JUNIPER NETWORKS | Engineering Simplicity

Junos® OS

FIPS Evaluated Configuration Guide for SRX345 and SRX380 Devices

Published
2022-12-28

Release
20.4R1

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

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

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

Junos® OS FIPS Evaluated Configuration Guide for SRX345 and SRX380 Devices
20.4R1
Copyright © 2022 Juniper Networks, Inc. All rights reserved.

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

About the Documentation | vi

Documentation and Release Notes | **vi**

Documentation Conventions | **vi**

Documentation Feedback | **ix**

Requesting Technical Support | **ix**

- Self-Help Online Tools and Resources | **x**
- Creating a Service Request with JTAC | **x**

1 **Junos OS in FIPS Mode of Operation for SRX Series Security Devices**

Understanding Junos OS in FIPS Mode of Operation | 12

- About the Cryptographic Boundary on Your Device | **12**
- How FIPS Mode of Operation Differs from Non-FIPS Mode of Operation | **13**
- Validated Version of Junos OS in FIPS Mode of Operation | **14**

Identifying Secure Delivery | 14

Understanding FIPS Mode of Operation Terminology and Supported Cryptographic Algorithms | 15

- FIPS Terminology | **16**
- Supported Cryptographic Algorithms | **17**

Understanding Zeroization to Clear System Data for FIPS Mode of Operation | 18

- Why Zeroize? | **19**
- When to Zeroize? | **19**

Understanding FIPS Self-Tests | 19

- Performing Power-On Self-Tests on the Device | **20**

Applying Tamper-Evident Seals to the Cryptographic Module | 24

General Tamper-Evident Seal Instructions | **24**

- Applying Tamper-Evident Seals on SRX345 Devices | **25**
- Applying Tamper-Evident Seals on SRX380 Devices | **25**

2 Configuring Roles and Authentication Methods

Downloading Software Packages from Juniper Networks (FIPS Mode) | 27

Downloading and Installing Junos Software Packages (FIPS Mode) | 27

Understanding Roles and Services for Junos OS in FIPS Mode of Operation | 28

- Cryptographic Officer Role and Responsibilities | 29
- FIPS User Role and Responsibilities | 30
- What Is Expected of All FIPS Users | 30

Understanding the Associated Password Rules for an Authorized Administrator | 31

Understanding FIPS Authentication Methods | 33

- Username and Password Authentication over the Console and SSH | 33
- Username and Public Key Authentication over SSH | 33

Understanding Services for Junos OS in FIPS Mode of Operation | 34

- Understanding Authenticated Services | 34
- Critical Security Parameters | 35

3 Configuring SSH and Console Connection

Configuring a System Login Message and Announcement | 39

Limiting the Number of User Login Attempts for SSH Sessions | 40

Configuring SSH on the Evaluated Configuration | 41

4 Configuring Junos OS in FIPS Mode of Operation

Loading Firmware on the Device | 44

How to Enable and Configure Junos OS in FIPS Mode of Operation | 44

5 Configuring Clustering Mode

Understanding Cluster Mode | 50

Configuring L2 HA Link Encryption tunnel | 50

6

Junos-FIPS Configuration Restrictions

Understanding Configuration Limitations and Restrictions on Junos OS in FIPS Mode | 57

Unsupported Junos-FIPS Configuration Statements | 58

Unsupported Junos-FIPS Operational Commands | 59

About the Documentation

IN THIS SECTION

- Documentation and Release Notes | vi
- Documentation Conventions | vi
- Documentation Feedback | ix
- Requesting Technical Support | ix

Use this guide to operate SRX Series devices in Federal Information Processing Standards (FIPS) 140-2 Level 2 environment. FIPS 140-2 defines security levels for hardware and software that perform cryptographic functions

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.

Documentation Conventions

Table 1 on page vii 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 vii 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		

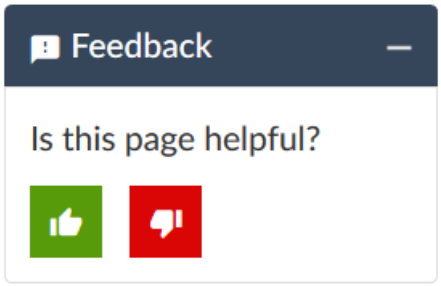
Table 2: Text and Syntax Conventions *(continued)*

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/.

CHAPTER 1

Junos OS in FIPS Mode of Operation for SRX Series Security Devices

Understanding Junos OS in FIPS Mode of Operation | **12**

Identifying Secure Delivery | **14**

Understanding FIPS Mode of Operation Terminology and Supported Cryptographic Algorithms | **15**

Understanding Zeroization to Clear System Data for FIPS Mode of Operation | **18**

Understanding FIPS Self-Tests | **19**

Applying Tamper-Evident Seals to the Cryptographic Module | **24**

Understanding Junos OS in FIPS Mode of Operation

IN THIS SECTION

- About the Cryptographic Boundary on Your Device | 12
- How FIPS Mode of Operation Differs from Non-FIPS Mode of Operation | 13
- Validated Version of Junos OS in FIPS Mode of Operation | 14

Federal Information Processing Standards (FIPS) 140-2 defines security levels for hardware and software that perform cryptographic functions. Junos FIPS mode is a version of the Junos operating system (Junos OS) that complies with Federal Information Processing Standard (FIPS) 140-2.

Operating SRX Series devices in a FIPS 140-2 Level 2 environment requires enabling and configuring FIPS mode of operation on the device from the Junos OS command-line interface (CLI).

> **NOTE:** In Junos OS Release 20.4R1, SRX345 and SRX380 devices are in progress for certification for FIPS 140-2 Level 2.

The *Cryptographic Officer* enables FIPS mode of operation in Junos OS Release 20.4R1 and sets up keys and passwords for the system and other *FIPS users* who can view the configuration. Both user types can also perform normal configuration tasks on the device (such as modify interface types) as individual user configuration allows.

> **BEST PRACTICE:** Be sure to verify the secure delivery of your device and apply tamper-evident seals to its vulnerable ports.

About the Cryptographic Boundary on Your Device

FIPS 140-2 compliance requires a defined *cryptographic boundary* around each *cryptographic module* on a device. Junos OS in FIPS mode of operation prevents the cryptographic module from running any software that is not part of the FIPS-certified distribution, and allows only FIPS-approved cryptographic algorithms

to be used. No critical security parameters (CSPs), such as passwords and keys, can cross the cryptographic boundary of the module by, for example, being displayed on a console or written to an external log file.

> **CAUTION:** Virtual Chassis features are not supported in FIPS mode of operation—they have not been tested by Juniper Networks. Do not configure a Virtual Chassis in FIPS mode of operation.

To physically secure the cryptographic module, all Juniper Networks devices require a tamper-evident seal on the USB and mini-USB ports.

How FIPS Mode of Operation Differs from Non-FIPS Mode of Operation

Unlike Junos OS in non-FIPS mode of operation, Junos OS in FIPS mode of operation is a *nonmodifiable operational environment*. In addition, Junos OS in FIPS mode of operation differs in the following ways from Junos OS in non-FIPS mode of operation:

- Self-tests of all cryptographic algorithms are performed at startup in both FIPS mode and non FIPS mode. But the results are displayed on console only in FIPS mode.
- Self-tests of random number and key generation are performed continuously.
- Weak cryptographic algorithms such as Data Encryption Standard (DES) and MD5 are disabled.
- FIPS Mode uses the HMAC-DRBG Random Number Generator, while Non-FIPS mode uses the Junos default yarrow Random Number Generator.
- Pairwise consistency test when a module generates a public and private key pair is performed only in FIPS Mode.
- DH and ECDH public key validation during generation is performed only in FIPS Mode
- Weak or unencrypted management connections must not be configured.
- Junos-FIPS administrator passwords must be at least 10 characters long.
- Cryptographic keys must be encrypted before transmission.

The FIPS 140-2 standard is available for download from the National Institute of Standards and Technology (NIST) at https://csrc.nist.gov/csrc/media/publications/fips/140/2/final/documents/fips1402.pdf.

Validated Version of Junos OS in FIPS Mode of Operation

To determine whether a Junos OS release is NIST-validated, see the compliance page on the Juniper Networks Web site (https://apps.juniper.net/compliance/fips.html).

RELATED DOCUMENTATION

Identifying Secure Delivery

Identifying Secure Delivery

There are several mechanisms provided in the delivery process to ensure that a customer receives a product that has not been tampered with. The customer should perform the following checks upon receipt of an appliance to verify the integrity of the platform:

- Shipping label—Ensure that the shipping label correctly identifies the correct customer name and address as well as the device.
- Outside packaging—Inspect the outside shipping box and tape. Ensure that the shipping tape has not been cut or otherwise compromised. Ensure that the box has not been cut or damaged to allow access to the device.
- Inside packaging—Inspect the plastic bag and seal. Ensure that the bag is not cut or removed. Ensure that the seal remains intact.

If the customer identifies a problem during the inspection, he or she should immediately contact the supplier. Provide the order number, tracking number, and a description of the identified problem to the supplier.

Additionally, there are several checks that can be performed to ensure that the customer has received a box sent by Juniper Networks and not a different company masquerading as Juniper Networks. The customer should perform the following checks upon receipt of a device to verify the authenticity of the device:

- Verify that the device was ordered using a purchase order. Juniper Networks devices are never shipped without a purchase order.
- When a device is shipped, a shipment notification is sent to the e-mail address provided by the customer when the order is taken. Verify that this e-mail notification was received. Verify that the e-mail contains the following information:

- Purchase order number
- Juniper Networks order number used to track the shipment
- Carrier tracking number used to track the shipment
- List of items shipped including serial numbers
- Address and contacts of both the supplier and the customer

- Verify that the shipment was initiated by Juniper Networks. To verify that a shipment was initiated by Juniper Networks, perform the following tasks:
  - Compare the carrier tracking number of the Juniper Networks order number listed in the Juniper Networks shipping notification with the tracking number on the package received.
  - Log in to the Juniper Networks online customer support portal at https://www.juniper.net/customers/csc/management to view the order status. Compare the carrier tracking number or the Juniper Networks order number listed in the Juniper Networks shipment notification with the tracking number on the package received.

RELATED DOCUMENTATION

Understanding Junos OS in FIPS Mode of Operation | 12

Understanding FIPS Mode of Operation Terminology and Supported Cryptographic Algorithms

IN THIS SECTION

FIPS Terminology | 16
Supported Cryptographic Algorithms | 17

Use the definitions of FIPS terms and supported algorithms to help you understand Junos OS in FIPS mode of operation.

FIPS Terminology

Critical security parameter (CSP)—Security-related information—for example, secret and private cryptographic keys and authentication data such as passwords and personal identification numbers (PINs)—whose disclosure or modification can compromise the security of a cryptographic module or the information it protects.

Cryptographic module—The set of hardware, software, and firmware that implements approved security functions (including cryptographic algorithms and key generation) and is contained within the cryptographic boundary. SRX Series devices are certified at FIPS 140-2 Level 2.

Cryptographic Officer—Person with appropriate permissions who is responsible for securely enabling, configuring, monitoring, and maintaining Junos OS in FIPS mode of operation on a device. For details, see *Understanding Roles and Services for Junos OS in FIPS Mode of Operation*.

ESP—Encapsulating Security Payload (ESP) protocol. The part of the IPsec protocol that guarantees the confidentiality of packets through encryption. The protocol ensures that if an ESP packet is successfully decrypted, and no other party knows the secret key the peers share, the packet was not wiretapped in transit.

FIPS—Federal Information Processing Standards. FIPS 140-2 specifies requirements for security and cryptographic modules. Junos OS in FIPS mode of operation complies with FIPS 140-2 Level 2.

IKE—The Internet Key Exchange (IKE) is part of IPsec and provides ways to securely negotiate the shared private keys that the authentication header (AH) and ESP portions of IPsec need to function properly. IKE employs Diffie-Hellman key-exchange methods and is optional in IPsec. (The shared keys can be entered manually at the endpoints.)

IPsec—The IP Security (IPsec) protocol. A standard way to add security to Internet communications. An IPsec security association (SA) establishes secure communication with another FIPS cryptographic module by means of mutual authentication and encryption.

KATs—Known answer tests. System self-tests that validate the output of cryptographic algorithms approved for FIPS and test the integrity of some Junos OS modules. For details, see *Understanding FIPS Self-Tests*.

SA—Security association (SA). A connection between hosts that allows them to communicate securely by defining, for example, how they exchange private keys. As Cryptographic Officer, you must manually configure an internal SA on devices running Junos OS in FIPS mode of operation. All values, including the keys, must be statically specified in the configuration.

SPI—Security parameter index (SPI). A numeric identifier used with the destination address and security protocol in IPsec to identify an SA. Because you manually configure the SA for Junos OS in FIPS mode of operation, the SPI must be entered as a parameter rather than derived randomly.

SSH—A protocol that uses strong authentication and encryption for remote access across a nonsecure network. SSH provides remote login, remote program execution, file copy, and other functions. It is

intended as a secure replacement for **rlogin**, **rsh**, and **rcp** in a UNIX environment. To secure the information sent over administrative connections, use SSHv2 for CLI configuration. In Junos OS, SSHv2 is enabled by default, and SSHv1, which is not considered secure, is disabled.

Zeroization—Erasure of all CSPs and other user-created data on a device before its operation as a FIPS cryptographic module—or in preparation for repurposing the device for non-FIPS operation. The Cryptographic Officer can zeroize the system with a CLI operational command. For details, see *Understanding Zeroization to Clear System Data for FIPS Mode of Operation*.

Supported Cryptographic Algorithms

Each implementation of an algorithm is checked by a series of known answer test (KAT) self-tests. Any self-test failure results in a FIPS error state.

The following cryptographic algorithms are supported in FIPS mode of operation. Symmetric methods use the same key for encryption and decryption, while asymmetric methods (preferred) use different keys for encryption and decryption.

AES—The Advanced Encryption Standard (AES), defined in FIPS PUB 197. The AES algorithm uses keys of 128, 192, or 256 bits to encrypt and decrypt data in blocks of 128 bits.

Diffie-Hellman—A method of key exchange across a nonsecure environment (such as the Internet). The Diffie-Hellman algorithm negotiates a session key without sending the key itself across the network by allowing each party to pick a partial key independently and send part of that key to the other. Each side then calculates a common key value. This is a symmetrical method, and keys are typically used only for a short time, discarded, and regenerated.

ECDH—Elliptic Curve Diffie-Hellman. A variant of the Diffie-Hellman key exchange algorithm that uses cryptography based on the algebraic structure of elliptic curves over finite fields. ECDH allows two parties, each having an elliptic curve public-private key pair, to establish a shared secret over an insecure channel. The shared secret can be used either as a key or to derive another key for encrypting subsequent communications using a symmetric key cipher.

ECDSA—Elliptic Curve Digital Signature Algorithm. A variant of the Digital Signature Algorithm (DSA) that uses cryptography based on the algebraic structure of elliptic curves over finite fields. The bit size of the elliptic curve determines the difficulty of decrypting the key. The public key for ECDSA is twice the size of the security strength in bits.

HMAC—Defined as Keyed-Hashing for Message Authentication in RFC 2104, HMAC combines hashing algorithms with cryptographic keys for message authentication.

3DES (3des-cbc)—Encryption standard based on the original Data Encryption Standard (DES) from the 1970s that used a 56-bit key and was cracked in 1997. The more secure 3DES is DES enhanced with

three multiple stages and effective key lengths of about 112 bits. For Junos OS in FIPS mode of operation, 3DES is implemented with cipher block chaining (CBC).

RELATED DOCUMENTATION

Understanding FIPS Self-Tests | 19

Understanding Zeroization to Clear System Data for FIPS Mode of Operation

IN THIS SECTION

- Why Zeroize? | 19
- When to Zeroize? | 19

Zeroization completely erases all configuration information on the device, including all plaintext passwords, secrets, and private keys for SSH, local encryption, local authentication, and IPsec. To exit the FIPS mode you need to zeroize the device.

The cryptographic module provides a non-approved mode of operation in which non-approved cryptographic algorithms are supported. When moving from the non-approved mode of operation to the approved mode of operation, the Cryptographic Officer must zeroize the non-approved mode critical security parameters (CSPs). The Cryptographic Officer initiates the zeroization process by entering the **request system zeroize** command from the CLI after enabling FIPS mode of operation. Use of this command is restricted to the Cryptographic Officer.

CAUTION: Perform system zeroization with care. After the zeroization process is complete, no data is left on the device. This command erases all the CSPs and configurations on the device..

Zeroization can be time-consuming. Although all configurations are removed in a few seconds, the zeroization process goes on to overwrite all media, which can take considerable time depending on the size of the media.

Why Zeroize?

Your device is not considered a valid FIPS cryptographic module until all CSPs have been entered—or reentered—while the device is in FIPS mode of operation.

When to Zeroize?

As a Cryptographic Officer, perform zeroization in the following situations:

- **Before FIPS operation**—To prepare your device for operation as a FIPS cryptographic module, perform zeroization to remove the non-approved mode critical security parameters (CSPs) and enable FIPS mode on the device.
- **Before non-FIPS operation**—To begin repurposing your device for non-FIPS operation, perform zeroization on the device.

  > **NOTE:** Juniper Networks does not support installing non-FIPS software in a FIPS mode of operation, but doing so might be necessary in certain test environments. Be sure to zeroize the system first.

- **When a tamper-evident seal is disturbed**—If the seal on an insecure port has been tampered with, the system is considered to be compromised. After applying new tamper-evident seals to the appropriate locations, zeroize the system and set up new passwords and CSPs.

Understanding FIPS Self-Tests

The cryptographic module enforces security rules to ensure that a device running the Juniper Networks Junos operating system (Junos OS) in FIPS mode of operation meets the security requirements of FIPS 140-2 Level 2. To validate the output of cryptographic algorithms approved for FIPS and test the integrity of some system modules, the device performs the following series of known answer test (KAT) self-tests:

- **kernel_kats**—KAT for kernel cryptographic routines
- **md_kats**—KAT for libmd and libc
- **openssl_kats**—KAT for OpenSSL cryptographic implementation
- **quicksec_7_0_kats**—KAT for QuickSec Toolkit cryptographic implementation

- **octcrypto_kats**—KAT for Octeon
- **JSF_Crypto_(Octeon)_KATS**—KAT for JSF crypto octeon

The KAT self-tests are performed automatically at startup and reboot, when FIPS mode of operation is enabled on the device. Conditional self-tests are also performed automatically to verify digitally signed software packages, generated random numbers, RSA and DSA key pairs, and manually entered keys.

If the KATs are completed successfully, the system log (syslog) file is updated to display the tests that were executed.

If the device fails a KAT, the device writes the details to a system log file, enters FIPS error state (panic), and reboots.

The **file show /var/log/messages** command displays the system log.

Performing Power-On Self-Tests on the Device

Each time the cryptographic module is powered on, the module tests that the cryptographic algorithms still operate correctly and that sensitive data has not been damaged. Power-on self-tests are performed on demand by power cycling the module.

On powering on or resetting the device, the module performs the following self-tests. All KATs must be completed successfully prior to any other use of cryptography by the module. If one of the KATs fail, the module enters the Critical Failure error state.

The module displays the following status output for SRX345 and SRX380 devices while running the power-on self-tests:

```
Verified jboot signed by PackageDevelopmentECP256_2020 method ECDSA256+SHA256
Verified junos signed by PackageDevelopmentECP256_2020 method ECDSA256+SHA256
veriexec: cannot update veriexec for /usr/lib/libext_db.so.3: Too many links
veriexec: cannot update veriexec for /usr/lib/libpsu.so.3: Too many links
veriexec: cannot update veriexec for /usr/lib/libxml2.so.3: Too many links
veriexec: cannot update veriexec for /usr/lib/libyaml.so.3: Too many links
veriexec: cannot update veriexec for /var/jailetc/mime.types: No such file or
directory
veriexec: cannot update veriexec for /var/jailetc/php_mod.ini: No such file or
directory
Verified junos-20.4R1 signed by PackageDevelopmentECP256_2020 method ECDSA256+SHA256
Checking integrity of BSD labels:
  s1: Passed
  s2: Passed
  s3: Passed
```

```
  s4: Passed
** /dev/bo0s3e
FILE SYSTEM CLEAN; SKIPPING CHECKS
clean, 599646 free (30 frags, 74952 blocks, 0.0% fragmentation)
** /dev/bo0s3f
FILE SYSTEM CLEAN; SKIPPING CHECKS
clean, 18789959 free (471 frags, 2348686 blocks, 0.0% fragmentation)
Checking integrity of licenses:
  DemoLabJUNOS634993695.lic: No recovery data
  DemoLabJUNOS747689902.lic: No recovery data
  DemoLabJUNOS867795690.lic: No recovery data
Checking integrity of configuration:
  rescue.conf.gz: No recovery data

LPC bus driver
lpcbus0 on cpld0
tpm0: <Trusted Platform Module> on lpcbus0
tpm: IFX SLB 9660 TT 1.2 rev 0x10
Loading configuration ...
mgd: warning: schema: dbs_remap_daemon_index: could not find daemon name 'ikemd'mgd:
 Running FIPS Self-tests
mgd: Testing JSF Crypto (Octeon) KATs:
mgd:   AES-CBC Known Answer Test:                        Passed
mgd:   AES-GCM Known Answer Test:                        Passed
mgd:   RSA-SIGN Known Answer Test:                       Passed
mgd:   ECDSA-SIGN Known Answer Test:                     Passed
mgd:   KAS-ECC-EPHEM-UNIFIED-NOKC Known Answer Test:    Passed
mgd:   KAS-FFC-EPHEM-NOKC Known Answer Test:            Passed
mgd: Testing kernel KATS:
mgd:   NIST 800-90 HMAC DRBG Known Answer Test:         Passed
mgd:   DES3-CBC Known Answer Test:                       Passed
mgd:   HMAC-SHA1 Known Answer Test:                      Passed
mgd:   HMAC-SHA2-256 Known Answer Test:                  Passed
mgd:   SHA-2-384 Known Answer Test:                      Passed
mgd:   SHA-2-512 Known Answer Test:                      Passed
mgd:   AES128-CMAC Known Answer Test:                    Passed
mgd:   AES-CBC Known Answer Test:                        Passed
mgd: Testing MACSec KATS:
mgd:   AES128-CMAC Known Answer Test:                    Passed
mgd:   AES256-CMAC Known Answer Test:                    Passed
mgd:   AES-ECB Known Answer Test:                        Passed
mgd:   AES-KEYWRAP Known Answer Test:                    Passed
mgd:   KBKDF Known Answer Test:                          Passed
mgd: Testing libmd KATS:
```

```
mgd:   HMAC-SHA1 Known Answer Test:                      Passed
mgd:   HMAC-SHA2-256 Known Answer Test:                  Passed
mgd:   SHA-2-512 Known Answer Test:                      Passed
mgd: Testing Octeon KATS:
mgd:   DES3-CBC Known Answer Test:                       Passed
mgd:   HMAC-SHA1 Known Answer Test:                      Passed
mgd:   HMAC-SHA2-256 Known Answer Test:                  Passed
mgd:   AES-CBC Known Answer Test:                        Passed
mgd: Testing OpenSSL KATS:
mgd:   NIST 800-90 HMAC DRBG Known Answer Test:          Passed
mgd:   FIPS ECDSA Known Answer Test:                     Passed
mgd:   FIPS ECDH Known Answer Test:                      Passed
mgd:   FIPS RSA Known Answer Test:                       Passed
mgd:   DES3-CBC Known Answer Test:                       Passed
mgd:   HMAC-SHA1 Known Answer Test:                      Passed
mgd:   HMAC-SHA2-224 Known Answer Test:                  Passed
mgd:   HMAC-SHA2-256 Known Answer Test:                  Passed
mgd:   HMAC-SHA2-384 Known Answer Test:                  Passed
mgd:   HMAC-SHA2-512 Known Answer Test:                  Passed
mgd:   AES-CBC Known Answer Test:                        Passed
mgd:   AES-GCM Known Answer Test:                        Passed
mgd:   ECDSA-SIGN Known Answer Test:                     Passed
mgd:   KDF-IKE-V1 Known Answer Test:                     Passed
mgd:   KDF-SSH-SHA256 Known Answer Test:                 Passed
mgd:   KAS-ECC-EPHEM-UNIFIED-NOKC Known Answer Test:     Passed
mgd:   KAS-FFC-EPHEM-NOKC Known Answer Test:             Passed
mgd: Testing QuickSec 7.0 KATS:
mgd:   NIST 800-90 HMAC DRBG Known Answer Test:          Passed
mgd:   DES3-CBC Known Answer Test:                       Passed
mgd:   HMAC-SHA1 Known Answer Test:                      Passed
mgd:   HMAC-SHA2-224 Known Answer Test:                  Passed
mgd:   HMAC-SHA2-256 Known Answer Test:                  Passed
mgd:   HMAC-SHA2-384 Known Answer Test:                 Passedmgd:   HMAC-SHA2-512
 Known Answveriexec: no fingerprint for file='/sbin/kats/cannot-exec' fsid=83
fileid=5048524 gen=1 uid=0 pid=1073
er Test:                 Passed
mgd:   AES-CBC Known Answer Test:                        Passed
mgd:   AES-GCM Known Answer Test:                        Passed
mgd:   SSH-RSA-ENC Known Answer Test:                    Passed
mgd:   SSH-RSA-SIGN Known Answer Test:                   Passed
mgd:   SSH-ECDSA-SIGN Known Answer Test:                 Passed
mgd:   KDF-IKE-V1 Known Answer Test:                     Passed
mgd:   KDF-IKE-V2 Known Answer Test:                     Passed
mgd: Testing QuickSec KATS:
```

```
mgd:   NIST 800-90 HMAC DRBG Known Answer Test:         Passed
mgd:   DES3-CBC Known Answer Test:                      Passed
mgd:   HMAC-SHA1 Known Answer Test:                     Passed
mgd:   HMAC-SHA2-224 Known Answer Test:                 Passed
mgd:   HMAC-SHA2-256 Known Answer Test:                 Passed
mgd:   HMAC-SHA2-384 Known Answer Test:                 Passed
mgd:   HMAC-SHA2-512 Known Answer Test:                 Passed
mgd:   AES-CBC Known Answer Test:                       Passed
mgd:   AES-GCM Known Answer Test:                       Passed
mgd:   SSH-RSA-ENC Known Answer Test:                   Passed
mgd:   SSH-RSA-SIGN Known Answer Test:                  Passed
mgd:   KDF-IKE-V1 Known Answer Test:                    Passed
mgd:   KDF-IKE-V2 Known Answer Test:                    Passed
mgd: Testing SSH IPsec KATS:
mgd:   NIST 800-90 HMAC DRBG Known Answer Test:         Passed
mgd:   DES3-CBC Known Answer Test:                      Passed
mgd:   HMAC-SHA1 Known Answer Test:                     Passed
mgd:   HMAC-SHA2-256 Known Answer Test:                 Passed
mgd:   AES-CBC Known Answer Test:                       Passed
mgd:   SSH-RSA-ENC Known Answer Test:                   Passed
mgd:   SSH-RSA-SIGN Known Answer Test:                  Passed
mgd:   KDF-IKE-V1 Known Answer Test:                    Passed
mgd: Testing file integrity:
mgd:   File integrity Known Answer Test:                Passed
mgd: Testing crypto integrity:
mgd:   Crypto integrity Known Answer Test:              Passed
mgd: Expect an exec Authentication error...
mgd: /sbin/kats/run-tests: /sbin/kats/cannot-exec: Authentication error
mgd: FIPS Self-tests Passed
```

NOTE: The module implements cryptographic libraries and algorithms that are not utilized in the approved mode of operation.

RELATED DOCUMENTATION

How to Enable and Configure Junos OS in FIPS Mode of Operation

Applying Tamper-Evident Seals to the Cryptographic Module

IN THIS SECTION

- General Tamper-Evident Seal Instructions | 24

The cryptographic modules physical embodiment is that of a multi-chip standalone device that meets Level 2 physical security requirements. The module is completely enclosed in a rectangular cold rolled steel enclosure. There are no ventilation holes, gaps, slits, cracks, slots, or crevices that would allow for any sort of observation of any component contained within the cryptographic boundary. Tamper-evident seals allow the operator to verify if the enclosure has been breached. These seals are not factory-installed and must be applied by the Cryptographic Officer.

NOTE: Seals are available for order from Juniper Networks using part number JNPR-FIPS-TAMPER-LBLS.

As a Cryptographic Officer, you are responsible for:

- Applying seals to secure the cryptographic module
- Controlling any unused seals
- Controlling and observing any changes, such as repairs or booting from an external USB drive to the cryptographic module, that require removing or replacing the seals to maintain the security of the module

As per the security inspection guidelines, upon receipt of the cryptographic module, the Cryptographic Officer must check that the labels are free of any tamper evidence.

General Tamper-Evident Seal Instructions

All FIPS-certified switches require a tamper-evident seal on the USB ports. While applying seals, follow these general instructions:

- Handle the seals with care. Do not touch the adhesive side. Do not cut or otherwise resize a seal to make it fit.

- Make sure all surfaces to which the seals are applied are clean and dry and clear of any residue.
- Apply the seals with firm pressure across the seal to ensure adhesion. Allow at least 24 hours for the adhesive to cure.

Applying Tamper-Evident Seals on SRX345 Devices

On SRX345 devices, apply 27 tamper-evident seals at the following locations:

1. Apply five seals at the top of the chassis, covering one of the five chassis screws.
2. Apply four seals on the I/O slots.
3. Apply two seals on the rare panel, covering the blank faceplate and the SSD.
4. Apply 16 seals, on the side panels over the screw holes.

Applying Tamper-Evident Seals on SRX380 Devices

On SRX380 devices, apply tamper-evident seals at the following locations:

1. Apply four seals on the front I/O slots.
2. Apply five seals at the top of the chassis, covering one of the five chassis screws.
3. Apply two seals at the front of the chassis on either side of the LED matrix on the right of the device.
4. Apply two seals on the rear panel, covering the blank faceplate. If the grounding connection is not used, apply a seal across this as well.

RELATED DOCUMENTATION

How to Enable and Configure Junos OS in FIPS Mode of Operation | 44

2

CHAPTER

Configuring Roles and Authentication Methods

Downloading Software Packages from Juniper Networks (FIPS Mode) | **27**

Downloading and Installing Junos Software Packages (FIPS Mode) | **27**

Understanding Roles and Services for Junos OS in FIPS Mode of Operation | **28**

Understanding the Associated Password Rules for an Authorized Administrator | **31**

Understanding FIPS Authentication Methods | **33**

Understanding Services for Junos OS in FIPS Mode of Operation | **34**

Downloading Software Packages from Juniper Networks (FIPS Mode)

You can download the following Junos OS software packages from the Juniper Networks website:

- Junos OS for SRX345, Release 20.4R1
- Junos OS for SRX380, Release 20.4R1

Before you begin to download the software, ensure that you have a Juniper Networks Web account and a valid support contract. To obtain an account, complete the registration form at the Juniper Networks website: https://userregistration.juniper.net/entitlement/setupAccountInfo.do.

To download software packages from Juniper Networks:

1. Using a Web browser, follow the links to the download URL on the Juniper Networks webpage.

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

2. Log in to the Juniper Networks authentication system using the username (generally your e-mail address) and password supplied by Juniper Networks representatives.

3. Download the software. See Downloading Software

RELATED DOCUMENTATION

Installation and Upgrade Guide

Downloading and Installing Junos Software Packages (FIPS Mode)

SRX Series devices can provide the security defined by Federal Information Processing Standards (FIPS) 140-2 Level 2 if these devices are operated in the Junos OS in FIPS mode. To operate in Junos OS in FIPS mode, the device must have the following software packages installed:

- Junos OS for SRX345, Release 20.4R1
- Junos OS for SRX380, Release 20.4R1

To download software packages from Juniper Networks:

1. Using a Web browser, follow the links to the download URL on the Juniper Networks webpage.

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

2. Log in to the Juniper Networks authentication system using the username (generally your e-mail address) and password supplied by Juniper Networks representatives.

3. Download the software. See Downloading Software

RELATED DOCUMENTATION

- Installation and Upgrade Guide

Understanding Roles and Services for Junos OS in FIPS Mode of Operation

IN THIS SECTION

- Cryptographic Officer Role and Responsibilities | **29**
- FIPS User Role and Responsibilities | **30**
- What Is Expected of All FIPS Users | **30**

The Juniper Networks Junos operating system (Junos OS) running in non-FIPS mode of operation allows a wide range of capabilities for users, and authentication is identity-based. In contrast, the FIPS 140-2 standard defines two user roles: *Cryptographic Officer* and *FIPS user*. These roles are defined in terms of Junos OS user capabilities.

All other user types defined for Junos OS in FIPS mode of operation (operator, administrative user, and so on) must fall into one of the two categories: Cryptographic Officer or FIPS user. For this reason, user authentication in FIPS mode of operation is role-based rather than identity-based.

In addition to their FIPS roles, both user types can perform normal configuration tasks on the device as individual user configuration allows.

Cryptographic Officers and FIPS users perform all FIPS-related configuration tasks and issue all statements and commands for Junos OS in FIPS mode of operation. Cryptographic Officer and FIPS user configurations must follow the guidelines for Junos OS in FIPS mode of operation.

Cryptographic Officer Role and Responsibilities

The Cryptographic Officer is the person responsible for enabling, configuring, monitoring, and maintaining Junos OS in FIPS mode of operation on a device. The Cryptographic Officer securely installs Junos OS on the device, enables FIPS mode of operation, establishes keys and passwords for other users and software modules, and initializes the device before network connection. The Cryptographic Officer can configure and monitor the module through a console or SSH connection.

> **BEST PRACTICE:** We recommend that the Cryptographic Officer administer the system in a secure manner by keeping passwords secure and checking audit files.

The permissions that distinguish the Cryptographic Officer from other FIPS users are **secret**, **security**, **maintenance**, and **control**. For FIPS compliance, assign the Cryptographic Officer to a login class that contains all of these permissions. A user with the Junos OS maintenance permission can read files containing critical security parameters (CSPs).

> **NOTE:** Junos OS in FIPS mode of operation does not support the *FIPS 140-2 maintenance role*, which is different from the Junos OS maintenance permission.

Among the tasks related to Junos OS in FIPS mode of operation, the Cryptographic Officer is expected to:

- Set the initial root password.
- Reset user passwords for FIPS-approved algorithms during upgrades from Junos OS.
- Set up manual IPsec SAs for configuration with dual Routing Engines.
- Examine log and audit files for events of interest.
- Erase user-generated files and data on (zeroize) the device.

FIPS User Role and Responsibilities

All FIPS users, including the Cryptographic Officer, can view the configuration. Only the user assigned as the Cryptographic Officer can modify the configuration.

The permissions that distinguish Cryptographic Officers from other FIPS users are **secret**, **security**, **maintenance**, and **control**. For FIPS compliance, assign the FIPS user to a class that contains *none* of these permissions.

FIPS users configure networking features on the device and perform other tasks that are not specific to FIPS mode of operation. FIPS users who are not Cryptographic Officers can perform reboots and view status output.

What Is Expected of All FIPS Users

All FIPS users, including the Cryptographic Officer, must observe security guidelines at all times.

All FIPS users must:

- Keep all passwords confidential.
- Store devices and documentation in a secure area.
- Deploy devices in secure areas.
- Check audit files periodically.
- Conform to all other FIPS 140-2 security rules.
- Follow these guidelines:
  - Users are trusted.
  - Users abide by all security guidelines.
  - Users do not deliberately compromise security.
  - Users behave responsibly at all times.

RELATED DOCUMENTATION

Understanding Zeroization to Clear System Data for FIPS Mode of Operation | 18

Understanding the Associated Password Rules for an Authorized Administrator

The authorized administrator is associated with a defined login class, and the administrator is assigned with all permissions. Data is stored locally for fixed password authentication.

> **NOTE:** We recommend that you not use control characters in passwords.

Use the following guidelines and configuration options for passwords and when selecting passwords for authorized administrator accounts. Passwords should be:

- Easy to remember so that users are not tempted to write it down.
- Changed periodically.
- Private and not shared with anyone.
- Contain a minimum of 10 characters. The minimum password length is 10 characters.

```
[ edit ]
administrator@host# set system login password minimum-length 10
```

- Include both alphanumeric and punctuation characters, composed of any combination of upper and lowercase letters, numbers, and special characters such as, “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”. There should be at least a change in one case, one or more digits, and one or more punctuation marks.
- Contain character sets. Valid character sets include uppercase letters, lowercase letters, numbers, punctuation, and other special characters.

```
[ edit ]
administrator@host# set system login password change-type character-sets
```

- Contain the minimum number of character sets or character set changes. The minimum number of character sets required in plain-text passwords in Junos FIPS is 2.

```
[ edit ]
administrator@host# set system login password minimum-changes 2
```

NOTE: The authentication algorithm for plain-text passwords must be configured as sha256.

```
[ edit ]
administrator@host# set system login password format sha256
```

When you change the password algorithm to SHA256, change even the user password. Until then, the old hash algorithm is used.

Weak passwords are:

- Words that might be found in or exist as a permuted form in a system file such as **/etc/passwd**.
- The hostname of the system (always a first guess).
- Any words appearing in a dictionary. This includes dictionaries other than English, and words found in works such as Shakespeare, Lewis Carroll, Roget's Thesaurus, and so on. This prohibition includes common words and phrases from sports, sayings, movies, and television shows.
- Permutations on any of the above. For example, a dictionary word with vowels replaced with digits (for example f00t) or with digits added to the end.
- Any machine-generated passwords. Algorithms reduce the search space of password-guessing programs and so should not be used.

Strong reusable passwords can be based on letters from a favorite phrase or word, and then concatenated with other, unrelated words, along with additional digits and punctuation.

NOTE: Passwords should be changed periodically.

RELATED DOCUMENTATION

Understanding Junos OS in FIPS Mode of Operation | 12

Identifying Secure Delivery | 14

Understanding FIPS Authentication Methods

The Juniper Networks Junos operating system (Junos OS) running in FIPS mode of operation allows a wide range of capabilities for users, and authentication is identity-based. The following types of identity-based authentication are supported in the FIPS mode of operation:

- Username and Password Authentication over the Console and SSH on page 33
- Username and Public Key Authentication over SSH on page 33

Username and Password Authentication over the Console and SSH

In this authentication method, the user is requested to enter the username and password. The device enforces the user to enter a minimum of 10 characters password that is chosen from the 96 human-readable ASCII characters.

> **NOTE:** The maximum password length is 20 characters.

In this method, the device enforces a timed access mechanism—for example, first two failed attempts to enter the correct password (assuming 0 time to process), no timed access is enforced. When the user enters the password for the third time, the module enforces a 5 second delay. Each failed attempt thereafter results in an additional 5 second delay above the previous failed attempt. For example, if the fourth failed attempt is a 10 second delay, then the fifth failed attempt is a 15 second delay, the sixth failed attempt is a 20 second delay, and the seventh failed attempt is a 25 second delay.

Therefore, this leads to a maximum of seven possible attempts in a 1 minute period for each getty active terminal. So, the best approach for the attacker would be to disconnect after 4 failed attempts, and wait for a new getty to be spawned. This would allow the attacker to perform roughly 9.6 attempts per minute (576 attempts per hour or 60 minutes). This would be rounded off to 9 attempts per minute, because there is no such thing as 0.6 attempts. Thus the probability of a successful random attempt is 1/9610, which is less than 1/1 million. The probability of a success with multiple consecutive attempts in a 1 minute period is 9/(9610), which is less than 1/100,000.

Username and Public Key Authentication over SSH

In SSH public key authentication, you provide the username and validate the ownership of the private key corresponding to the public key stored on the server. The device supports ECDSA (P-256, P-384, and

P-521) and RSA (2048, 3072, and 4092 modulus bit length) key-types. The probability of a success with multiple consecutive attempts in a 1-minute period is 5.6e7/(2128).

NOTE: The ssh-rsa authentication method is one of the allowed algorithms in FIPS mode.

Understanding Services for Junos OS in FIPS Mode of Operation

All services implemented by the module are listed in the tables that follow.

Understanding Authenticated Services

Table 3 on page 34 lists the authenticated services on the device running Junos OS.

Table 3: Authenticated services

Authenticated Services	Description	Cryptographic Officer	User (read-only)	User (network)
Configure security	Security relevant configuration	x	–	–
Configure	Non-security relevant configuration	x	–	–
Secure traffic	IPsec protected routing	–	–	x
Status	Display the status	x	x	–
Zeroize	Destroy all critical security parameters (CSPs)	x	–	–

Table 3: Authenticated services *(continued)*

Authenticated Services	Description	Cryptographic Officer	User (read-only)	User (network)
SSH connect	Initiate SSH connection for SSH monitoring and control (CLI)	x	x	–
IPsec connect	Initiate IPsec connection (IKE)	x	–	x
Console access	Console monitoring and control (CLI)	x	x	–
Remote reset	Software-initiated reset	x	–	–

Table 4: Unauthenticated traffic

Service	Description
Local reset	Hardware reset or power cycle
Traffic	Traffic requiring no cryptographic services

Critical Security Parameters

Critical security parameters (CSPs) are security-related information such as cryptographic keys and passwords that can compromise the security of the cryptographic module or the security of the information protected by the module if they are disclosed or modified.

Zeroization of the system erases all traces of CSPs in preparation for operating the device as a cryptographic module.

Table 5 on page 36 lists the CSP access rights within services.

Table 5: CSP Access Rights Within Services

Service	CSPs					
	DRBG_Seed	DRBG_State	SSH PHK	SSH DH	SSH-SEK	ESP-SEK
Configure security	–	E	G, W	–	–	–
Configure	–	–	–	–	–	–
Secure Traffic	–	–	–	–	–	E
Status	–	–	–	–	–	–
Zeroize	Z	Z	Z	Z	Z	Z
SSH connect	–	E	E	G, E	G, E	–
IPSec connect	–	E	–	–	–	G
Console access	–	–	–	–	–	–
Remote reset	G, E	G	–	Z	Z	Z
Local Reset	G, E	G	–	Z	Z	Z
Traffic	–	–	–	–	–	–

Service	CSPs				
	IKE-PSK	IKE-Priv	IKE-SKEYI	IKE-SKE	IKE-DH-PRI
Configure security	W	G, W	–	–	–
Configure	–	–	–	–	–
Secure Traffic	–	–	–	E	–
Status	–	–	–	–	–
Zeroize	Z	Z	–	–	–
SSH connect	–	–	–	–	–
IPSec connect	E	E	G	G	G

Service	CSPs				
	IKE-PSK	IKE-Priv	IKE-SKEYI	IKE-SKE	IKE-DH-PRI
Console access	–	–	–	–	–
Remote reset	–	–	Z	Z	Z
Local Reset	–	–	Z	Z	Z
Traffic	–	–	–	–	–

Here:

- G = Generate: The device generates the CSP.
- E = Execute: The device runs using the CSP.
- W = Write: The CSP is updated or written to the device.
- Z = Zeroize: The device zeroizes the CSP.

RELATED DOCUMENTATION

Understanding Zeroization to Clear System Data for FIPS Mode of Operation | 18

Understanding FIPS Authentication Methods | 33

3
CHAPTER

Configuring SSH and Console Connection

Configuring a System Login Message and Announcement | **39**

Limiting the Number of User Login Attempts for SSH Sessions | **40**

Configuring SSH on the Evaluated Configuration | **41**

Configuring a System Login Message and Announcement

A system login message appears before the user logs in and a system login announcement appears after the user logs in. By default, no login message or announcement is displayed on the device.

To configure a system login message, use the following command:

```
[edit]
user@host# set system login message login-message-banner-text
```

To configure system announcement, use the following command:

```
[edit]
user@host# set system login announcement system-announcement-text
```

NOTE:

- If the message text contains any spaces, enclose it in quotation marks.
- You can format the message using the following special characters:
  - \n—New line
  - \t—Horizontal tab
  - \'—Single quotation mark
  - \"—Double quotation mark
  - \\—Backslash

RELATED DOCUMENTATION

Configuring SSH on the Evaluated Configuration | 41

Limiting the Number of User Login Attempts for SSH Sessions

A remote administrator may login to a device through SSH. Administrator credentials are stored locally on the device. If the remote administrator presents a valid username and password, access to the TOE is granted. If the credentials are invalid, the TOE allows the authentication to be retried after an interval that starts after 1 second and increases exponentially. If the number of authentication attempts exceed the configured maximum, no authentication attempts are accepted for a configured time interval. When the interval expires, authentication attempts are again accepted.

You can configure the device to limit the number of attempts to enter a password while logging through SSH. Using the following command, the connection can be terminated if a user fails to login after a specified number of attempts:

```
[edit system login]
user@host# set retry options tries-before-disconnect <number>
```

Here, **tries-before-disconnect** is the number of times a user can attempt to enter a password when logging in. The connection closes if a user fails to log in after the number specified. The range is from 1 through 10, and the default value is 10.

You can also configure a delay, in seconds, before a user can try to enter a password after a failed attempt.

```
[edit system login]
user@host# set retry options backoff-threshold <number>
```

Here, **backoff-threshold** is the threshold for the number of failed login attempts before the user experiences a delay in being able to enter a password again. Use the **backoff-factor** option to specify the length of the delay in seconds. The range is from 1 through 3, and the default value is 2 seconds.

In addition, the device can be configured to specify the threshold for the number of failed attempts before the user experiences a delay in entering the password again.

```
[edit system login]
user@host# set retry options backoff-factor <number>
```

Here, **backoff-factor** is the length of time, in seconds, before a user can attempt to log in after a failed attempt. The delay increases by the value specified for each subsequent attempt after the threshold. The range is from 5 through 10, and the default value is 5 seconds.

RELATED DOCUMENTATION

Configuring SSH on the Evaluated Configuration | 41

Configuring SSH on the Evaluated Configuration

SSH is an allowed remote management interface in the evaluated configuration. This topic describes how to configure SSH on the device.

Before you begin, log in with your root account on the device running Junos OS Release 20.4R1 and edit the configuration.

NOTE: The commands shown configure SSH to use all of the allowed cryptographic algorithms. You can enter the configuration commands in any order and commit all the commands at once. You may decide to leave some of the algorithms out.

NOTE: You can enter the configuration commands in any order and commit all the commands at once.

To configure SSH on the TOE:

1. Specify the permissible SSH host-key algorithms.

```
[edit system services ssh]
user@host# set hostkey-algorithm ssh-ecdsa
user@host# set hostkey-algorithm ssh-rsa
```

2. Specify the SSH key-exchange algorithms.

```
[edit system services ssh]
user@host#set key-exchange [ ecdh-sha2-nistp256 ecdh-sha2-nistp384 ecdh-sha2-nistp521 ]
```

3. Specify all the permissible message authentication code algorithms.

```
[edit system services ssh]
```

```
user@host#set macs [ hmac-sha1 hmac-sha2-256 hmac-sha2-512 ]
```

4. Specify the ciphers allowed for protocol version 2.

```
[edit system services ssh]
user@host#set ciphers [aes128-cbc aes128-ctr aes192-cbc aes192-ctr aes256-cbc aes256-ctr]
```

RELATED DOCUMENTATION

- Understanding FIPS Authentication Methods | 33
- How to Enable and Configure Junos OS in FIPS Mode of Operation | 44
- Limiting the Number of User Login Attempts for SSH Sessions | 40

4

CHAPTER

Configuring Junos OS in FIPS Mode of Operation

Loading Firmware on the Device | 44

How to Enable and Configure Junos OS in FIPS Mode of Operation | 44

Loading Firmware on the Device

The Junos OS 20.4R1 FIPS images only accept the firmware signed with ECDSA and rejects any firmware signed with RSA+SHA1. You cannot downgrade to images that are signed with RSA+SHA1 from "ECDSA signed only" images. In this scenario, the SRX Series device does not load the firmware.

RELATED DOCUMENTATION

How to Enable and Configure Junos OS in FIPS Mode of Operation | 44

How to Enable and Configure Junos OS in FIPS Mode of Operation

You, as Cryptographic Officer, can enable and configure Junos OS in FIPS mode of operation on your device. Before you begin enabling and configuring FIPS mode of operation on the device:

- Verify the secure delivery of your device. See *Identifying Secure Delivery*.
- Apply tamper-evident seals. See "Applying Tamper-Evident Seals to the Cryptographic Module" on page 24.

To enable the Junos OS in FIPS mode of operation, perform the following steps:

1. Zeroize the device before enabling FIPS mode of operation

   user@host> request system zerioze

2. Enable the FIPS mode on the device.

   user@host# set system fips level 2

3. Remove the CSPs on commit check and reboot the device.

   user@host# commit

4. Run integrity and self-tests on powering on the device when the module is operating in FIPS mode.

5. Configure IKEv2 when AES-GCM is used for encryption of IKE and/or IPSec.

```
user@host# set security ike proposal <ike_proposal_name> encryption-algorithm ?
Possible completions:
3des-cbc 3DES-CBC encryption algorithm
aes-128-cbc AES-CBC 128-bit encryption algorithm
aes-128-gcm AES-GCM 128-bit encryption algorithm
aes-192-cbc AES-CBC 192-bit encryption algorithm
aes-256-cbc AES-CBC 256-bit encryption algorithm
aes-256-gcm AES-GCM 256-bit encryption algorithm
user@host# set security ike proposal <ike_proposal_name> encryption-algorithm aes-256-gcm
user@host# set security ipsec proposal <ipsec_proposal_name> encryption-algorithm ?
Possible completions:
3des-cbc 3DES-CBC encryption algorithm
aes-128-cbc AES-CBC 128-bit encryption algorithm
aes-128-gcm AES-GCM 128-bit encryption algorithm
aes-192-cbc AES-CBC 192-bit encryption algorithm
aes-192-gcm AES-GCM 192-bit encryption algorithm
aes-256-cbc AES-CBC 256-bit encryption algorithm
aes-256-gcm AES-GCM 256-bit encryption algorithm
user@host# set security ipsec proposal <ipsec_proposal_name> encryption-algorithm aes-128-gcm
user@host# set security ike gateway <gateway_name> version ?
Possible completions:
v1-only The connection must be initiated using IKE version 1
v2-only The connection must be initiated using IKE version 2
user@host# set security ike gateway <gateway_name> version v2-only
user@host# commit
commit complete
```

6. Ensure that the backup image of the firmware is also a JUNOS-FIPS image by issuing the **request system snapshot** command.

NOTE: The **show configuration security ike** and **show configuration security ipsec** commands display the approved and configured IKE/IPsec configuration for the device operating in FIPS-approved mode.

```
user@host:fips> show version
Hostname: host-srx380
Model: srx380-poe-ac
Junos: 20.4R1
JUNOS Software Release [20.4R1]
```

The **fips** keyword next to the **hostname** in the output indicates that the module is operating in FIPS mode for Junos Software Release 20.4R1.

```
user@host:fips> show configuration security ike
proposal ike-proposal1 {
    authentication-method pre-shared-keys;
    dh-group group14;
    encryption-algorithm aes-256-gcm;
}
policy ike-policy1 {
    mode main;
    proposals ike-proposal1;
   pre-shared-key ascii-text "$9$Hq.5zF/tpBUj9Au0IRdbwsaZ"; ## SECRET-DATA
}
gateway gw1 {
    ike-policy ike-policy1;
    address 198.51.100.0;
    local-identity inet 203.0.113.0;
    external-interface ge-0/0/3;
    version v2-only;
}
```

```
user@host:fips> show configuration security ipsec
proposal ipsec-proposal1 {
    protocol esp;
    encryption-algorithm aes-128-gcm;
}
policy ipsec-policy1 {
    perfect-forward-secrecy {
```

```
        keys group14;
    }
    proposals ipsec-proposal1;
}
vpn vpn1 {
    bind-interface st0.0;
    ike {
        gateway gw1;
        ipsec-policy ipsec-policy1;
    }
}
```

RELATED DOCUMENTATION

- *Loading Firmware on the Device*

5
CHAPTER

Configuring Clustering Mode

Understanding Cluster Mode | 50

Configuring L2 HA Link Encryption tunnel | 50

Understanding Cluster Mode

The Administrator of the TOE might set up the cluster mode for High Availability (HA) by connecting ge-0/0/1 on node 0 to ge-0/0/1 on node 1. The factory-default configuration does not include HA configuration. To enable HA, if the physical interfaces used by HA have some configurations, these configurations need to be removed. The two hosts constituting a chassis cluster must have identical configuration except for one being configured to node 0 and the other to node 1.

The TOE has a dedicated physical interface which acts as the fxp0 interface for the HA management of the TOE. The fxp1 interface for HA control link is ge-0/0/1. The fabric interface may be defined by the Administrator. After the cluster has been defined and set up by the Administrator, the two devices constituting a chassis cluster have identical cluster-id but different node ID as one host must be node 0 and the other one node 1. For SRX345 and SRX380 devices, the ge-0/0/1 interface on node 1 changes to ge-5/0/1.

The node 1 renumbers its interfaces by adding the total number of system FPCs to the original FPC number of the interface. The renumbering constant for SRX345 and SRX380 is 5 which results in the node 0 interface name remaining ge-0/0/0 and the node 1 interface name being renamed to ge-5/0/0. The fabric interface remains Administrator-defined.

With L2 HA link encryption tunnel, any Security Sensitive Parameters (Critical Security Parameters) exchanged over the control link between the two chassis in cluster mode is protected using IPsec. Using IPsec for internal communication between nodes, the configuration information and IKE HA messages that passes through the chassis cluster link from the primary node to the secondary node is protected from active and passive eavesdropping. Without the internal IPsec key, an attacker cannot gain privilege access or observe traffic.

RELATED DOCUMENTATION

Configuring L2 HA Link Encryption tunnel

- Physically connect the two devices and ensure that they are the same models.
- Connect the dedicated control ports on node 0 and node 1.
- Connect the user defined fabricated ports on node 0 and node 1.

To configure two chassis in cluster mode, follow the below steps.

1. Zeroize both the SRX devices before you use for cluster. If the devices are already in cluster mode please make sure you disable them before zeroize.

   For information on how to disable chassis cluster, see Disabling a Chassis Cluster

2. Delete the web management services.

   user@host# delete system services web-management https

3. Configure FIPS mode and bring up the devices in FIPS mode.

   [edit]
   user@host# **set groups global system fips level 2**

   [edit]
   user@host# **set groups global system root-authentication plain-textpassword**
   New password: type password here
   Retype new password: retype password here

   [edit]
   user@host# **commit**
   user@host> **request system reboot**

4. Configure device 1 with standard cluster commands for operating in cluster mode as node0.

   [edit]
   user@host# **set groups node0 system host-name *node0-host-name***
   user@host# **set groups node0 system backup-router *gateway-address***
   user@host# **set groups node0 system backup-router destination *value***
   user@host# **set groups node0 interfaces fxp0 unit 0 family inet address *node0-ip-address***
   user@host# **set groups node1 system host-name *node1-host-name***
   user@host# **set groups node1 system backup-router *gateway-address***
   user@host# **set groups node1 system backup-router destination *value***
   user@host# **set groups node1 interfaces fxp0 unit 0 family inet address *node1-ip-address***
   user@host# **set apply-groups global**
   user@host# **set apply-groups "$(node)**
   user@host# **delete apply-groups re0**
   user@host# **set system ports console log-out-on-disconnect**
   user@host# **set chassis cluster reth-count 5**
   user@host# **set chassis cluster redundancy-group 0 node 0 priority 254**
   user@host# **set chassis cluster redundancy-group 0 node 1 priority 1**
   user@host# **commit**
   user@host> **set chassis cluster cluster-id 1 node 0 reboot**

See
https://www.juniper.net/documentation/en_US/junos/topics/topic-map/security-chassis-cluster-verification.html

5. After the device 1 is up, configure HA link encryption as shown in sample configuration below, commit and reboot. device 1 needs to be configured with both node0 and node1 HA link encryption configuration before commit and reboot.

```
[edit]
user@host# set groups node0 security ike traceoptions file ikelog
user@host# set groups node0 security ike traceoptions file size 100m
user@host# set groups node0 security ike traceoptions flag all
user@host# set groups node0 security ike traceoptions level 15
user@host# set groups node0 security ike proposal IKE_PROP_PSK authentication-method
    pre-shared-keys
user@host# set groups node0 security ike proposal IKE_PROP_PSK dh-group group20
user@host# set groups node0 security ike proposal IKE_PROP_PSK authentication-algorithm
    sha-256
user@host# set groups node0 security ike proposal IKE_PROP_PSK encryption-algorithm aes-256-cbc
user@host# set groups node0 security ike policy IKE_POL_PSK proposals IKE_PROP_PSK
user@host# prompt groups node0 security ike policy IKE_POL_PSK pre-shared-key ascii-text
New ascii-text (secret): juniper
Retype new ascii-text (secret): juniper

user@host# set groups node0 security ike gateway S2S_GW ike-policy IKE_POL_PSK
user@host# set groups node0 security ike gateway S2S_GW version v2-only
user@host# set groups node0 security ipsec proposal IPSEC_PROP_PSK protocol esp
user@host# set groups node0 security ipsec proposal IPSEC_PROP_PSK authentication-algorithm
    hmac-sha1-96
user@host# set groups node0 security ipsec proposal IPSEC_PROP_PSK encryption-algorithm
    aes-256-cbc
user@host# set groups node0 security ipsec proposal IPSEC_PROP_PSK lifetime-seconds 200
user@host# set groups node0 security ipsec policy IPSEC_POL_PSK perfect-forward-secrecy keys
    group20
user@host# set groups node0 security ipsec policy IPSEC_POL_PSK proposals IPSEC_PROP_PSK
user@host# set groups node0 security ipsec vpn S2S_VPN ha-link-encryption
user@host# set groups node0 security ipsec vpn S2S_VPN ike gateway S2S_GW
user@host# set groups node0 security ipsec vpn S2S_VPN ike ipsec-policy IPSEC_POL_PSK
user@host# set groups node1 security ike traceoptions file ikelog
user@host# set groups node1 security ike traceoptions file size 100m
user@host# set groups node1 security ike traceoptions flag all
user@host# set groups node1 security ike traceoptions level 15
user@host# set groups node1 security ike proposal IKE_PROP_PSK authentication-method
    pre-shared-keys
user@host# set groups node1 security ike proposal IKE_PROP_PSK dh-group group20
```

user@host# **set groups node1 security ike proposal IKE_PROP_PSK authentication-algorithm sha-256**
user@host# **set groups node1 security ike proposal IKE_PROP_PSK encryption-algorithm aes-256-cbc**
user@host# **set groups node1 security ike policy IKE_POL_PSK proposals IKE_PROP_PSK**
user@host# **prompt groups node1 security ike policy IKE_POL_PSK pre-shared-key ascii-text**
New ascii-text (secret): juniper
Retype new ascii-text (secret): juniper

user@host# **set groups node1 security ike gateway S2S_GW ike-policy IKE_POL_PSK**
user@host# **set groups node1 security ike gateway S2S_GW version v2-only**
user@host# **set groups node1 security ipsec proposal IPSEC_PROP_PSK protocol esp**
user@host# **set groups node1 security ipsec proposal IPSEC_PROP_PSK authentication-algorithm hmac-sha1-96**
user@host# **set groups node1 security ipsec proposal IPSEC_PROP_PSK encryption-algorithm aes-256-cbc**
user@host# **set groups node1 security ipsec proposal IPSEC_PROP_PSK lifetime-seconds 200**
user@host# **set groups node1 security ipsec policy IPSEC_POL_PSK perfect-forward-secrecy keys group20**
user@host# **set groups node1 security ipsec policy IPSEC_POL_PSK proposals IPSEC_PROP_PSK**
user@host# **set groups node1 security ipsec vpn S2S_VPN ha-link-encryption**
user@host# **set groups node1 security ipsec vpn S2S_VPN ike gateway S2S_GW**
user@host# **set groups node1 security ipsec vpn S2S_VPN ike ipsec-policy IPSEC_POL_PSK**
user@host# **set groups global interfaces fab0 fabric-options member-interfaces ge-0/0/3**
user@host# **set groups global interfaces fab1 fabric-options member-interfaces ge-5/0/3**
user@host# **commit**
user@host>**request system reboot**

6. To proceed further with device 2 configuration and commit, you need to ensure device1 and device 2 are not reachable to each other. One way to achieve this is to power off device 1 at this point.

7. Configure device 2 with standard cluster command for operating in cluster mode as node1

[edit]
user@host# **set groups node0 system host-name *node0-host-name***
user@host# **set groups node0 system backup-router *gateway-address***
user@host# **set groups node0 system backup-router destination *value***
user@host# **set groups node0 interfaces fxp0 unit 0 family inet address *node0-ip-address***
user@host# **set groups node1 system host-name *node1-host-name***
user@host# **set groups node1 system backup-router *gateway-address***
user@host# **set groups node1 system backup-router destination *value***
user@host# **set groups node1 interfaces fxp0 unit 0 family inet address *node1-ip-address***
user@host# **set apply-groups global**
user@host# **set apply-groups "$(node)"**
user@host# **delete apply-groups re0**

user@host# **set system ports console log-out-on-disconnect**
user@host# **set chassis cluster reth-count 5**
user@host# **set chassis cluster redundancy-group 0 node 0 priority 254**
user@host# **set chassis cluster redundancy-group 0 node 1 priority 1**
user@host# **commit**
user@host> **set chassis cluster cluster-id 1 node 1 reboot**

See
https://www.juniper.net/documentation/en_US/junos/topics/topic-map/security-chassis-cluster-verification.html

8. After the device 2 is up, configure HA link encryption as shown in sample configuration below on device 2. Device 2 needs to be configured with both node0 and node1 HA link encryption configuration. Commit on node1 (device 2), and finally reboot node1 (device 2).

[edit]
user@host# **set groups node0 security ike traceoptions file ikelog**
user@host# **set groups node0 security ike traceoptions file size 100m**
user@host# **set groups node0 security ike traceoptions flag all**
user@host# **set groups node0 security ike traceoptions level 15**
user@host# **set groups node0 security ike proposal IKE_PROP_PSK authentication-method pre-shared-keys**
user@host# **set groups node0 security ike proposal IKE_PROP_PSK dh-group group20**
user@host# **set groups node0 security ike proposal IKE_PROP_PSK authentication-algorithm sha-256**
user@host# **set groups node0 security ike proposal IKE_PROP_PSK encryption-algorithm aes-256-cbc**
user@host# **set groups node0 security ike policy IKE_POL_PSK proposals IKE_PROP_PSK**
user@host# **prompt groups node0 security ike policy IKE_POL_PSK pre-shared-key ascii-text**
New ascii-text (secret): juniper
Retype new ascii-text (secret): juniper

user@host# **set groups node0 security ike gateway S2S_GW ike-policy IKE_POL_PSK**
user@host# **set groups node0 security ike gateway S2S_GW version v2-only**
user@host# **set groups node0 security ipsec proposal IPSEC_PROP_PSK protocol esp**
user@host# **set groups node0 security ipsec proposal IPSEC_PROP_PSK authentication-algorithm hmac-sha1-96**
user@host# **set groups node0 security ipsec proposal IPSEC_PROP_PSK encryption-algorithm aes-256-cbc**
user@host# **set groups node0 security ipsec proposal IPSEC_PROP_PSK lifetime-seconds 200**
user@host# **set groups node0 security ipsec policy IPSEC_POL_PSK perfect-forward-secrecy keys group20**
user@host# **set groups node0 security ipsec policy IPSEC_POL_PSK proposals IPSEC_PROP_PSK**
user@host# **set groups node0 security ipsec vpn S2S_VPN ha-link-encryption**
user@host# **set groups node0 security ipsec vpn S2S_VPN ike gateway S2S_GW**
user@host# **set groups node0 security ipsec vpn S2S_VPN ike ipsec-policy IPSEC_POL_PSK**
user@host# **set groups node1 security ike traceoptions file ikelog**

```
user@host# set groups node1 security ike traceoptions file size 100m
user@host# set groups node1 security ike traceoptions flag all
user@host# set groups node1 security ike traceoptions level 15
user@host# set groups node1 security ike proposal IKE_PROP_PSK authentication-method
  pre-shared-keys
user@host# set groups node1 security ike proposal IKE_PROP_PSK dh-group group20
user@host# set groups node1 security ike proposal IKE_PROP_PSK authentication-algorithm
  sha-256
user@host# set groups node1 security ike proposal IKE_PROP_PSK encryption-algorithm aes-256-cbc
user@host# set groups node1 security ike policy IKE_POL_PSK proposals IKE_PROP_PSK
user@host# prompt groups node1 security ike policy IKE_POL_PSK pre-shared-key ascii-text
New ascii-text (secret): juniper
Retype new ascii-text (secret): juniper

user@host# set groups node1 security ike gateway S2S_GW ike-policy IKE_POL_PSK
user@host# set groups node1 security ike gateway S2S_GW version v2-only
user@host# set groups node1 security ipsec proposal IPSEC_PROP_PSK protocol esp
user@host# set groups node1 security ipsec proposal IPSEC_PROP_PSK authentication-algorithm
  hmac-sha1-96
user@host# set groups node1 security ipsec proposal IPSEC_PROP_PSK encryption-algorithm
  aes-256-cbc
user@host# set groups node1 security ipsec proposal IPSEC_PROP_PSK lifetime-seconds 200
user@host# set groups node1 security ipsec policy IPSEC_POL_PSK perfect-forward-secrecy keys
  group20
user@host# set groups node1 security ipsec policy IPSEC_POL_PSK proposals IPSEC_PROP_PSK
user@host# set groups node1 security ipsec vpn S2S_VPN ha-link-encryption
user@host# set groups node1 security ipsec vpn S2S_VPN ike gateway S2S_GW
user@host# set groups node1 security ipsec vpn S2S_VPN ike ipsec-policy IPSEC_POL_PSK
user@host# set groups global interfaces fab0 fabric-options member-interfaces ge-0/0/3
user@host# set groups global interfaces fab1 fabric-options member-interfaces ge-5/0/3
user@host# commit
user@host> request system reboot
```

9. Power ON node0 (device 1).

10. Both the nodes will be in cluster mode with HA link encryption enabled.

> **NOTE:** To enable HA link encryption on node1 in step 6, the other node needs to be in lost state for the commit to go through. So this timing needs to be taken care by you, else step 6 needs to be redone until enabling HA link encryption on node1 commit goes through.

CHAPTER 6

Junos-FIPS Configuration Restrictions

Understanding Configuration Limitations and Restrictions on Junos OS in FIPS Mode | 57

Unsupported Junos-FIPS Configuration Statements | 58

Unsupported Junos-FIPS Operational Commands | 59

Understanding Configuration Limitations and Restrictions on Junos OS in FIPS Mode

In FIPS mode, an SRX Series device operates as a nonmodifiable operational environment in which only files shipped as part of Junos OS can be executed.

In contrast to non-FIPS mode, Junos OS in FIPS mode:

- Conforms to FIPS 140-2.
- Requires special installation procedures.
- Mandates the use of internal, manual IPsec tunnels with specific requirements.
- Limits services used for remote access.
- Allows only the use of approved ciphers.
- Requires user logout on disconnect at the console.
- Sets strict requirements for passwords.
- Requires special system logging considerations.
- Disables the following Junos OS protocols and services so that you cannot configure them. Attempts to configure these services or to load configurations with these services configured result in a configuration syntax error.
  - finger
  - FTP
  - rlogin
  - rsh
  - Telnet
  - Trivial File Transfer Protocol (TFTP)
  - Transport Layer Security (TLS) protocol
  - xnm-clear-text

If you try to load a configuration that includes statements not supported by Junos OS in FIPS mode, you see a warning message. For example, suppose you attempt to configure Telnet for remote access:

```
[edit]
crypto-officer:fips# set system services telnet
```

You receive the following warning and cannot add the **system services telnet** statement to the loaded configuration:

```
[edit]
'telnet'
warning: not allowed in JUNOS-FIPS; ignored
```

RELATED DOCUMENTATION

Understanding Junos OS in FIPS Mode of Operation

Unsupported Junos-FIPS Configuration Statements

The following configuration statements are not supported on Junos-FIPS:

Statement	Description
set system services { ftp | finger | telnet | web-management | xnm-clear-text | tftp}	Junos-FIPS does not allow an unencrypted or weakly encrypted or a connection that relies on a vulnerable key establishment protocol.
set system services ssh protocol-version	Junos-FIPS allows the SSHv2 setting only.
set system login password format { des | md5 }	You must encrypt administrator passwords using strong algorithms, such as Secure Hash Algorithm (sha-256 and sha-512).
set system ike policy *policy name* proposal-set	Junos-FIPS does not support preconfigured proposal sets. You must configure an IKE proposal explicitly.
set system ike proposal *proposal name* authentication-algorithm md5 **set system ipsec proposal *proposal name* authentication-algorithm hmac-md5-96**	Junos-FIPS does not support Message Digest 5 (MD5). However it does support (sha-256 and sha-384).

Statement	Description
set system ike proposal *proposal name* encryption-algorithm des-cbc **set system ipsec proposal *proposal name* encryption-algorithm des-cbc**	Junos-FIPS does not support Data Encryption Standard (DES). However it does support Advanced Encryption Standard (AES) or 3DES.
set system ike proposal *proposal name* protocol ah	Authentication Header (AH) protocol provides authentication but not encryption. Enhanced Security Protocol (ESP) is required.
set system ike proposal *proposal name* dh-group {group1 | group2}	Junos-FIPS does not support Diffie-Hellman (DH) groups 1 and 2. However, DH-group 14 and higher are supported on Junos-FIPS.

RELATED DOCUMENTATION

Unsupported Junos-FIPS Operational Commands | 59

Unsupported Junos-FIPS Operational Commands

The following operating commands are not supported on Junos-FIPS:

Command	Description
request system software reboot *at time in minutes* usb	You must load the firmware image directly from the internal flash memory. Junos-FIPS does not support loading from an external source.
set wlan	Junos-FIPS does not support wireless configuration.
request system software rollback	You must upload a new version of the firmware explicitly. Junos-FIPS does not support the rollback to a previously saved version.
request security pki ca-certificate enroll	You must enroll the certificate authority enrollment manually. Simple Certificate Enrollment Protocol (SCEP) is disabled.

RELATED DOCUMENTATION