Juniper Networks PTX1000 Packet Transport Router

Firmware: Junos OS 20.3X75

Non-Proprietary FIPS 140-2 Cryptographic Module Security Policy

Document Version: 1.0

Date: April 07, 2023

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408.745.2000
1.888 JUNIPER
www.juniper.net

Contents

List of Tables

List of Figures

1 Introduction

This is a non-proprietary Cryptographic Module Security Policy for the Juniper Networks PTX1000 Packet Transport Router. The Juniper Networks PTX1000 Packet Transport Router is a fixed-configuration router that supports 10-Gbps, 40-Gbps, and 100-Gbps port speeds in a compact 2U form factor, enabling service providers to organically distribute peering points throughout the network. The FIPS validated version of firmware is Junos OS 20.3X75.

The cryptographic boundary for the module is defined as follows for the validation:

- the outer edge of the chassis including the Routing Engine (RE):
  - 1 built-in RE (RE-PTX1000).
- excluding the power distribution module on the rear of the device.

The cryptographic module provides for an encrypted connection, using SSH, between the management station and the module. All other data input to or output from the module are considered plaintext for this FIPS 140-2 validation.

The cryptographic module is defined as a multiple-chip standalone module that executes Junos OS 20.3X75 firmware on the Juniper Networks PTX1000 Packet Transport Router as listed in Table 1 below.

Table 1 – Cryptographic Module Hardware Configuration

Chassis PN	Power PN	RE PN
PTX1000	JPSU-1600W-AC-AFO JPSU-1600W-DC-AFO	Built-in Routing Engine (RE-PTX1000)

The module is designed to meet FIPS 140-2 Level 1 overall:

Table 2 – Security Level of Security Requirements

Area	Description	Level
1	Module Specification	1
2	Ports and Interfaces	1
3	Roles, Services, and Authentication	3
4	Finite State Model	1
5	Physical Security	1
6	Operational Environment	N/A
7	Key Management	1
8	EMI/EMC	1
9	Self-test	1
10	Design Assurance	3
11	Mitigation of Other Attacks	N/A
	Overall	1

The module has a limited operational environment as per the FIPS 140-2 definitions. It includes a firmware load service to support necessary updates. New firmware versions within the scope of this validation must be validated through the FIPS 140-2 CMVP. Any other firmware loaded into the module are out of the scope of this validation and require a separate FIPS 140-2 validation.

The module does not implement any mitigation of other attacks as defined by FIPS 140-2.

1.1 Hardware and Physical Cryptographic Boundary

The cryptographic module's operational environment is a limited operational environment.

The image below depicts the physical boundary of the module which includes the Routing Engine. The module excludes the power supplies from the requirements of FIPS 140-2. The power supplies do not contain any security relevant components and cannot affect the security of the module.

Figure 1 – Cryptographic Boundary (PTX1000)

Table 3 – Ports and Interfaces

Port	Device (# of ports)	Description	Logical Interface Type
Ethernet	74 ports: (Management ports (2); 40-Gigabit Ethernet QSFP+ network ports (72))	LAN Communications/Remote management	Control in, Data in, Data out, Status out
Ethernet	SFP PTP Ethernet (1000BASE-T) port (1)	Reserved for future use	N/A
Serial	1	Console serial port	Control in, Data in, Data out Status out
USB	1	USB port - load Junos image	Control in, Data in
Power	4	Power connector	Power
LEDs	80	Status indicator lighting	Status out
Reset Button	1	Reset	Control in

10 Hz pulses-per-second (PPS) SubMiniature B (SMB) connector	1	Reserved for future use	N/A
10 MHz SMB timing connector (10MHz).	1	Reserved for future use	N/A

1.2 Modes of Operation

The module supports one FIPS Approved mode of operation and a non-Approved mode of operation. The module must always be zeroized when switching between the FIPS Approved mode of operation and the non-Approved mode of operation and vice versa.

1.2.1 FIPS Approved Mode

The hardware version contained in Table 1, with Junos OS 20.3X75 installed, contains one FIPS-Approved mode of operation and a non-Approved mode of operation. The Junos OS 20.3X75 firmware image must be installed on the device. The module is configured during initialization to operate in the approved mode or the non-approved mode.

The Crypto-Officer places the module in the Approved mode of operation by following the instructions in cryptographic officer guidance (Section 6.1).

The Crypto-Officer can verify that the cryptographic module is in the Approved mode by observing the console prompt and running the “show version local” command. When operating in FIPS mode, the prompt will read “<user>@<device name>:fips>” (e.g., crypto-officer@PTX1000:fips>). The “show version local” command will allow the Crypto-Officer to verify that the validated firmware version is running on the module. The Crypto-Officer can also use the “show system fips” command to determine if the module is operating in FIPS mode.

1.2.2 Non-Approved Mode

The cryptographic module supports a non-Approved mode of operation. When operated in the non-Approved mode of operation, the module supports the algorithms identified in Section 2.2 as well as the algorithms supported in the Approved mode of operation.

The Crypto-Officer can place the module into a non-approved mode of operation by following the instructions in the cryptographic officer guidance (Section 6.1).

1.3 Zeroization

The cryptographic module provides a non-Approved mode of operation in which non-Approved cryptographic algorithms are supported. When transitioning between the non-Approved mode of operation and the FIPS-Approved mode of operation, or vice-versa, the cryptographic officer shall zeroize all keys and CSPs.

Zeroization completely erases all configuration information on the router. The Crypto Officer initiates the zeroization process by entering the "*request vmhost zeroize"* operational command from the CLI.

Use of the zeroize command is restricted to the Cryptographic Officer. The cryptographic officer shall perform zeroization in the following situations:

1. Before FIPS Operation: To prepare the device for operation as a FIPS cryptographic module by erasing all CSPs and other user-created data on a device before its operation as a FIPS cryptographic module.
2. Before non-FIPS Operation: To conduct erasure of all CSPs and other user-created data on a device in preparation for repurposing the device for non-FIPS operation.

CAUTION: Perform system zeroization with care. After the zeroization process is complete, no data is left on the Routing Engine. The device is returned to the factory default state, without any configured users or configuration files.

To zeroize the device:

1. From the CLI, enter

    crypto-officer@device> **request vmhost zeroize**
    warning: System will be rebooted and may not boot without configuration
    Erase all data, including configuration and log files? [yes, no] (no)

2. To initiate the zeroization process, type yes at the prompt:

    Erase all data, including configuration and log files? [yes, no] (no)
yes

3. When the system finishes rebooting the system will be in a factory default state.

Note: The Cryptographic Officer must retain control of the module while zeroization is in process.

2 Cryptographic Functionality

2.1 Allowed Algorithms and Protocols

The module implements the FIPS Approved cryptographic functions listed in Tables 4, 5, 6, and 7 below. Table 8 summarizes the high-level protocol algorithm support. There are some algorithm modes that were tested but not implemented by the module. Only the algorithms, modes, and key sizes that are implemented by the module are shown in this/these table(s).

Table 4 – Kernel Approved Cryptographic Functions

CAVP Cert.	Algorithm	Standard	Mode	Description	Functions
A2356	DRBG	SP800-90A	HMAC	SHA-256	Random Bit Generation
A2356	HMAC	PUB 198	SHA-1[1]	Key size: 160 bits, λ = 96	Message Authentication
			SHA-256	Key size: 256 bits, λ = 128, 256	Message Authentication, DRBG Primitive
A2356	SHS	PUB 180-4	SHA-1 SHA-256 SHA-384 SHA-512		Message Digest Generation

Table 5 – LibMD Approved Cryptographic Functions

CAVP Cert.	Algorithm	Standard	Mode	Description	Functions
A2357	HMAC	PUB 198	SHA-1	Key size: 160 bits, λ = 96	Message Authentication
			SHA-256	Key size: 256 bits, λ = 128, 256	
A2357	SHS	PUB 180-4	SHA-1 SHA-256 SHA-512		Message Digest Generation

Table 6 – OpenSSL Approved Cryptographic Functions

CAVP Cert.	Algorithm	Standard	Mode	Description	Functions
A2359	AES[2]	PUB 197-38A	CBC, ECB	Key Sizes: 128, 192, 256	Encrypt, Decrypt
			CTR		Encrypt
N/A[3]	CKG	SSH-PUB 133	Section 6.1 Section 6.2		Asymmetric key generation using

[1] The usage of SHA-1 is restricted to message authentication, digest generation and as a primitive in a KDF for all cryptographic libraries/implementations in the module.

[2] The AES-ECB mode was used for testing the AES-CTR mode.

[3] Vendor Affirmed

					unmodified DRBG output
N/A	KAS	SP 800-56Arev3	KAS-SSC-ECC per SP 800-56Arev3 (Cert. #A2359) with Key Derivation per SSH KDF CVL (Cert. #A2359)		Key Agreement Scheme IG D.8 Scenario X1 option 2
A2359	KAS-SSC	SP 800-56Arev3	ECC	P-256 (SHA 256) P-384 (SHA 384) P-521 (SHA 512)	Key Agreement Scheme – Shared Secret Computation
A2359	CVL	SP 800-135	SSH	SHA 1, 256, 384, 512	Key Derivation
A2359	DRBG	SP 800-90A	HMAC	SHA-256	Random Number Generation
A2359	ECDSA	PUB 186-4		P-256 (SHA 256) P-384 (SHA 384) P-521 (SHA 512)	SigGen, KeyGen, SigVer, PKV
A2359	HMAC	PUB 198	SHA-1	Key size: 160 bits, λ = 160	Message Authentication
			SHA-224	Key size: 224 bits, λ = 192	
			SHA-512	Key size: 512 bits, λ = 512	
			SHA-256	Key size: 256, bits, λ = 256	Message Authentication, DRBG Primitive
N/A	KTS		AES Cert. #A2359 and HMAC Cert. #A2359		key establishment methodology provides between 128 and 256 bits of encryption strength
A2359	RSA	PUB 186-4		n=2048 (SHA 256, 512) n=3072 (SHA 256, 512) n=4096 (SHA 256, 512)	KeyGen, SigGen, SigVer
A2359	SHS	PUB 180-4	SHA-1 SHA-256 SHA-384 SHA-512		Message Digest Generation, KDF Primitive
			SHA-224		Message Digest Generation

Table 7 – Entropy

Algorithm	Caveat	Use

SP 800-90B ENT (NP)	The module generates a minimum of 256 bits of entropy for key generation.	Seeding the DRBG

Table 8 – Protocols Allowed in FIPS Mode

Protocol	Key Exchange	Auth	Cipher	Integrity
SSHv2	EC Diffie-Hellman P-256, P-384, P-521	RSA 2048, 4096 ECDSA P-256	AES CBC 128/192/256 AES CTR 128/192/256	HMAC-SHA-1 HMAC-SHA-256 HMAC-SHA-512

No part of these protocols, other than the KDF, have been tested by the CAVP and CMVP. The SSH algorithm allows independent selection of key exchange, authentication, cipher, and integrity. In Table 8 above, each column of options is independent and may be used in any viable combination.

2.2 Disallowed Algorithms and Protocols

These algorithms and protocols are non-Approved algorithms and protocols that are disabled when the module is operated in the Approved mode of operation. The algorithms are available as part of the SSH connect service when the module is operated in the non-Approved mode.

Algorithms

- RSA with key size less than 2048
- ECDSA with ed25519 curve
- ECDH with ed25519 curve
- ARCFOUR
- Blowfish
- CAST
- DSA (SigGen, SigVer; non-compliant)
- HMAC-MD5
- HMAC-RIPEMD160
- UMAC
- Chacha20
- Poly
- Diffie-Hellman

Protocols

- Finger
- ftp
- rlogin
- telnet
- tftp
- xnm-clear-text

2.3 Critical Security Parameters

All CSPs and public keys used by the module are described in this Section.

Table 9 – Critical Security Parameters (CSPs)

Name	Description and usage
DRBG_Seed	Seed material used to seed or reseed the DRBG
DRBG_State	Values V and Key which comprise the HMAC_DRBG state
Entropy Input	256 bits entropy (min) input used to instantiate the DRBG
ECDH Shared Secret	The Diffie-Hellman shared secret used in EC Diffie-Hellman (ECDH) exchange. Created per the EC Diffie-Hellman protocol. Provides between 128-256 bits of security.
SSH PHK	SSH Private host key. 1st time SSH is configured, the keys are generated. ECDSA P-256. RSA 2048, 4096
SSH ECDH	Ephemeral EC Diffie-Hellman private key used in SSH. ECDH P-256, P-384, or P-521
SSH-SEKs	SSH Session Keys: SSH Session Encryption Key: AES (128,192,256); SSH Session Integrity Key: HMAC (SHA-1, SHA-256, SHA2-512).
HMAC Key	The LibMD HMAC keys: message digest for hashing password and critical function test.
User Password	Passwords used to authenticate Users to the module.
CO Password	Passwords used to authenticate COs to the module.

Table 10 – Public Keys

Name	Description and usage
SSH-PUB	SSH Public Host Key used to identify the host. ECDSA P-256. RSA 2048, 4096.
SSH-ECDH-PUB	Ephemeral EC Diffie-Hellman public key used in SSH key establishment. ECDH P-256, P-384, P-521
Auth-User Pub	User Authentication Public Keys. Used to authenticate users to the module. ECDSA P-256, P-384, P-521 or RSA 2048, 4096
Auth-CO Pub	CO Authentication Public Keys. Used to authenticate CO to the module. ECDSA P-256, P-384, P-521 or RSA 2048, 4096
Root CA	ECDSA P-256 X.509 Certificate; Used to verify the validity of the Juniper Package CA at software load and also at runtime for integrity.
Package CA	ECDSA P-256 X.509 Certificate; Used to verify the validity the Juniper Image at software load and also at runtime for integrity.

3 Roles, Authentication and Services

3.1 Roles and Authentication of Operators to Roles

The module supports two roles: Cryptographic Officer (CO) and User. The module supports concurrent operators but does not support a maintenance role and/or bypass capability. The module enforces the separation of roles using identity-based operator authentication.

The Cryptographic Officer role configures and monitors the module via a console or SSH connection. As root or super-user, the Cryptographic Officer has permission to view and edit secrets within the module.

The User role monitors the router via the console or SSH. The User role cannot change the configuration.

3.2 Authentication Methods

The module implements two forms of Identity-Based authentication, Username and password over the Console and SSH as well as Username and ECDSA or RSA public key over SSH.

Password authentication: The module enforces 10-character passwords (at minimum) chosen from the 96 human readable ASCII characters. The maximum password length is 20-characters. Thus, the probability of a successful random attempt is $1/96^{10}$, which is less than 1/1 million.

The module enforces a timed access mechanism as follows: For the first two failed attempts (assuming 0 time to process), no timed access is enforced. Upon the third attempt, the module enforces a 5-second delay. Each failed attempt thereafter results in an additional 5-second delay above the previous (e.g. 4^{th} failed attempt = 10-second delay, 5^{th} failed attempt = 15-second delay, 6^{th} failed attempt = 20-second delay, 7^{th} failed attempt = 25-second delay).

This leads to a maximum of 7 possible attempts in a one-minute period for each getty. 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/60 mins); this would be rounded down to 9 per minute, because there is no such thing as 0.6 attempts. The probability of a success with multiple consecutive attempts in a one-minute period is $9/(96^{10})$, which is less than 1/100,000.

ECDSA signature verification: SSH public-key authentication. The module supports ECDSA (P-256, P-384, and P-521), which has a minimum equivalent computational resistance to attack of either 2^{128} , 2^{192} or 2^{256} depending on the curve. Thus, the probability of a successful random attempt is $1/ (2^{128})$, which is less than 1/1,000,000. Configurable SSH connection establishment rate limits the number of connection attempts, and thus failed authentication attempts in a one-minute period to a maximum of 15,000 attempts. The probability of a success with multiple consecutive attempts in a one-minute period is $15,000/(2^{128})$, which is less than 1/100,000.

RSA signature verification: SSH public-key authentication. The module supports RSA (2048, 4096), which has a minimum equivalent computational resistance to attack of 2^{112} (2048). Thus, the probability of a successful random attempt is $1/ (2^{112})$, which is less than 1/1,000,000. Configurable SSH connection establishment rate limits the number of connection attempts, and thus failed authentication attempts in a one-minute period to a maximum of 15,000 attempts. The probability of a success with multiple

consecutive attempts in a one-minute period is 15,000/ (2^{112}), which is less than 1/100,000.

3.3 Approved and Allowed Services

All services implemented by the module are listed in the tables below. Table 13 lists the access to CSPs by each service.

Table 11 – Authenticated Services

Service	Description	CO	User
Configure security	Security relevant configuration	x	
Configure	Non-security relevant configuration	x	
Status	Show status	x	x
Zeroize	Destroy all CSPs	x	
SSH connect	Initiate SSH connection for SSH monitoring and control (CLI)	x	x
Console access	Console monitoring and control (CLI)	x	x
Remote reset	Software initiated reset, performs self-tests on demand.	x	
Load Image	Verification and loading of a validated firmware image	x	

Table 12 – Unauthenticated Services

Service	Description
Local reset	Hardware reset or power cycle
Traffic	Traffic requiring no cryptographic services (e.g., OSPF, BGP)
LED Status	Basic

Table 13 – CSP Access Rights within Services

Service	CSPs									
	DRBG_Seed	DRBG_State	Entropy Input	ECDH Shared Secret	SSH PHK	SSH ECDH	SSH-SEK	HMAC Key	CO-PW	User-PW
Configure security	--	E	--	GWR	G W R	--	--	G	W	W
Configure	--	--	--	--	--	--	--	--	--	--
Status	--	--	--	--	--	--	--	--	--	--
Zeroize	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
SSH connect	--	E	--	E	E	GE	GE	--	E	E
Console access	--	--	--	--	--	--	--	--	E	E
Remote reset	GEZ	GZ	GZ	Z	--	Z	Z	--	--	--
Load Image	--	--	--	--	--	--	--	--	--	--
Local reset	GEZ	GZ	GZ	Z	--	Z	Z	--	--	--
Traffic	--	--	--	--	--	--	--	--	--	--
LED Status	--	--	--	--	--	--	--	--	--	--

G = Generate: The module generates the CSP
R = Read: The CSP is read from the module (e.g. the CSP is output)
E = Execute: The module executes using the CSP
W = Write: The CSP is updated or written to the module (persistent storage)
Z = Zeroize: The module zeroizes the CSP.

3.4 Non-Approved Services

The following services are available in the non-Approved mode of operation. The security functions provided by the non-Approved services are identical to the Approved counterparts with the exception of SSH Connect (non-compliant). SSH Connect (non-compliant) supports the security functions identified in Section 2.2 and the SSHv2 row of Table 8.

Table 14 – Non-Approved Authenticated Services

Service	Description	CO	User
Configure security (non-compliant)	Security relevant configuration	x	
Configure (non-compliant)	Non-security relevant configuration	x	
Status (non-compliant)	Show status	x	x

Service	Description	CO	User
Zeroize (non-compliant)	Destroy all CSPs	x	
SSH connect (non-compliant)	Initiate SSH connection for SSH monitoring and control (CLI)	x	x
Console access (non-compliant)	Console monitoring and control (CLI)	x	x
Remote reset (non-compliant)	Software initiated reset, performs self-tests on demand	x	
Load Image (non-compliant)	Verification and loading of a validated firmware image into the router.	x	

Table 15 – Non-Approved Unauthenticated Services

Service	Description
Local reset	Hardware reset or power cycle
Traffic	Traffic requiring no cryptographic services (e.g. OSPF, BGP)
LED Status	Basic

4 Self-tests

Each time the module is powered up it tests that the cryptographic algorithms still operate correctly, and that sensitive data have not been damaged. Power-up self–tests are available on demand by power cycling the module (Remote reset service).

On power up or reset, the module performs the self-tests described below. All KATs must be completed successfully prior to any other use of cryptography by the module in the FIPS Approved Mode of operation. If any one of the Routing Engine KATs fails, the module enters the Error state.

The module performs the following power-up self-tests:

Routing Engine (RE)

- **Firmware Integrity check**: using ECDSA P-256 with SHA-256
- **Kernel KATs**
    - NIST 800-90A HMAC DRBG KAT
        - Health-tests initialize, re-seed, and generate
    - HMAC-SHA-1 KAT
    - HMAC-SHA2-256 KAT
    - SHA-2-384 KAT
    - SHA-2-512 KAT
- **OpenSSL KATs**
    - NIST 800-90A HMAC DRBG KAT
        - Health-tests initialize, re-seed, and generate
    - ECDSA P-256 Sign/Verify PCT
    - RSA 2048 w/ SHA-256 Sign KAT
    - RSA 2048 w/ SHA-256 Verify KAT
    - HMAC-SHA1 KAT
    - HMAC-SHA2-224 KAT
    - HMAC-SHA2-256 KAT
    - HMAC-SHA2-384 KAT
    - HMAC-SHA2-512 KAT
    - AES-CBC KAT
    - KDF-SSH-SHA256 KAT
    - KAS-ECC KAT
- **LibMD KATs**
    - HMAC-SHA1 KAT
    - HMAC-SHA2-256 KAT
    - SHA-2-512 KAT

- Critical Function Test
    - The cryptographic module performs a verification of a limited operational environment, and verification of optional non-critical packages.

- NIST SP 800-90B ENT (NP) Start Up Health Tests
    - Adaptive Proportion Test and the Repetition Count Test over 1024 samples.

The module also performs the following conditional self-tests:

- Adaptive Proportion Test (APT) and Repetition Count Test (RCT) on the SP 800-90B compliant entropy source (ENT (NP)).
- Pairwise consistency test when generating ECDSA, and RSA key pairs.
- Firmware Load Test (ECDSA signature verification).

5 Physical Security Policy

The module's physical embodiment is that of a multi-chip standalone device that meets Level 1 Physical Security requirements. The module is completely enclosed in a rectangular nickel or clear zinc coated, cold rolled steel, plated steel and brushed aluminum 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.

6 Security Rules and Guidance

The module design corresponds to the security rules below. The term *shall* in this context specifically refers to a requirement for correct usage of the module in the Approved mode; all other statements indicate a security rule implemented by the module.

1. The module clears previous authentications on power cycle.
2. When the module has not been placed in a valid role, the operator does not have access to any cryptographic services.
3. Power up self-tests do not require any operator action.
4. Data output is inhibited during key generation, self-tests, zeroization, and error states.
5. Status information does not contain CSPs or sensitive data that if misused could lead to a compromise of the module.
6. There are no restrictions on which keys or CSPs are zeroized by the zeroization service.
7. The module does not support a maintenance interface or role.
8. The module does not support manual key entry.
9. The module does not output intermediate key values.
10. The module requires two independent internal actions to be performed prior to outputting plaintext CSPs.
11. The cryptographic officer shall verify that the firmware image to be loaded on the module is a FIPS validated image. If any non-validated firmware image is loaded the module will no longer be a FIPS validated module.
12. The cryptographic officer shall retain control of the module while zeroization is in process.
13. Virtual Chassis is not supported in FIPS mode and shall not be configured on the module.
14. RSA key generated shall only be 2048 bits or greater.

6.1 Cryptographic-Officer Guidance

The cryptographic officer must check to verify the firmware image on the router is the FIPS 140-2 validated image. If the image is the FIPS 140-2 validated image, then proceed to Section 6.1.2.

6.1.1 Installing the FIPS-Approved Firmware Image

Download the validated firmware image from https://www.juniper.net/support/downloads/junos.html. Log in to the Juniper Networks authentication system using the username (generally your e-mail address) and password supplied by Juniper Networks representatives. Select the validated firmware image. Download the firmware image to a local host or to an internal software distribution site.

Connect to the console port on the device from your management device and log in to the Junos OS CLI. Copy the firmware package to the device to the /var/tmp/ directory. Install the new package on the device:

user@device> request vmhost software add /var/tmp/*package*.tgz.

NOTE: If you need to terminate 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, junos-vmhost-install-ptx-x86-64-20.3X75.tgz. This is your last chance to stop the installation.

Reboot the device to load the installation and start the new firmware image:

```
user@device> request vmhost reboot
```

After the reboot has completed, log in and use the "show version local" command to verify that the new version of the firmware is successfully installed. Also verify that jpfe-fips and fips-mode packages needed for enabling the executing the self-tests and enabling the FIPS-Approved mode of operation respectively are available. If unavailable, add the packages as follows:

```
user@device> request system software add optional://jpfe-fips.tgz
Verified jpfe-fips signed by Package Production ECP256_2021 method ECDSA256+SHA256

user@device> request system software add optional://fips-mode.tgz
Verified fips-mode signed by Package Production ECP256_2021 method ECDSA256+SHA256
```

6.1.2 Enabling FIPS-Approved Mode of Operation

The cryptographic officer is responsible for initializing the module in a FIPS-Approved mode of operation. The FIPS-Approved mode of operation is not automatically enabled. The cryptographic officer shall place the module in the FIPS-Approved mode by first zeroizing the device to delete all keys and CSPs. The instructions for zeroizing the module are in Section 1.3 of this document. Next, the cryptographic officer shall follow the steps found in the *Junos OS FIPS Evaluated Configuration Guide for PTX1000, Release 20.3X75* document Chapter 2 to place the module into a FIPS-Approved mode of operation. The steps from the aforementioned document are repeated below:

The FIPS Approved Mode of operation is not automatically enabled once the firmware image is installed on the platform. These steps are for putting the module into the FIPS Approved Mode.

To enable FIPS mode in Junos OS on the device:

1. Zeroize the device as explained in Section 1.3. Once device comes up in amnesiac mode post zeroize, connect to device using console port with username "root", set the root-authentication password and configure crypto-officer credentials.

2. Login to the device with crypto-officer credentials and enter configuration mode:

```
crypto-officer@device> edit
Entering configuration mode
[edit]
crypto-officer@device#
```

3. Enable FIPS mode on the device by setting the FIPS level to 1, and verify the level:

[edit]
crypto-officer@device # **set system fips chassis level 1**

[edit]
crypto-officer@device # **show system fips**
level 1;

4. Commit the configuration

   [edit]
   crypto-officer@device# **commit**
   configuration check succeeds
   [edit]
   'system'
   reboot is required to transition to FIPS level 1
   commit complete

5. Reboot the device:

   [edit]
   crypto-officer@device# **run request vmhost reboot**
   Reboot the system ? [yes,no] (no) **yes**

   During the reboot, the device runs Known Answer Tests (KATS). It returns a login prompt.

6. After the reboot has completed, log in and use the “show version local” command to verify the firmware version is the validated version.

   crypto-officer@device:fips> show version local

6.1.3 Placing the Module in a Non-Approved Mode of Operation

As cryptographic officer, the operator needs to disable the FIPS-Approved mode of operation on the device to return it to a non-Approved mode of operation. To disable FIPS-Approved mode on the device, the router must be zeroized. Follow the steps found in Section 1.3 to zeroize the router.

6.2 User Guidance

The user should verify that the module is operating in the desired mode of operation (FIPS-Approved mode or non-Approved mode) by observing the command prompt when logged into the device. If the string “:fips” is present, then the router is operating in a FIPS-Approved mode. Otherwise, it is operating in a non-Approved mode.

All FIPS users, including the Crypto 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.

7 References and Definitions

The following standards are referred to in this Security Policy.

Table 16 – References

Abbreviation	Full Specification Name
[FIPS140-2]	*Security Requirements for Cryptographic Modules*, May 25, 2001
[SP800-131A]	*Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths*, January 2011
[IG]	*Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program*

Table17 – Acronyms and Definitions

Acronym	Definition
AES	Advanced Encryption Standard
DH	Diffie-Hellman
DSA	Digital Signature Algorithm
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
EMI	Electromagnetic Interference
EMC	Electromagnetic Compatibility
ESD	Electrostatic Discharge
FIPS	Federal Information Processing Standard
HMAC	Keyed-Hash Message Authentication Code
MD5	Message Digest 5
RE	Routing Engine
RSA	Public-key encryption technology developed by RSA Data Security, Inc.
SCB	Router Control Board
SHA	Secure Hash Algorithms
SSH	Secure Shell

Table 18 – Datasheet

Model	Title	URL
PTX1000	PTX1000, PTX10001, PTX10002, and PTX10003 Fixed Configuration Routers	https://www.juniper.net/content/dam/www/assets/datasheets/us/en/routers/ptx1000-ptx10001-ptx10002-ptx10003-fixed-configuration-packet-transport-routers-datasheet.pdf