exit
  Restart Supported           : false
  Restart Node                : false
  Mark node spare             : false
```

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 Metro 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

ACX Series Autoinstallation Overview | 467
Before You Begin Autoinstallation on an ACX Series Universal Metro Router | 469
Autoinstallation Configuration of ACX Series Universal Metro Routers | 470
USB Autoinstallation on ACX Series Routers | 472
Verifying Autoinstallation on ACX Series Universal Metro Routers | 470
autoinstallation

List of Sample Output

show system autoinstallation status on page 719

Output Fields

Table 32 on page 719 describes the output fields for the **show system autoinstallation status** command. Output fields are listed in the approximate order in which they appear.

Table 32: show system autoinstallation status Output Fields

Field Name	Field Description

Autoinstallation status	Display autoinstallation status information:
	• **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.
	  ◦ **Name**—Name of the interface.
	  ◦ **State**—Interface state.
	• **Address acquisition**—Display IP address acquired and protocol used for acquisition upon startup.
	  ◦ **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 15.1X49-D35 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 20.1R1for SRX380 devices.

Description

Perform checks and show status of all autorecovered items.

Required Privilege Level

view

RELATED DOCUMENTATION

request system autorecovery state | 542

List of Sample Output

show system autorecovery state on page 721

Output Fields

Table 33 on page 720 lists the output fields for the **show system autorecovery state** command. Output fields are listed in the approximate order in which they appear.

Table 33: 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

```
show system boot-messages
```

Syntax (EX Series Switches)

```
show system boot-messages
<all-members>
<local>
<member member-id>
```

Syntax (TX Matrix Router)

```
show system boot-messages
<all-chassis | all-lcc | lcc number | scc>
```

Syntax (TX Matrix Plus Router)

```
show system boot-messages
<all-chassis | all-lcc | lcc number | sfc number>
```

Syntax (MX Series Router)

```
show system boot-messages
<all-members>
<local>
<member member-id>
```

Syntax (QFX Series)

```
show system boot-messages
infrastructure name | interconnect-device name | node-group 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 for the TX Matrix Plus router in Junos OS Release 9.6.

Command introduced in Junos OS Release 11.1 for the QFX Series.

Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display initial messages generated by the system kernel upon startup. These messages are the contents of **/var/run/dmesg.boot**.

Options

none—Display all boot time messages.

all-chassis—(TX Matrix routers andTX 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

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/compil
e/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,<b
16>,<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,VME,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:0xfffffff8
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

Configuring Dual-Root Partitions | 373

List of Sample Output

show system auto-snapshot on page 733

Output Fields

Table 34 on page 733 describes the output fields for the **show system auto-snapshot** command. Output fields are listed in the approximate order in which they appear.

Table 34: show system auto-snapshot status Output Fields

Field Name	Field Description
Auto-snapshot configuration	Status of the configuration: • 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: • **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

```
show system download <download-id>
```

Release Information

Command introduced in Junos OS Release 15.1X49-D35 for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.
Command introduced in Junos OS Release 20.1R1for SRX380 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

- **download-id**—(Optional) The ID number of the download instance.

Required Privilege Level

view

RELATED DOCUMENTATION

request system download start | 552

List of Sample Output

show system download on page 735
show system download 1 on page 735

Output Fields

Table 35 on page 734 lists the output fields for the **show system download** command. Output fields are listed in the approximate order in which they appear.

Table 35: 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.

Table 35: show system download Output Fields *(continued)*

Field Name	Field Description
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 filename | keys | revoked-info | usage>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 13.3 for the MX Series 5G Universal Routing Platform.
Customer ID added to output of data center users in Junos OS Release 15.1.
Corrected output for duration of license added in Junos OS Release 17.4R1.

Description

Display licenses and information about how they are used.

Options

none—Display all license information.

key-content *filename*—(Optional) Display license key contents of the specified filename.

installed—(Optional) Display installed licenses only.

keys—(Optional) Display a list of license keys. Use this information to verify that each expected license key is present.

revoked-info—(Optional) Display information about revoked licenses.

usage—(Optional) Display the state of licensed features.

Required Privilege Level

maintenance

List of Sample Output

show system license (Virtual devices such as vMX and vSRX) on page 738
show system license on page 739
show system license installed on page 739
show system license keys on page 740
show system license usage on page 740
show system license (MX104 Routers) on page 740
show system license installed (MX104 Routers) on page 741
show system license keys (MX104 Routers) on page 741
show system license usage (MX104 Routers) on page 742

show system license (MX104 Routers) on page 742
show system license installed (MX104 Routers) on page 743
show system license keys (MX104 Routers) on page 743
show system license usage (MX104 Routers) on page 743
show system license (MX104 Routers) on page 744
show system license installed (MX104 Routers) on page 744
show system license keys (MX104 Routers) on page 745
show system license usage (MX104 Routers) on page 745
show system license (QFX Series) on page 745
show system license (QFX5110 Switch with Disaggregated Feature License) on page 746
show system license key-content srx_1year_sub.lic on page 747

Output Fields

Table 36 on page 737 lists the output fields for the **show system license** command. Output fields are listed in the approximate order in which they appear.

Table 36: 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 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. 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.
Licenses installed	Information about the installed license key: • **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).

Table 36: show system license Output Fields *(continued)*

Field Name	Field Description
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 (Virtual devices such as vMX and vSRX)

```
user@host> show system license
```

```
License usage:
                                 Licenses     Licenses    Licenses    Expiry
  Feature name                       used    installed      needed
  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:
                                  Licenses     Licenses    Licenses    Expiry
  Feature name                        used    installed      needed
  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 CORPORATIOM
  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:
                                 Licenses     Licenses    Licenses    Expiry
  Feature name                       used    installed      needed
  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:
                                    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

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**

```
          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
```

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:
                                  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

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
           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:
                                  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

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**

```
           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 (QFX Series)

`user@switch>` **show system license**

```
License usage:
                   Licenses     Licenses     Licenses    Expiry
```

```
  Feature name         used    installed      needed
  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:
                                 Licenses     Licenses     Licenses     Expiry
  Feature name                       used    installed       needed
  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

```
show system license
<installed | keys | status | usage>
```

Release Information

Command introduced in Junos OS Release 9.5. Logical system status option added in Junos OS Release 11.2.

Description

Display licenses and information about how licenses are used.

Options

none—Display all license information.

installed—(Optional) Display installed licenses only.

keys—(Optional) Display a list of license keys. Use this information to verify that each expected license key is present.

status—(Optional) Display license status for a specified logical system or for all logical systems.

usage—(Optional) Display the state of licensed features.

Required Privilege Level

view

List of Sample Output

show system license on page 749
show system license installed on page 750
show system license keys on page 750
show system license usage on page 751
show system license status logical-system all on page 751

Output Fields

Table 37 on page 749 lists the output fields for the **show system license** command. Output fields are listed in the approximate order in which they appear.

Table 37: 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.
Licenses installed	Information about the installed license key: • **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:
                                 Licenses     Licenses    Licenses    Expiry
  Feature name                       used    installed      needed
  av_key_kaspersky_engine               1            1           0    2012-03-30
01:00:00 IST
  wf_key_surfcontrol_cpa                0            1           0    2012-03-30
01:00:00 IST
  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**

```
                                   Licenses     Licenses    Licenses  Expiry
Feature name                           used     installed      needed
  av_key_kaspersky_engine             1            1           0    2012-03-30
01:00:00 IST
  wf_key_surfcontrol_cpa              0            1           0    2012-03-30
01:00:00 IST
  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 lockout

Syntax

```
show system login lockout
```

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

retry-options

clear system login lockout | 538

List of Sample Output

show system login lockout on page 753

Output Fields

Table 38 on page 752 lists the output fields for the **show system login lockout** command. Output fields are listed in the approximate order in which they appear.

Table 38: show system login lockout

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 lockout

`user@host>` **show system login lockout**

```
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 754

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 snapshot

List of Syntax

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. You can find which platforms run Junos OS with Upgraded FreeBSD here: "Release Information for Junos OS with Upgraded FreeBSD" on page 35.

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 | 590

List of Sample Output

Output Fields
Table 39 on page 757 lists the output fields for the **show system snapshot** command. Output fields are listed in the approximate order in which they appear.

Table 39: 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 (Junos OS Evolved)

`user@host>` **show system snapshot**

```
                 -------------------------------
                 node: re0
                 -------------------------------
                 Current snapshot device: /dev/sdb
                 Snapshot boot device: sdb
                 List of installed version(s) in Snapshot boot device sdb:
                 [1] ->  junos-linux-install-ptx-fixed-x86-64-18.4-20180819.2 -
[2018-08-21 07:26:13]
                     '-' running version
                     '>' next boot version
                     '<' rollback boot version
```

show system snapshot media external (Switch)

`user@switch>` **show system snapshot media external**

```
Information for snapshot on       external (/dev/da1s1a) (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) | 603
request system reboot (Junos OS with Upgraded FreeBSD) | 580
Release Information for Junos OS with Upgraded FreeBSD | 35

List of Sample Output

show system snapshot on page 761
show system snapshot (updated as of Junos OS Release 17.2) on page 761

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

```
show system snapshot < media (compact-flash | external | harddisk | internal | usb) >
```

Release Information

Command introduced in Junos OS Release 10.2 .

Description

Display information about the partitioning scheme present on the media. Information for only one root is displayed for single-root partitioning, whereas information for both roots is displayed for dual-root partitioning.

Options

- compact-flash— Show snapshot information from the CompactFlash card. (Supported on SRX5400, SRX5600, SRX5800)
- external— Show snapshot information from the external CompactFlash card. (Not supported on SRX5000 Series devices)
- hard-disk— Show snapshot information from the Hard Disk. (Supported on SRX5400, SRX5600, SRX5800)
- internal— Show snapshot information from internal media. (Not supported on SRX5000 Series devices)
- usb— Show snapshot information from device connected to USB port.

Required Privilege Level

View

RELATED DOCUMENTATION

Creating a Snapshot and Using It to Boot an SRX Series device | 56

List of Sample Output

Output Fields

Table 40 on page 764 lists the output fields for the **show system snapshot media** command. Output fields are listed in the approximate order in which they appear.

Table 40: show system snapshot media 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 media compact-flash

show system snapshot media compact-flash

```
Information for snapshot on compact-flash (ad0s1)
Creation date: Aug 21 11:58:14 2017
JUNOS version on snapshot:
  junos  : 12.3X48-D40.5-domestic
```

show system snapshot media external

show system snapshot media external

```
Information for snapshot on      external (/dev/da1s2a) (primary)
Creation date: Apr 9 09:41:16 2018
JUNOS version on snapshot:
  junos  : 12.3X48-D40.5-domestic
Information for snapshot on       external (/dev/da1s1a) (backup)
Creation date: Apr 9 09:41:16 2018
JUNOS version on snapshot:
  junos  : 12.3X48-D40.5-domestic
```

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 (da0s1)
Creation date: Apr 9 08:44:46 2018
JUNOS version on snapshot:
  junos  : 12.3X48-D40.5-domestic
```

show system snapshot media hard-disk

show system snapshot media hard-disk

```
Information for snapshot on hard-disk (ad2s1)
Creation date: Apr 9 16:40:18 2018
JUNOS version on snapshot:
  junos  : 12.3X48-D40.5-domestic
```

show system software list

Syntax

```
show system software list
```

Release Information

Command introduced in Junos OS Evolved Release 18.3R1.

Description

List all software versions installed.

Required Privilege Level

view

List of Sample Output

show system software list on page 766

Output Fields

For a description of the output fields, see Table 41 on page 766. Output fields are listed in the approximate order in which they appear.

Table 41: show system software list Output Fields

Field Name	Description
node	Name of the node.
List of installed version(s)	Ordered list of software that is or has been installed on the node: • - indicates the running software version. • **>** indicates the next boot software version, which occurs only after an upgrade or a downgrade. If no upgrade or downgrade has been performed, this symbol will not be represented in the list of installed versions. • **<** indicates the rollback boot software version if there is one.

Sample Output

show system software list

user@host> **show system software list**

```
-------------------------------
node: RE0
-------------------------------
Active boot device is primary: /dev/sda
List of installed version(s) :

    '-' running version
    '>' next boot version after upgrade/downgrade
    '<' rollback boot version

     - junos-evo-install-ptx-fixed-x86-64-18.4-20181211.1 - [2018-11-23 00:06:02]

     < junos-evo-install-ptx-fixed-x86-64-18.4-20181122.2 - [2018-11-22 19:03:02]

       junos-evo-install-ptx-fixed-x86-64-18.3-20181122.2 - [2018-11-22 16:34:59]
```

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 | 651

List of Sample Output

show system software restore-point status on page 769

Output Fields

Table 42 on page 768 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 42: 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/LogVol03
dg1  Aug 15 07:42:27 2014  success  /dev/VolGroup00/LogVol03
```

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 771

Output Fields

Table 43 on page 771 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 43: 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                            VERSION
                  FILENAME
   ----     ------                            -------
                  --------
    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 storage partitions

List of Syntax

Syntax (EX Series) on page 772
Syntax (SRX Series) on page 772

Syntax (EX Series)

```
show system storage partitions
<all-members>
<local>
<member member-id>
```

Syntax (SRX Series)

```
show system storage partitions
```

Release Information

Command introduced in Junos OS Release 15.1X49-D35 for SRX300, SRX320, SRX340, SRX345, and SRX550HM devices.
Command introduced in Junos OS Release 20.1R1for SRX380 devices.
Command introduced in Junos OS Release 11.1 for EX Series switches.

Description

Display information about the disk partitioning scheme.

Options

none—Display partition information.

all-members—(Virtual Chassis systems only) (Optional) Display partition information for all members of the Virtual Chassis.

local—(Virtual Chassis systems only) (Optional) Display partition information for the local Virtual Chassis member.

member *member-id*—(Virtual Chassis systems only) (Optional) Display partition information for the specified member of the Virtual Chassis configuration.

Required Privilege Level

view

RELATED DOCUMENTATION

List of Sample Output

Output Fields

Table 44 on page 773 describes the output fields for the **show system storage partitions** command. Output fields are listed in the approximate order in which they appear.

Table 44: 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: • 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 system ztp

Syntax

```
show system ztp
```

Release Information

Command introduced in Junos OS Evolved Release 19.1R1.

Description

Show Zero Touch Provisioning (ZTP) state information.

Required Privilege Level

view

RELATED DOCUMENTATION

Zero Touch Provisioning Overview | 410

List of Sample Output

show system ztp on page 778

Output Fields

For a description of the output fields, see Table 45 on page 776. Output fields are listed in the approximate order in which they appear. The state field can have multiple settings. The rest of the fields are self explanatory based on DHCP arguments provided by the server.

Table 45: show system ztp Output Fields

Field Name	Description
ZtpState	ZTP state field values for starting: • **INITIALIZED**—ZTP is initializing. • **STARTED**—ZTP started running.
	ZTP state field values for image download: • **IMAGE_DOWNLOADING**—ZTP is downloading the next software image. • **IMAGE_DOWNLOADED**—ZTP is finished downloading the next software image. • **RETRY_IMAGE_DOWNLOAD**—ZTP is retrying image download. • **IMAGE_NOT_FOUND**—ZTP could not find the image at the specified location on the server.

Table 45: show system ztp Output Fields *(continued)*

Field Name	Description
	ZTP state field values for configuration download: • **CONFIG_DOWNLOADING**—ZTP is downloading the configuration. • **CONFIG_DOWNLOADED**—ZTP is finished downloading the configuration. • **RETRY_CONFIG_DOWNLOAD**—ZTP is retrying configuration download. • **CONFIG_NOT_FOUND**—ZTP could not find the configuration.
	ZTP state field values for upgrading configuration: • **IMAGE_CONFIG_UPGRADING**—ZTP got an image and a configuration from the server. • **CONFIG_UPGRADING**—ZTP is upgrading the configuration.
	ZTP state field values for upgrading image: • **RETRY_IMAGE_UPGRADE**—ZTP is retrying image upgrade. • **IMAGE_CONFIG_UPGRADING**—ZTP got an image and a configuration from the server. • **IMAGE_UPGRADING**—ZTP is downloading the image. • **IMAGE_UPGRADED**—ZTP is finished upgrading the image.
	ZTP state field values for scripts: • **SCRIPT_UPGRADING**—ZTP is running the script provided by server. • **SCRIPT_UPGRADED**—ZTP is finished upgrading the script. • **SCRIPT_UPGRADE_SUCCEEDED**—ZTP script upgrade finished with success. • **SCRIPT_UPGRADE_FAILED**—ZTP script upgrade finished with failure status.
	ZTP state field values for reboot: • **REBOOTING**—ZTP is rebooting the system. • **REBOOTED**—ZTP is finished rebooting the system.
	ZTP state field values for configuration commit: • **CONFIG_COMMIT_SUCCEEDED**—ZTP succeeded in committing user configuration. • **CONFIG_COMMIT_FAILED**—ZTP user configuration commit failed.
	ZTP state field values for finishing: • **FAILED**—ZTP failed. • **SUCCEEDED**—ZTP succeeded.
ZtpInterface	Name of interface.
FtpIpAddr	IP address.

Table 45: show system ztp Output Fields *(continued)*

Field Name	Description
DefaultRouter	When the log server, NTP server, or FTP server are on a remote subnet, the value of **DefaultRouter** is used to configure a route to reach the servers.
LogServers	ZTP allows specification of a remote log server address. ZTP logs are then streamed to the remote log server.
NtpServers	ZTP allows specification of a remote NTP server address.
TransferMode	Options for **TransferMode** are ftp, tftp, http, or https
ImageFileType	It can be a symbolic link.
ConfigFileName	Configuration filename.
ConfigUrl	Configuration URL.
ConfigStatus	This field specifies whether the config file is downloading, is downloaded, or the download is being retried.
ZtpRetryCount	If the ZTP state machine, which applies the image and configuration, fails, the number of retries attempted.
DhcpRetryCount	If the DHCP state machine, which fetches parameters for ZTP from the DHCP server, fails, the number of times it retries.
ZTP State History	Lists the last 10 state transitions by **Time** (date and time) and **Description** or which state it was in then.

Sample Output

show system ztp

`user@host>` **show system ztp**

```
Attribute          Value
 --------------------------
 ZtpState           SUCCEEDED
 ZtpInterface       vmb
 FtpIpAddr          10.92.14.35
```

```
TransferMode       ftp
ImageFileType      None
ImageStatus        None
ConfigStatus       CONFIG DOWNLOADED
ZtpRetryCount      0
DhcpRetryCount     0
ZTP State History(last 10 transitions)
  Time                       Description
  Mon Apr 29 18:38:58 2019   Succeeded
```

show version

Syntax

```
show version
<brief | detail>
<node node-id | local | primary>
```

Release Information

Command introduced in Junos OS Release 10.4.

Description

Display the hostname and version information about the software running on the device.

Options

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 *node-id*—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

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]

...
```

14

CHAPTER

VM Host Administration Commands

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 devices 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**).

NOTE: The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

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 | 679

Output Fields

When you enter this command, you are provided feedback on the status of your request.

request vmhost copy jnode-to-vjunos

Syntax

```
request vmhost copy jnode-to-vjunos from-jnode host-filename to-vjunos junos-filename
```

Release Information

Command introduced in Junos OS Release 18.4R1 on Enhanced Automation variants of Junos OS. For more information, see *Overview of Junos Automation Enhancements on Devices Running Junos OS with Enhanced Automation*.

Description

Copy files from Linux host to Junos OS.

Options

from-jnode *host-filename*—Name of the host file to be copied.

to-vjunos *junos-filename*—Name of the Junos OS file to which the host file is copied.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request vmhost copy vjunos-to-jnode | 788

request vmhost copy vjunos-to-jnode

Syntax

```
request vmhost copy vjunos-to-jnode from-vjunos junos-filename to-jnode host-filename
```

Release Information

Command introduced in Junos OS Release 18.4R1 on Enhanced Automation variants of Junos OS. For more information, see *Overview of Junos Automation Enhancements on Devices Running Junos OS with Enhanced Automation*.

Description

Copy files from Junos OS to Linux host.

Options

from-vjunos *junos-filename*—Name of the Junos OS file to be copied.

to-jnode *host-filename*—Name of the host file to which to copy the file.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request vmhost copy jnode-to-vjunos | 787

request vmhost file-copy

Syntax

```
request vmhost file-copy (crash|log) j-node host file-name v-junos host file-name
```

Release Information

Command introduced in Junos OS Release 15.1F3 for the PTX5000, MX240, MX480, and MX960 routers.
Command introduced in Junos OS Release 15.1F5 for the MX2010 and MX2020 routers.
Command introduced in Junos OS Release 16.1R4 for the PTX3000 routers.

> **NOTE:** PTX3000 router supports the Routing and Control Board, RCBPTX.

Command introduced in Junos OS Release 17.1R1 for EX9200 switches.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

Description

Copy crash files or log files from the host OS to Junos OS. You can use these files for analysis and debugging purposes.

Options

crash—Files in **/var/crash** on the host.

from j-node *filename*—Name of the host file to be copied.

log—Files in **/var/log** on the host.

to v-junos *filename*—Name of the Junos OS file to which the host file is copied.

Additional Information

You can use the **show vmhost crash** and **show vmhost logs** commands to list or identify the files in the host OS to be copied to Junos OS.

Required Privilege Level

maintenance

List of Sample Output

request vmhost file-copy on page 790

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

Description

Stop the host OS and Junos OS running on the device.

Options

none—Stop the host OS and Junos OS on the device.

re0 | re1—(Optional) On devices 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:

> **NOTE:** The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 | 523

List of Sample Output
request vmhost halt on page 792

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.

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 | 523

List of Sample Output

request vmhost hard-disk-test on page 794

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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.

> **NOTE:** The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

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 | 523

List of Sample Output

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.

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.

> **NOTE:** The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request vmhost power-off | 795

List of Sample Output

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 router.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

Description

Reboot both the Junos OS and the host OS running on the device.

Options

none—Reboot the device 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:

> **NOTE:** The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 | 570
request vmhost power-on | 797

List of Sample Output
request vmhost reboot on page 800

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 router.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M 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 device, 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.

> **NOTE:** The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

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 | 854

List of Sample Output
request vmhost snapshot on page 804
request vmhost snapshot recovery on page 804

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 in-service-upgrade

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.2R1 for MX10008 router.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 | 810

request vmhost software add

List of Syntax

MX Series on page 806
PTX Series on page 806

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 router.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

Description

Install Junos OS and host software packages on the device.

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, PTX10016, PTX10002-60C, and QFX10002-60C devices 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, PTX10016, PTX10002-60C, and QFX10002-60C devices do not support Junos only upgrade. If you try to load Junos only image, then these devices go down or vmhost commands do not work or the device state is unpredictable.

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 device.
- ***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 Protcol (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 device, 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 device 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.

Before installing software on a device that has one or more custom YANG data models added to it, back up and remove the configuration data corresponding to the custom YANG data models from the active configuration. For more information see “Managing YANG Packages and Configurations During a Software Upgrade or Downgrade” on page 150.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system software add | 613
request vmhost software rollback | 815

List of Sample Output

request vmhost software add (Multiple Packages) on page 809

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>
< verbose>
```

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.
Option introduced in Junos OS Release 18.2R1 for the MX Series 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 18.2R1 for MX10008 router.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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.

Starting inJunos OS Release 18.2R1, MX10003 routers support unified in-service software upgrade (ISSU) using **request vmhost software in-service-upgrade** command. MX10003 does not support upgrading Junos OS only image using *request system software in-service-upgrade* command.

NOTE: On MX10003 routers:

- ISSU is not supported on MACsec MIC (JNP-MIC1-MACSEC).
- ISSU is not supported for the interfaces that are configured with 1-Gigabit Ethernet mode. If ISSU upgrade is carried out in 1-Gigabit Ethernet mode, then the behavior is unexpected and traffic loss can be expected.
- ISSU is not supported on timing protocols (like, Precision Time Protocol and Synchronous Ethernet), MACsec protocols, and BBE protocols. If these protocols are already enabled, then it will not work after ISSU is enabled.
- The MAC statistics (retrieved using *show interfaces extensive* command) are reset during ISSU which means that the MAC statistics does not provide the correct statistics after ISSU.

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 don 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.

verbose—(Optional) (MX Series) Use this option to display the daemon related information during the upgrade.

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.
- During a unified ISSU, any unsupported protocols running on the device causes packet loss.
- During a unified ISSU, you cannot bring any PICs online or take them offline on certain routing devices.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system software in-service-upgrade

List of Sample Output

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.
  ...
  ...
```

Sample Output

request vmhost software in-service-upgrade verbose

`user@host>` **request vmhost software in-service-upgrade verbose**

```
  ...
  ...
Oct 26 15:20:02 [INFO ] Verified py-extensions signed by PackageDevelopmentEc_2017
 method ECDSA256+SHA256
Oct 26 15:20:02 [INFO ] Adding
py-extensions-x86-32-20171024.002108_builder_release_174_throttle ...
Oct 26 15:20:02 [INFO ] Verified vrr-mx signed by PackageDevelopmentEc_2017 method
 ECDSA256+SHA256
Oct 26 15:20:02 [INFO ] NOTICE: 'pending' set will be activated at next reboot...
Oct 26 15:20:02 [INFO ] [Oct 26 02:36:10]:ISSU: Installing package
/var/tmp/junos-install-mx-x86-64-17.4-20171024.0.tgz on re1 done
Oct 26 15:20:02 [INFO ] [Oct 26 02:36:10]:ISSU: Rebooting Backup RE
Oct 26 15:20:02 [INFO ]
Oct 26 15:20:02 [INFO ] Rebooting re1
Oct 26 15:20:02 [INFO ] [Oct 26 02:36:11]:ISSU: Backup RE Prepare Done
Oct 26 15:20:02 [INFO ] [Oct 26 02:36:11]:ISSU: Waiting for Backup RE reboot
Oct 26 15:20:02 [INFO ] [Oct 26 02:39:26]:ISSU: Backup RE reboot done. Backup RE
is up
Oct 26 15:20:02 [INFO ] [Oct 26 02:39:26]:ISSU: Waiting for Backup RE state
synchronization
Oct 26 15:20:02 [INFO ] [Oct 26 02:39:51]:ISSU: Backup RE state synchronization
done
Oct 26 15:20:02 [INFO ] [Oct 26 02:39:51]:ISSU: GRES operational
Oct 26 15:20:02 [INFO ] [Oct 26 02:40:52]: "Initiating Chassis In-Service-Upgrade"
Oct 26 15:20:02 [INFO ] Chassis ISSU Started
Oct 26 15:20:02 [INFO ] [Oct 26 02:40:57]:ISSU: Preparing Daemons
Oct 26 15:20:02 [INFO ] [Oct 26 02:40:57]: Daemon [rpd] transitioned to READY state
Oct 26 15:20:02 [INFO ] [Oct 26 02:40:57]: Daemon [lfmd] transitioned to READY
state
Oct 26 15:20:02 [INFO ] [Oct 26 02:40:57]: Daemon [l2cpd] transitioned to READY
state
Oct 26 15:20:02 [INFO ] [Oct 26 02:40:57]: Daemon [smid] transitioned to READY
state
Oct 26 15:20:02 [INFO ] [Oct 26 02:40:57]: Daemon [bfdd] transitioned to READY
state
Oct 26 15:20:02 [INFO ] [Oct 26 02:41:17]:ISSU: Daemons Ready for ISSU
Oct 26 15:20:02 [INFO ] [Oct 26 02:41:17]: Daemon [apsd] transitioned to READY
state
```

```
Oct 26 15:20:02 [INFO ] [Oct 26 02:41:22]:ISSU: Offline Incompatible FRUs
Oct 26 15:20:02 [INFO ] [Oct 26 02:41:27]:ISSU: Starting Upgrade for FRUs
Oct 26 15:20:02 [INFO ] [Oct 26 02:41:27]: [FPC 1] None -> Prepare
Oct 26 15:20:02 [INFO ] [Oct 26 02:41:48]: [FPC 1] Prepare -> Ready for Reboot
Oct 26 15:20:02 [INFO ] [Oct 26 02:41:52]: [FPC 1] Ready for Reboot -> Reboot
Oct 26 15:20:02 [INFO ] [Oct 26 02:42:01]: [FPC 1] Reboot -> Blob Resync
Oct 26 15:20:02 [INFO ] [Oct 26 02:42:28]: [FPC 1] Blob Resync -> Ready Software
State Sync
Oct 26 15:20:02 [INFO ] [Oct 26 02:42:32]: [FPC 1] Ready Software State Sync ->
Software State Sync
Oct 26 15:20:02 [INFO ] [Oct 26 02:44:02]: [FPC 1] Software State Sync -> Ready
Hardware State Sync
Oct 26 15:20:02 [INFO ] [Oct 26 02:44:02]: [FPC 1] Ready Hardware State Sync ->
Hardware State Sync
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:07]: [FPC 1] Hardware State Sync ->
Reconnected
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:07]:ISSU: FRU Upgrade Done
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:07]:ISSU: Preparing for Switchover
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:07]: Daemon [lfmd] transitioned to
SWITCHOVER_READY state
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:07]: Daemon [rpd] transitioned to
SWITCHOVER_READY state
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:07]: Daemon [l2cpd] transitioned to
SWITCHOVER_READY state
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:07]: Daemon [smid] transitioned to
SWITCHOVER_READY state
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:07]: Daemon [bfdd] transitioned to
SWITCHOVER_READY state
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:27]: Daemon [apsd] transitioned to
SWITCHOVER_READY state
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:29]: Checking In-Service-Upgrade status
Oct 26 15:20:02 [INFO ]   Item           Status                  Reason
Oct 26 15:20:02 [INFO ]   FPC 1          Online (ISSU)
Oct 26 15:20:02 [INFO ] Resolving mastership...
Oct 26 15:20:02 [INFO ] Complete. The other routing engine becomes the master.
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:30]:ISSU: RE switchover Done
Oct 26 15:20:02 [INFO ] [Oct 26 02:47:30]:ISSU: Upgrading Old Master RE
Oct 26 15:20:02 [INFO ] Verified junos-install-mx-x86-64-17.4-20171024.0 signed
by PackageDevelopmentEc_2017 method ECDSA256+SHA256
Oct 26 15:20:02 [INFO ] Verified manifest signed by PackageDevelopmentEc_2017
method ECDSA256+SHA256
  ...
    ...
```

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 router.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 devices 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**).

> **NOTE:** The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

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 | 806

request vmhost software abort in-service-upgrade | 805

List of Sample Output
request vmhost software rollback on page 816

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 software validate

Syntax

```
request vmhost software validate package-name
```

Release Information

Statement introduced in Junos OS Release 18.4R1

> **NOTE:** The command is supported on the routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines with VM host support only.

Description

Verify and validate the software package compatibility with the current configuration of the router.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request system software abort

request vmhost software in-service-upgrade | 810

Sample Output

request vmhost software validate

user@host> **request vmhost software validate junos-vmhost-install-mx-x86-64-18.3R1-S1.1.tgz**

```
Junos Validation begin. Procedure will take few minutes.
Initializing...
Verified os-libs-11-x86-64-20180816 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting os-libs-11-x86-64-20180816.8630ec5_builder_stable_11
Verified os-runtime-x86-64-20180816 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting os-runtime-x86-64-20180816.8630ec5_builder_stable_11
Verified os-zoneinfo-x86-64-20180816 signed by PackageProductionEc_2018 method
```

```
ECDSA256+SHA256
Mounting os-zoneinfo-x86-64-20180816.8630ec5_builder_stable_11
Verified junos-net-prd-x86-64-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting junos-net-prd-x86-64-20181022.105815_builder_junos_183_r1_s1
Verified junos-libs-x86-64-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting junos-libs-x86-64-20181022.105815_builder_junos_183_r1_s1
Verified os-libs-compat32-11-x86-64-20180816 signed by PackageProductionEc_2018
method ECDSA256+SHA256
Mounting os-libs-compat32-11-x86-64-20180816.8630ec5_builder_stable_11
Verified os-compat32-x86-64-20180816 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting os-compat32-x86-64-20180816.8630ec5_builder_stable_11
Verified junos-libs-compat32-x86-64-20181022 signed by PackageProductionEc_2018
method ECDSA256+SHA256
Mounting junos-libs-compat32-x86-64-20181022.105815_builder_junos_183_r1_s1
Verified junos-runtime-x86-32-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting junos-runtime-x86-32-20181022.105815_builder_junos_183_r1_s1
Verified junos-vmguest-mtx-x86-64-20181022 signed by PackageProductionEc_2018
method ECDSA256+SHA256
Mounting junos-vmguest-mtx-x86-64-20181022.105815_builder_junos_183_r1_s1
Verified sflow-mx-x86-32-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting sflow-mx-x86-32-20181022.105815_builder_junos_183_r1_s1
Verified py-extensions-x86-32-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting py-extensions-x86-32-20181022.105815_builder_junos_183_r1_s1
Verified py-base-x86-32-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting py-base-x86-32-20181022.105815_builder_junos_183_r1_s1
Verified os-vmguest-x86-64-20180816 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting os-vmguest-x86-64-20180816.8630ec5_builder_stable_11
Verified os-crypto-x86-64-20180816 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting os-crypto-x86-64-20180816.8630ec5_builder_stable_11
Verified na-telemetry-x86-32-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting na-telemetry-x86-32-20181022.105815_builder_junos_183_r1_s1
Verified junos-libs-compat32-mx-x86-64-20181022 signed by PackageProductionEc_2018
 method ECDSA256+SHA256
Mounting junos-libs-compat32-mx-x86-64-20181022.105815_builder_junos_183_r1_s1
```

```
Verified junos-runtime-mx-x86-32-20181022 signed by PackageProductionEc_2018 method
 ECDSA256+SHA256
Mounting junos-runtime-mx-x86-32-20181022.105815_builder_junos_183_r1_s1
Verified junos-platform-x86-32-20181022 signed by PackageProductionEc_2018 method
 ECDSA256+SHA256
Mounting junos-platform-x86-32-20181022.105815_builder_junos_183_r1_s1
Verified junos-openconfig-x86-32-20181022 signed by PackageProductionEc_2018 method
 ECDSA256+SHA256
Mounting junos-openconfig-x86-32-20181022.105815_builder_junos_183_r1_s1
Verified junos-modules-x86-64-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting junos-modules-x86-64-20181022.105815_builder_junos_183_r1_s1
Verified junos-modules-mx-x86-64-20181022 signed by PackageProductionEc_2018 method
 ECDSA256+SHA256
Mounting junos-modules-mx-x86-64-20181022.105815_builder_junos_183_r1_s1
Verified junos-libs-mx-x86-64-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting junos-libs-mx-x86-64-20181022.105815_builder_junos_183_r1_s1
Verified junos-dp-crypto-support-mtx-x86-32-20181022 signed by
PackageProductionEc_2018 method ECDSA256+SHA256
Mounting junos-dp-crypto-support-mtx-x86-32-20181022.105815_builder_junos_183_r1_s1
Verified junos-daemons-x86-64-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting junos-daemons-x86-64-20181022.105815_builder_junos_183_r1_s1
Verified junos-daemons-mx-x86-64-20181022 signed by PackageProductionEc_2018 method
 ECDSA256+SHA256
Mounting junos-daemons-mx-x86-64-20181022.105815_builder_junos_183_r1_s1
Verified jsdn-x86-32-18.3R1-S1 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting jsdn-x86-32-18.3R1-S1.1
Verified jsd-x86-32-18.3R1-S1 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting jsd-x86-32-18.3R1-S1.1-jet-1
Verified jpfe-common-x86-32-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting jpfe-common-x86-32-20181022.105815_builder_junos_183_r1_s1
Verified jinsight-x86-32-18.3R1-S1 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting jinsight-x86-32-18.3R1-S1.1
Verified jfirmware-x86-32-18.3R1-S1 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
Mounting jfirmware-x86-32-18.3R1-S1.1
Verified jdocs-x86-32-20181022 signed by PackageProductionEc_2018 method
ECDSA256+SHA256
```

```
Mounting jdocs-x86-32-20181022.105815_builder_junos_183_r1_s1
Hardware Database regeneration succeeded
Validating against /config/juniper.conf.gz
mgd: commit complete
Validation succeeded
```

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 router.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M 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**).

NOTE: The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

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 | 802

List of Sample Output

request vmhost zeroize on page 824

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
Zeroise 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
Zeroise secondary internal disk ...
Proceeding with zeroize on secondary disk
```

15

CHAPTER

VM Host Monitoring Commands

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 device 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 devices 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**).

NOTE: The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

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 829

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 830

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

```
show vmhost hard-disk -test { disk disk-name | status}
```

Release Information

Command introduced in Junos OS Release 17.2R1 for the PTX5000, MX240, MX480, MX960 , MX2010, and MX2020 routers.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

Description

Display memory and diagnostics monitoring test status on the solid-state drive (SSD). The test output provides information about the various attributes of the SSD that help to monitor the status of the hard disk memory. This command should be used only after initiating the disk test with the **request vmhost hard-disk-test** command.

Options

disk *disk-name*— Display the name of the SSD.

status—Display the status of the test.

Required Privilege Level

maintenance

RELATED DOCUMENTATION

request vmhost hard-disk-test | 793

List of Sample Output

show vmhost hard-disk-test on page 833

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.
                                        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.
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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 device 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 devices 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:

NOTE: The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 835

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 device 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 devices 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:

NOTE: The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 838

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

Description

Display trace logs information of the host OS.

Options

re0 | re1—(Optional) On devices 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:

> **NOTE:** The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 840

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 for the PTX5000, MX240, MX480, MX960, MX2010 and MX2020 routers

> **NOTE:** The command is supported on the routers with RE-MX-X6, RE-MX-X8, and RE-PTX-X8 Routing Engines only.

Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 842

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 device 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 devices 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:

NOTE: The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 845

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.

Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

Description

Display information about the host processes that are running on the device.

Options

invoke-on—(Optional) Display information about the host processes that are running on a device 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 devices 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:

NOTE: The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 848

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 [posixcputmr/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 [posixcputmr/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 [posixcputmr/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 [posixcputmr/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 [posixcputmr/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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 device 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 devices 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:

> **NOTE:** The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 851

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-jlvmspare   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
```

It is normal to see 100 percentage or close to 100 percentage of CPU usage in the **%guest** column in the **CPU Usage** section.

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 device 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 devices 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:

NOTE: The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 | 815
request vmhost snapshot | 802

List of Sample Output
show vmhost snapshot on page 855

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 devices 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:

NOTE: The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 858

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers

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 device 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 devices that support dual or redundant Routing Engines, display the uptime information about the hoston 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:

> **NOTE:** The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 860

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.
Command introduced in Junos OS Release 17.3R1 for the MX10003 routers.
Command introduced in Junos OS Release 17.4R1 for the MX204 routers.
Command introduced in Junos OS Release 18.1R1 for the QFX10002-60C switch.
Command introduced in Junos OS Release 18.2R1 for the PTX10002-60C router.
Command introduced in Junos OS Release 18.2R1 for MX10008 Routers
Statement introduced in Junos OS Release 19.2R1-S1 for the ACX5448-D routers and the ACX5448-M routers.

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 device 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 devices 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:

NOTE: The QFX10002-60C and PTX10002-60C devices do not have master and backup routing engines.

- **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 862

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
```

16

CHAPTER

Configuration Statements from Junos SDK Guide

control-cores

Syntax

```
control-cores control-number;
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]
```

Release Information

Statement introduced in Junos OS Release 9.0.

Description

Configure control cores. Any cores not configured as either control or data cores are treated as user cores. When the number of control cores is changed, the PIC reboots.

Options

control-number—Number of control cores. At least one core must be a control core.

Range: 1 through 8

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

data-cores | 867

data-cores

Syntax

```
data-cores data-number;
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]
```

Release Information

Statement introduced in Junos OS Release 9.0.

Description

Configure data cores. Any cores not configured as either data or control cores are treated as user cores. When the number of data cores is changed, the PIC reboots.

Options

data-number—Number of data cores. Although it is not mandatory to dedicate any cores as data cores, it is advisable, depending on the nature of the application, to dedicate a minimum of five as data cores to achieve good performance.

Range: 0 through 7

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

control-cores | 866

data-flow-affinity

Syntax

```
data-flow-affinity {
   hash-key (layer-3 | layer-4);
}
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]
```

Release Information

Statement introduced in Junos OS Release 9.5.

Description

Enable flow affinity distribution for packets over data CPUs on the PIC. Once enabled, the default behavior distributing data packets changes from a round-robin distribution to a flow affinity distribution based on a hash distribution. Adding or deleting this statement causes the PIC to reboot.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

destination (Chassis)

Syntax

```
destination destination;
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider syslog facility]
```

Release Information

Statement introduced in Junos OS Release 10.1.

Description

Configure where log messages go. By default, all messages go to the **/var/log** directory on the Routing Engine. Enhancements to the existing infrastructure make debugging on the Multiservices PIC easier by giving the user the option of redirecting log messages. When the **syslog destination** statement is configured to redirect the log messages, you can use the **set system syslog** command, a command available in the native Junos OS CLI, to override the syslog settings made on the Multiservices PIC.

Options

destination—Choose one of the following options:

- **routing-engine**—Forward log messages to the Routing Engine.
- **pic-console**—Forward log messages to the console of the PIC.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

extension-provider | 870

extension-provider

Syntax

```
extension-provider {
  control-cores control-number;
  data-cores data-number;
  data-flow-affinity {
    hash-key (layer-3 | layer-4);
  }
  forwarding-db-size size;
  object-cache-size size;
  package package-name;
  policy-db-size size;
  syslog {
    facility {
      severity;
      destination destination;
    }
  }
  wired-max-processes num-procs;
  wired-process-mem-size mem-size;
}
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package]
```

Release Information

Statement introduced in Junos OS Release 9.0.

Description

Configure an application on a PIC.

> **CAUTION:** Do not configure these settings unless it is specified you should do so. The default settings work under most normal scenarios. Unneeded settings can cause negative effects.

When the **extension-provider** statement is first configured, the PIC reboots.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

forwarding-db-size

Syntax

```
forwarding-db-size size;
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]
```

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Configure the size of the forwarding database (FDB). When this setting is changed, the PIC reboots.

> **NOTE:** You need to enable the **forwarding-options sampling** statement for the FDB to be created.

Options

size—Size of the FDB, in megabytes (MB). The size of the FDB and the size of the policy database together must be smaller than the size of the object cache.

Range: 0 through 12879 MB

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

policy-db-size | 880

wired-process-mem-size | 898

object-cache-size | 876

hash-key (Chassis)

Syntax

```
hash-key (layer-3 | layer-4);
```

Hierarchy Level

[edit chassis fpc *slot-number* pic *pic-number* adaptive-services service-package extension-provider **data-flow-affinity**]

Release Information

Statement introduced in Junos OS Release 10.2.

Description

Set the hashing distribution of flow affinity. This is an optional setting. Once the **data-flow-affinity** statement is enabled, you may need to choose the hashing distribution. Modifying this statement causes the PIC to reboot.

Default

If you do not configure the **hash-key** statement, the hashing distribution is 5-tuple hashing, or **layer-4**.

Options

layer-3—3-tuple hashing (source IP address, destination IP address, and IP protocol).

layer-4—5-tuple hashing (3-tuple plus source and destination TCP or UDP ports).

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

extension-provider | 870

ip-address-owner

Syntax

```
ip-address-owner owner;
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number]
```

Release Information

Statement introduced in Junos OS Release 13.2R1.

Description

Define the owner for IP addresses hosted on an ms- interface. This statement is used to specify that the steering of control plane packets to the Multiservices PIC be preserved. The provider of your application or its user documentation will tell you when you need to set this configuration.

Options

owner—Owner of the IP address for the interface. There are two options:

Values:

routing-engine—IP address defined on the interface is hosted by the Routing Engine. This option is not used.

service-plane—IP address defined on the interface is hosted by the service plane. This option is used to preserve the packet steering behavior built into your application. The provider of your application or its user documentation will tell you when you need to set this configuration.

Required Privilege Level

view

jdaf

Syntax

```
jdaf {
    routing-instances [ routing-instance-names ];
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 14.1.

Description

Configure Juniper distributed application framework (JDAF).

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.

object-cache-size

Syntax

```
object-cache-size value;
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]
```

Release Information

Statement introduced in Junos OS Release 9.0.

Description

Configure the size of the object cache. When this setting is changed, the PIC reboots.

Options

value—Amount of object cache, in MB. Only values in increments of 128 MB are allowed.

Range: For Multiservices 100 PIC, range is 128 MB through 512 MB. If the **wired-process-mem-size** statement at the same hierarchy level has a value of 512 MB, the maximum value for this statement is 128 MB.

Range: For Multiservices 400 PIC, range is 128 MB through 1280 MB. If the **wired-process-mem-size** statement at the same hierarchy level has a value of 512 MB, the maximum value for this statement is 512 MB.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

forwarding-db-size | 872
policy-db-size | 880
wired-process-mem-size | 898

package (Loading on PIC)

Syntax

```
package package-name;
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]
```

Release Information

Statement introduced in Junos OS Release 9.1.

Description

Identify a package to be loaded on the PIC. When a package is added or removed, the PIC reboots.

Options

package-name—Name of the package to be loaded on the PIC. There can be up to eight packages loaded on a PIC; however, only one data package is allowed per PIC. An error message is displayed if more than eight packages are specified.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

package (Resource Limits)

Syntax

```
package package-name {
    resources {
        cpu {
            priority number;
            time seconds;
        }
        file {
            core-size bytes;
            open number;
            size bytes;
        }
        memory {
            data-size bytes;
            locked-in bytes;
            resident-set-size bytes;
            socket-buffers bytes;
            stack-size bytes;
        }
    }
}
```

Hierarchy Level

```
[edit system extensions resource-limits]
```

Release Information

Statement introduced in Junos OS Release 9.6.

Description

Set resource limits for an entire package of an application.

Options

package-name—Name of the application package.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

admin—To view this statement in the configuration.
admin-control—To add this statement to the configuration.

RELATED DOCUMENTATION

process | 881

resources | 887

extensions

policy-db-size

Syntax

```
policy-db-size size;
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Configure the size of the policy database. When this setting is changed, the PIC reboots.

> **NOTE:** At least one data core must be configured to configure the size of the policy database.

Options

size—Size of the policy database, in megabytes (MB). The size of the forwarding database and the size of the policy database together must be smaller than the size of the object cache.

Range: 0 through 1279 MB

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

forwarding-db-size | 872
object-cache-size | 876
wired-process-mem-size | 898

process

Syntax

```
process process-ui-name {
    resources {
        cpu {
            priority number;
            time seconds;
        }
        file {
            core-size bytes;
            open number;
            size bytes;
        }
        memory {
            data-size bytes;
            locked-in bytes;
            resident-set-size bytes;
            socket-buffers bytes;
            stack-size bytes;
        }
    }
}
```

Hierarchy Level

```
[edit system extensions resource-limits]
```

Release Information

Statement introduced in Junos OS Release 9.6.

Description

Set resource limits for a process in an application package. Limits defined for individual processes override the limits defined for an entire package.

Options

process-ui-name—Name of the process.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

admin—To view this statement in the configuration.
admin-control—To add this statement to the configuration.

RELATED DOCUMENTATION

package (Resource Limits) | 878

resources | 887

extensions

process-monitor

Syntax

```
process-monitor {
    disable;
    traceoptions {
        file filename files number match regex size size (world-readable | no-world-readable);
        flag flag;
        level level;
        no-remote-trace;
    }
}
```

Hierarchy Level

```
[edit system processes]
```

Release Information

Statement introduced in Junos OS Release 9.0.

Description

Configure tracing options for the process health monitor process (pmond).

> **NOTE:** Starting with Junos OS Release 15.1R2, the pmond process is enabled by default on the Routing Engines of MX Series routers, even when no service interfaces are configured.

Options

disable—Disable the process health monitor process.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

admin—To view this statement in the configuration.
admin-control—To add this statement to the configuration.

RELATED DOCUMENTATION

traceoptions (Process Monitor) | 893

resource-cleanup

Syntax

```
resource-cleanup {
    disable;
    traceoptions {
        file filename files number match regex size size (world-readable | no-world-readable);
        flag flag;
        level level;
        no-remote-trace;
    }
}
```

Hierarchy Level

```
[edit system processes]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Selectively turn on or off the debugging of trace messages for the resource cleanup process.

Options

disable—Disable the resource cleanup process.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

trace—To view this statement in the configuration.
trace-control—To add this statement to the configuration.

RELATED DOCUMENTATION

traceoptions (Resource Cleanup) | 895

resource-limits

Syntax

```
resource-limits {
    package package-name {
        resources {
            cpu {
                priority number;
                time seconds;
            }
            file {
                core-size bytes;
                open number;
                size bytes;
            }
            memory {
                data-size bytes;
                locked-in bytes;
                resident-set-size bytes;
                socket-buffers bytes;
                stack-size bytes;
            }
        }
    }
    process process-ui-name {
        resources {
            cpu {
                priority number;
                time seconds;
            }
            file {
                core-size bytes;
                open number;
                size bytes;
            }
            memory {
                data-size bytes;
                locked-in bytes;
                resident-set-size bytes;
                socket-buffers bytes;
                stack-size bytes;
            }
        }
```

```
    }
}
```

Hierarchy Level

```
[edit system extensions]
```

Release Information

Statement introduced in Junos OS Release 9.6.

Description

Set resource limits for applications using the command-line interface (CLI). You can set limits for all applications listed in the application package’s manifest file or define limits for individual processes in the package. Limits defined for individual processes override the limits defined for an entire package.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

admin—To view this statement in the configuration.
admin-control—To add this statement to the configuration.

resources

Syntax

```
resources {
    cpu {
        priority number;
        time seconds;
    }
    file {
        core-size bytes;
        open number;
        size bytes;
    }
    memory {
        data-size bytes;
        locked-in bytes;
        resident-set-size bytes;
        socket-buffers bytes;
        stack-size bytes;
    }
}
```

Hierarchy Level

```
[edit system extensions resource-limits package package-name],
[edit system extensions resource-limits process process-ui-name]
```

Release Information

Statement introduced in Junos OS Release 9.6.

Description

Set resource limits for applications.

Options

bytes—Maximum size of each file, in kilobytes (KB) or megabytes (MB).

Syntax: Where *x* is some number, use *xk* to specify KB or *xm* to specify MB.

cpu—CPU resources.

- **priority *number***—Highest priority number (nice level) at which the process can run.
- **time *seconds***—Maximum amount of CPU time that can be accumulated.

file—File system resources.

- **core-size *bytes***—Maximum size of a core file that can be created.
- **open *number***—Maximum number of simultaneous open files.
- **size *bytes***—Maximum size of a file that can be created.

memory—Memory resources.

- **data-size *bytes***—Maximum size of the data segment.
- **locked-in *bytes***—Maximum number of bytes that can be locked into memory.
- **resident-set-size *bytes***—Maximum amount of private or shared memory at any given moment.
- **socket-buffers *bytes***—Maximum amount of physical memory that may be dedicated to the socket buffers.
- **stack-size *bytes***—Maximum size of the stack segment.

Required Privilege Level
admin—To view this statement in the configuration.
admin-control—To add this statement to the configuration.

RELATED DOCUMENTATION

resource-limits | 885

extensions

routing-instances

Syntax

```
routing-instances [ routing-instance-names ];
```

Hierarchy Level

```
[edit services jdaf]
```

Release Information

Statement introduced in Junos OS Release 14.1.

Description

Configure the routing instances on which Juniper Distributed Application Framework (JDAF) is enabled. If the **jdaf** statement is not configured, then JDAF is disabled.

Options

routing-instances [*routing-instance-names*]—Name or names of routing instances for JDAF clients. If multiple names are being configured, these can be set as an open set of values.

Required Privilege Level

system—To view this statement in the configuration.
system-control—To add this statement to the configuration.

RELATED DOCUMENTATION

jdaf | 875

service-order

Syntax

```
service-order {
    forward-flow [ service-name1 service-name2 ];
    reverse-flow [ service-name1 service-name2 ];
}
```

Hierarchy Level

```
[edit services service-set service-set-name]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Define the order of services in service set to be applied to traffic coming to the PIC.

> **NOTE:** If the **extension-service** statement is specified, the **service-order** statement is mandatory.

Options

forward-flow—Order of services in service set to be applied in forward flow.

reverse-flow—Order of services in service set to be applied in reverse flow. If you want the order to be the reverse of that specified for forward flow, this is optional. However, if, for example, you want the order to be the same regardless of direction of flow, you must include this statement. (The exception to this is for the sampling service set type. If a service set is a sampling service set and the reverse-flow service order is not configured, all sampled traffic is considered to be forward traffic.)

Required Privilege Level

system—To view this statement in the configuration.
system-control—To add this statement to the configuration.

RELATED DOCUMENTATION

extension-service

syslog (Chassis)

Syntax

```
syslog {
  facility {
    severity;
    destination destination;
  }
}
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]
```

Release Information

Statement introduced in Junos OS Release 9.2.
Options **daemon** and **kernel** (for ***facility***) introduced in Junos OS Release 9.5.

Description

Enable PIC system logging to record or view system log messages on a specific PIC. The system log information is passed to the kernel for logging in the **/var/log** directory.

Options

facility—Group of messages that are either generated by the same software process or concern a similar condition or activity. Possible values include the following: **daemon**, **external**, **kernel**, and **pfe**.

severity—Classification of effect on functioning. Possible values are the following options:

- **any**—Include all severity levels.
- **none**—Disable logging of the associated facility to a destination.
- **emergency**—System panic or other condition that causes the routing platform to stop functioning.
- **alert**—Conditions that require immediate correction, such as a corrupted system database.
- **critical**—Critical conditions, such as hard errors.
- **error**—Error conditions that generally have less serious consequences than errors in the emergency, alert, and critical levels.
- **warning**—Conditions that warrant monitoring.
- **notice**—Conditions that are not errors but might warrant special handling.
- **info**—Events or nonerror conditions of interest.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

traceoptions (Process Monitor)

Syntax

```
traceoptions {
    file filename files number match regex size size (world-readable | no-world-readable);
    flag flag;
    level level;
    no-remote-trace;
}
```

Hierarchy Level

```
[edit system processes process-monitor]
```

Release Information

Statement introduced in Junos OS Release 9.0.

Description

Enable tracing options for the process health monitor process (pmond).

> **NOTE:** Starting with Junos OS Release 15.1R2, the pmond process is enabled by default on the Routing Engines of MX Series routers, even when no service interfaces are configured.

Options

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. To include the **file** statement, you must specify a filename.

files *number*—(Optional) 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.

If you specify a maximum number of files, you also must specify a maximum file size with the **size** 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**—Enable all trace options flags.
- **events**—Trace process state change and cleanup events.

- **gencfg**—Trace GENCFG blobs recorded for cleanup.
- **module**—Trace module code.
- **sysvsem**—Trace SYSV semaphores recorded for cleanup.
- **sysvshm**—Trace SYSV shared memory segments recorded for cleanup.
- **tracking**—Trace tracking code.
- **ui**—Trace user interface operations.

level *level*—Specify the level of debugging output:

- **all**—Match all levels.
- **error**—Match error conditions.
- **info**—Match informational messages.
- **notice**—Match conditions that warrant special handling (but are not errors).
- **verbose**—Match verbose messages.
- **warning**—Match warning messages.

match *regex*—(Optional) Refine the output to include lines that contain the regular expression.

no-remote-trace—Disable remote tracing.

size *size*—(Optional) 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.

Syntax: **xk** to specify KB, **xm** to specify MB, or **xg** to specify GB

Range: 10 KB through 1 GB

Default: 128 KB

world-readable | no-world-readable—(Optional). Grant all users permission to read log files, or restrict the permission only to the **root** user and users who have the Junos **maintenance** permission.

Required Privilege Level

trace—To view this statement in the configuration.

trace-control—To add this statement to the configuration.

traceoptions (Resource Cleanup)

Syntax

```
traceoptions {
    file filename files number match regex size size (world-readable | no-world-readable);
    flag flag;
    level level;
    no-remote-trace;
}
```

Hierarchy Level

```
[edit system processes resource-cleanup]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Enable debugging tracing for resource cleanup process.

Options

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. To include the **file statement**, you must specify a filename.

files *number*—(Optional) 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.

If you specify a maximum number of files, you also must specify a maximum file size with the **size** 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**—Enable all trace options flags.
- **events**—Trace process state change and cleanup events.
- **gencfg—**—Trace GENCFG blobs recorded for cleanup.
- **module**—Trace module code.
- **sysvsem**—Trace SYSV semaphores recorded for cleanup.
- **sysvshm**—Trace SYSV shared memory segments recorded for cleanup.

- **tracking**—Trace tracking code.
- **ui**—Trace user interface operations.

level *level*—Specify the level of debugging output:

- **all**—Match all levels.
- **error**—Match error conditions.
- **info**—Match informational messages.
- **notice**—Match conditions that warrant special handling (but are not errors).
- **verbose**—Match verbose messages.
- **warning**—Match warning messages.

match *regex*—(Optional) Refine the output to include lines that contain the regular expression.

no-remote-trace—Disable remote tracing.

size *size*—(Optional) 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.

Syntax: **xk** to specify KB, **xm** to specify MB, or **xg** to specify GB

Range: 10 KB through 1 GB

Default: 128 KB

world-readable | no-world-readable—(Optional). Grant all users permission to read log files, or restrict the permission only to the **root** user and users who have the Junos **maintenance** permission.

Required Privilege Level

trace—To view this statement in the configuration.
trace-control—To add this statement to the configuration.

wired-max-processes

Syntax

```
wired-max-processes num-procs;
```

Hierarchy Level

```
[edit chassis fpc slot-number pic slot-number adaptive-services service-package extension-provider]
```

Release Information

Statement introduced in Junos OS Release 11.4.

Description

Configure the number of processes that use wired process memory. Performance can degrade if a process uses memory beyond its Big TLB memory. If this setting is changed, the PIC will reboot.

Options

num-procs—Number of processes that use the reserved wired process memory.

Range: 1 through 8

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

wired-process-mem-size | 898
forwarding-db-size | 872
object-cache-size | 876
policy-db-size | 880

wired-process-mem-size

Syntax

```
wired-process-mem-size mem-size;
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Configure the size of the reserved wired process memory. You can also configure object cache. If this setting is changed, the PIC reboots.

Options

megabytes—Size of the reserved wired process memory, in MB. The only size you can set for this statement is 512 MB.

Default: 512 MB

Range: 0 through 512 MB

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- forwarding-db-size | 872
- object-cache-size | 876
- policy-db-size | 880
- wired-max-processes | 897