Juniper Networks, Inc

Juniper Networks SRX Series Services Gateways

FIPS 140-3 Non-Proprietary Security Policy

Version: Junos OS 22.2R3-S1

Prepared for:

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

Prepared by:

www.teronlabs.com

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This is a non-proprietary Cryptographic Module Security Policy for the Juniper Networks SRX Series Services Gateways consisting of the SRX1500, SRX4100, SRX4200, SRX4600, SRX5400, SRX5600 and SRX5800 models and running Junos OS 22.2R3-S1, hereafter referred to as the cryptographic module.

1.2 Security Levels

The cryptographic module meets requirements applicable to Level 2 of FIPS 140-3. The table below shows the security levels claimed for each section of the security requirements.

Section	Title	Security Level
1	General	2
2	Cryptographic module specification	2
3	Cryptographic module interfaces	2
4	Roles, services, and authentication	2
5	Software/Firmware security	2
6	Operational environment	N/A
7	Physical security	2
8	Non-invasive security	N/A
9	Sensitive security parameter management	2
10	Self-tests	2
11	Life-cycle assurance	2
12	Mitigation of other attacks	N/A
	Overall Level	2

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The SRX Series Services Gateways are a series of secure routers that provide essential capabilities to connect, secure, and manage work force locations sized from handfuls to hundreds of users. By consolidating fast, highly available switching, routing, security, and applications capabilities in a single device, enterprises can economically deliver new services, safe connectivity, and a satisfying end user experience.

Module Type: Hardware

Module Embodiment: MultiChipStand

Module Characteristics:

Cryptographic Boundary:

This Security Policy covers the following models:

- SRX1500,
- SRX4100,
- SRX4200,
- SRX4600,
- SRX5400,
- SRX5600, and
- SRX5800.

All models run Juniper's JUNOS firmware. The JUNOS firmware is FIPS-compliant when configured in the Approved mode called JUNOS-FIPS-MODE, version 22.2R3-S1.

The firmware images used in the different models are as follows:

- SRX1500: junos-srxentedge-x86-64-22.2R3-S1.9.tgz;
- SRX4100 and SRX4200: junos-srxmr-x86-64-22.2R3-S1.9.tgz;
- SRX4600: junos-srxhe-x86-64-22.2R3-S1.9.tgz; and
- SRX5400, SRX5600 and SRX5800: junos-vmhost-install-srx-x86-64-22.2R3-S1.9.tgz.

The physical form of the module is depicted in Figure 1 to Figure 14 below. The cryptographic boundary encompasses the entire Tested Operational Environment Physical Perimeter (TOEPP), which is defined as the outer edge of the chassis. The module does not rely on external devices for input and output of security sensitive parameters (SSPs).

Figure 1 – SRX1500 (front)

Figure 2 – SRX1500 (rear)

Figure 3 – SRX4100 (front)

Figure 4 – SRX4100 (rear)

Figure 5 – SRX4200 (front)

Figure 6 – SRX4200 (rear)

Figure 7 – SRX4600 (front)

Figure 8 – SRX4600 (rear)

Figure 9 – SRX5400 (front)

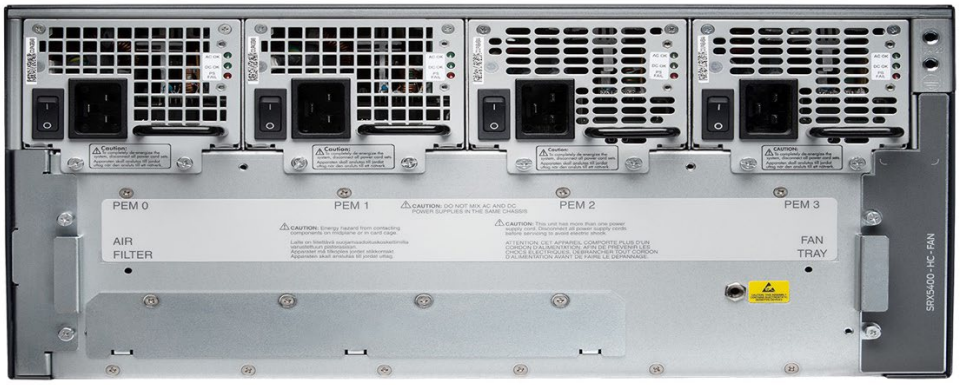

Figure 10 – SRX5400 (rear)

Figure 11 – SRX5600 (front)

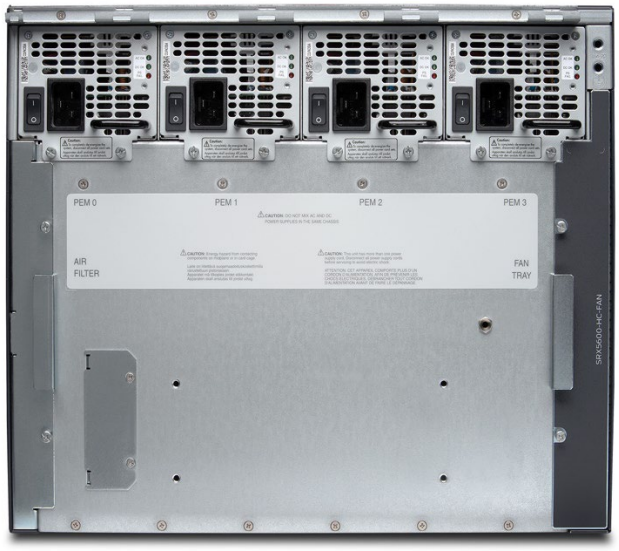

Figure 12 – SRX5600 (rear)

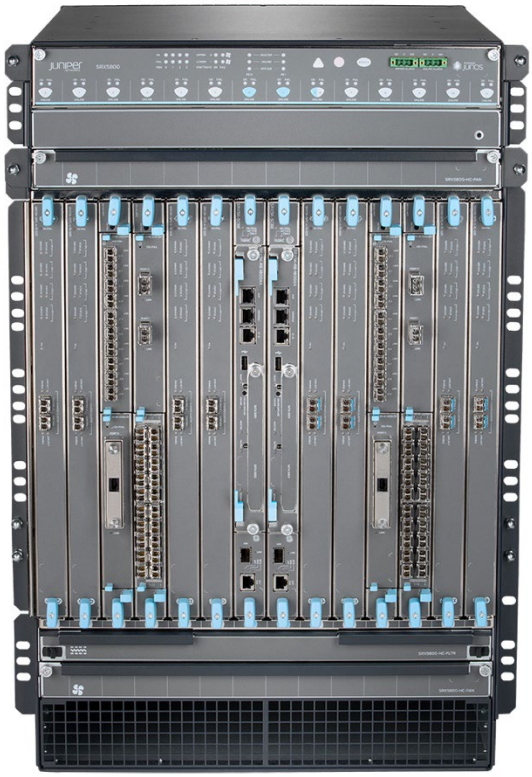

Figure 13 – SRX5800 (front)

Figure 14 – SRX5800 (rear)

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
SRX1500	SRX1500	JUNOS 22.2R3-S1.9	Intel Xeon E3-1200 v2	12x1GbE ports; 4x1GbE SFP ports; 4x10GbE SFP ports; 2 PIM slots (not used in validation)
SRX4100	SRX4100	JUNOS 22.2R3-S1.9	Intel Xeon E5-2640 v4	8 x 1GbE/10GbE ports
SRX4200	SRX4200	JUNOS 22.2R3-S1.9	Intel Xeon E5-2640 v4	8 x 1GbE/10GbE ports
SRX4600	SRX4600	JUNOS 22.2R3-S1.9	Intel Xeon E5-2658 v4	8 x 1GbE/10Gb Ethernet SFP ports, 4 x 40/100Gb Ethernet QSFP21 ports
SRX5400	SRX5400	JUNOS 22.2R3-S1.9	Intel Xeon C5518, Intel Xeon E5-2658 v4, Intel Xeon CPU E5-2608L v3	Routing Engine: SRX5K-RE3-128G Switch Control Board: SRX5K-SCB3 Service Processing Card: SRX5K-SPC-4-15-320, SRX5K-SPC3

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
				Module Interface Card: SRX5K-IOC4-10G, SRX5K-MPC3-40G10G
SRX5600	SRX5600	JUNOS 22.2R3-S1.9	Intel Xeon C5518, Intel Xeon E5-2658 v4, Intel Xeon CPU E5-2608L v3	Routing Engine: SRX5K-RE3-128G Switch Control Board: SRX5K-SCB3, SRX5K-SCB4 Service Processing Card: SRX5K-SPC-4-15-320, SRX5K-SPC3 Module Interface Card: SRX5K-IOC4-10G, SRX5K-MPC3-40G10G
SRX5800	SRX5800	JUNOS 22.2R3-S1.9	Intel Xeon C5518, Intel Xeon E5-2658 v4, Intel Xeon CPU E5-2608L v3	Routing Engine: SRX5K-RE3-128n Switch Control Board: SRX5K-SCB3, SRX5K-SCB4 Service Processing Card: SRX5K-SPC-4-15-320, SRX5K-SPC3 Module Interface Card: SRX5K-IOC4-10G, SRX5K-MPC3-40G10G

Table 2: Tested Module Identification – Hardware

Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets): N/A

The module is not classified as software, firmware, or hybrid; thus, this section is not applicable.

Tested Module Identification – Hybrid Disjoint Hardware: N/A

The module is not classified as hybrid disjoint hardware; thus, this section is not applicable.

Tested Operational Environments - Software, Firmware, Hybrid: N/A

The module is not classified as software, firmware, or hybrid; thus, this section is not applicable.

Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid: N/A

There are no vendor-affirmed operational environments claimed.

CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when so ported if the specific operational environment is not listed on the validation certificate.

2.3 Excluded Components

No components are excluded from the requirements of FIPS 140-3.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
JUNOS-FIPS-MODE	Approved mode of operation enabled by following the configuration commands in Section 11.1	Approved	Suffix string :fips in the cli prompt

Table 3: Modes List and Description

Once the module has been securely initialized following the instructions provided in Section 11.1, the module is in approved mode of operation. Failure to follow the secure initialization instructions results in the module being in a non-compliant state which is out of scope of the validation.

2.5 Algorithms

Approved Algorithms:

Although the module may have been tested for additional algorithms or modes, only those listed below are utilized by the module.

OpenSSL Approved Cryptographic Functions

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A3693	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CTR	A3693	Direction - Decrypt, Encrypt Key Length - 128, 192, 256 Payload Length - Payload Length: 8-128 Increment 8 Supports Counter larger than maximum value - No Incremental Counter - Yes Counter Tests Performed - Yes	SP 800-38A
ECDSA KeyGen (FIPS186-4)	A3693	Curve - P-256, P-384, P-521 Secret Generation Mode - Testing Candidates	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A3693	Curve - P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A3693	Curve - P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A3693	Curve - P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-4

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA-1	A3693	MAC - MAC: 160 Key Length - Key Length: 160	FIPS 198-1
HMAC-SHA2-256	A3693	MAC - MAC: 256 Key Length - Key Length: 256	FIPS 198-1
HMAC-SHA2-512	A3693	MAC - MAC: 512 Key Length - Key Length: 512	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A3610	Domain Parameter Generation Methods - P-256, P-384, P-521 Hash Function Z - SHA2-256, SHA2-384, SHA2-512 Scheme - ephemeralUnified - KAS Role - initiator	SP 800-56A Rev. 3
RSA KeyGen (FIPS186-4)	A3693	Key Generation Mode - B.3.3 Modulo - 2048, 4096 Primality Tests - Table C.2 Info Generated By Server - No Public Exponent Mode - Fixed Fixed Public Exponent - 010001 Private Key Format - Standard	FIPS 186-4
RSA SigGen (FIPS186-4)	A3693	Signature Type - PKCS 1.5 Modulo - 2048, 4096 Hash Pair - Hash Algorithm - SHA2-256	FIPS 186-4
RSA SigVer (FIPS186-4)	A3693	Signature Type - PKCS 1.5 Modulo - 2048, 4096 Hash Pair - Hash Algorithm - SHA2-256 Public Exponent Mode - Fixed Fixed Public Exponent - 010001	FIPS 186-4
SHA-1	A3693	Message Length - Message Length: 0-65536 Increment 8 Function - SHA1	FIPS 180-4
SHA2-256	A3693	Message Length - Message Length: 0-65536 Increment 8 Function - SHA2	FIPS 180-4
SHA2-384	A3693	Message Length - Message Length: 0-65536 Increment 8 Function - SHA2	FIPS 180-4

Algorithm	CAVP Cert	Properties	Reference
SHA2-512	A3693	Message Length - Message Length: 0-65536 Increment 8 Function - SHA2	FIPS 180-4

Table 4: Approved Algorithms - OpenSSL Approved Cryptographic Functions

Kernel Approved Cryptographic Functions

Algorithm	CAVP Cert	Properties	Reference
HMAC DRBG	A3493	Prediction Resistance - Yes Supports Reseed - No Mode - SHA2-256 Entropy Input - Entropy Input: 256 Nonce - Nonce: 128 Personalization String Length - Personalization String Length: 0, 256 Additional Input - Additional Input: 0, 256 Returned Bits - 1024	SP 800-90A Rev. 1
HMAC-SHA2-256	A3493	MAC - MAC: 256 Key Length - Key Length: 160, 256	FIPS 198-1
SHA2-256	A3493	Message Length - Message Length: 0-51200 Increment 8 Function - SHA2	FIPS 180-4
SHA2-512	A3361	Message Length - Message Length: 0-51200 Increment 8 Function - SHA2	FIPS 180-4

Table 5: Approved Algorithms - Kernel Approved Cryptographic Functions

LibMD Approved Cryptographic Functions

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA-1	A3367	MAC - MAC: 160 Key Length - Key Length: 112, 160	FIPS 198-1
HMAC-SHA2-256	A3367	MAC - MAC: 256 Key Length - Key Length: 160, 256	FIPS 198-1
SHA-1	A3367	Message Length - Message Length: 0-51200 Increment 8 Function - SHA1	FIPS 180-4
SHA2-256	A3367	Message Length - Message Length: 0-51200 Increment 8 Function - SHA2	FIPS 180-4
SHA2-512	A3367	Message Length - Message Length: 0-65536 Increment 8 Function - SHA2	FIPS 180-4

Table 6: Approved Algorithms - LibMD Approved Cryptographic Functions

OpenSSH Approved Cryptographic Functions

Algorithm	CAVP Cert	Properties	Reference
KDF SSH (CVL)	A4271	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA-1, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1

Table 7: Approved Algorithms - OpenSSH Approved Cryptographic Functions

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Key type:Asymmetric	Junos 22.2R1 - OpenSSL	SP 800-133 Rev.2 Section 4, example 1 direct output from DRBG.

Table 8: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

Non-Approved, Not Allowed Algorithms:

N/A for this module.

2.6 Security Function Implementations

The module implements the security functions listed in the following table.

Name	Type	Description	Properties	Algorithms
Enc/Dec (SSH)	BC-UnAuth	Unauthenticated encryption for SSH		AES-CBC: (A3693) AES-CTR: (A3693)
KAS-SSC (SSH)	KAS-SSC	Key Agreement Scheme Shared Secret Computation for SSH		KAS-ECC-SSC Sp800-56Ar3: (A3610)
ECDSA SigGen (SSH)	DigSig-SigGen	Signature Generation for peer authentication in SSH		ECDSA SigGen (FIPS186-4): (A3693) SHA2-256: (A3693)

Name	Type	Description	Properties	Algorithms
				SHA2-384: (A3693) SHA2-512: (A3693) HMAC DRBG: (A3493)
ECDSA SigVer (SSH)	DigSig-SigVer	Signature Verification for peer authentication in SSH		ECDSA SigVer (FIPS186-4): (A3693) SHA2-256: (A3693) SHA2-384: (A3693) SHA2-512: (A3693)
MAC (SSH)	MAC	Message Authentication for SSH		HMAC-SHA-1: (A3693) SHA2-256: (A3693) HMAC-SHA2-512: (A3693)
KDF (SSH)	KAS-135KDF	Key derivation Function for SSH		KDF SSH: (A4271) SHA-1: (A3693) SHA2-256: (A3693) SHA2-384: (A3693)
SHA (LibMD)	SHA	Message Digest Generation		SHA-1: (A3367) SHA2-256: (A3367) SHA2-512: (A3367)
MAC (LibMD)	MAC	Message authentication		HMAC-SHA-1: (A3367) HMAC-SHA2-256: (A3367)
DRBG (Kernel)	DRBG	Random Bit Generation		HMAC DRBG: (A3493) HMAC-SHA2-256: (A3493) SHA2-256: (A3493)
SHA (Kernel)	SHA	Entropy source conditioning component		SHA2-512: (A3361)
ECDSA KeyGen (PKID)	AsymKeyPair-KeyGen AsymKeyPair-KeyVer	ECDSA Key Generation used for SSH when authentication keys are internally generated		ECDSA KeyGen (FIPS186-4): (A3693) ECDSA KeyVer (FIPS186-4): (A3693) CKG: () Key type:

Name	Type	Description	Properties	Algorithms
				Asymmetric HMAC DRBG: (A3493)
RSA KeyGen (PKID)	AsymKeyPair-KeyGen	RSA Key generation used for SSH when authentication keys are internally generated		RSA KeyGen (FIPS186-4): (A3693) CKG: () Key type: Asymmetric HMAC DRBG: (A3493)
RSA SigGen (SSH)	DigSig-SigGen	RSA Signature Generation for SSH		RSA SigGen (FIPS186-4): (A3693)
RSA SigVer (SSH)	DigSig-SigVer	RSA Signature verification for SSH		RSA SigVer (FIPS186-4): (A3693)
Verify image	DigSig-SigVer	Verification of software image		ECDSA SigVer (FIPS186-4): (A3693) SHA2-256: (A3693) SHA2-384: (A3693)
Full KAS (SSH)	KAS-Full	Full Key Agreement for SSH		KAS-ECC-SSC Sp800-56Ar3: (A3610) KDF SSH: (A4271) SHA-1: (A3693) SHA2-256: (A3693) SHA2-384: (A3693)
KAS-ECC KeyGen (SSH)	AsymKeyPair-KeyGen AsymKeyPair-KeyVer	KAS-ECC Key Pair Generation for SSH		ECDSA KeyGen (FIPS186-4): (A3693) ECDSA KeyVer (FIPS186-4): (A3693) CKG: () Key type: Asymmetric HMAC DRBG: (A3493)
ENT	ENT-ESV	Entropy Source		SHA2-512: (A3361)

Name	Type	Description	Properties	Algorithms
KTS (SSH)	KTS-Wrap	Key transport using SSH as per IG D.G provisions	KTS:128, 256, 384, 521, 2048, 3072 or 4096 bits keys provide between 112 and 256 bits of encryption strength	AES-CBC: (A3693) AES-CTR: (A3693) HMAC-SHA-1: (A3693) HMAC-SHA2-256: (A3693) HMAC-SHA2-512: (A3693)

Table 9: Security Function Implementations

2.7 Algorithm Specific Information

The module includes RSA and ECDSA algorithms that have been validated using FIPS 186-4 CAVP tests, which are mathematically identical to FIPS 186-5 CAVP tests. Per IG C.K, all RSA and ECDSA algorithms implemented by the module are claimed compliant with FIPS 186-5. The module complies with IG C.F. RSA Key Generation, Signature Generation and Signature Verification have been tested and validated using CAVP testing for all implemented modulus lengths (2048, 3072 and 4096 bits). The number of Miller-Rabin tests used for primality testing as part of RSA Key Generation is consistent with Table C.3.

The module implements the following Approved key agreement methods which have been CAVP tested and validated:

- KAS-ECC per SP 800-56A Rev. 3 (FIPS 140-3 IG D.F Scenario 2, path 2).

The module obtains the FIPS 140-3 IG D.F required key agreement assurances in accordance with Section 5.6.2 of SP800-56A Rev. 3. All the key agreement protocols implemented by the module are Diffie-Hellman based.

The module includes approved KDF algorithms for the SSH protocols. No parts of these protocols, other than the approved cryptographic algorithms and the KDFs, have been tested by the CAVP and CMVP.

2.8 RBG and Entropy

Cert Number	Vendor Name
E56	Juniper Networks

Table 10: Entropy Certificates

N/A for this module.

The entropy source is used to seed the module's HMAC DRBG with the minimum required 256-bits of entropy. Each 512-bit block of conditioned output from the entropy source contains 448 bits of entropy. The HMAC DRBG is used for all random data required by the module, including key generation.

There are no initialization procedures required by the users of the module to operate the entropy source in a compliant manner. The module complies to the ESV Public Use document of the validated entropy source (Cert. E56).

2.9 Key Generation

The cryptographic module implements the key generation methods listed above in the Security Functions implementation table.

2.10 Key Establishment

The cryptographic module implements the key establishment methods listed above in the Security Functions implementation table.

2.11 Industry Protocols

The cryptographic module supports the protocols listed below. No part of these protocols, other than the approved cryptographic algorithms and the KDFs, have been tested by the CAVP and CMVP. The SSH algorithms allow independent selection of key exchange, authentication, cipher, and integrity. In reference to the supported protocols table below, each column of options for a given protocol is independent and may be used in any viable combination.

Protocol	Key Exchange	Auth	Cipher	Integrity
SSHv2	KAS-ECC (P-256, P-384, P-521)	RSA 2048 ECDSA P-256	AES CBC 128/192/256 AES CTR 128/192/256	HMAC-SHA-1 HMAC-SHA2-256 HMAC-SHA2-512

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

The following table maps each physical interface to one or more logical interface types defined in the FIPS 140-3 standard.

Physical Port	Logical Interface(s)	Data That Passes
Ethernet (data)	Data Input Data Output Control Input Status Output	LAN communications
Ethernet (mgmt.)	Data Input Data Output Control Input Status Output	Remote management
Serial	Control Input Status Output	Local management
Reset Button	Control Input	Reset
ToD	Control Input Status Output	RJ-45 Time of Day Port
BITS	Control Input Status Output	BITS RJ-45 port
GPS	Control Input Status Output	10 Mhz clock synchronization
PPS	Control Input Control Output	1 pulse per second
Offline	Control Input	Offline button
LED	Status Output	Status indicator lighting
Power	Power	Power

Table 11: Ports and Interfaces

4 Roles, Services, and Authentication

4.1 Authentication Methods

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
Password authentication	User and CO authentication via SSH or console. Minimum of 10 ASCII character passwords.	SHA (LibMD)	Probability of guessing: 1/(96^10) 1/1,000,000.	Timed access mechanism allows max of 10 attempts / min. Probability of guessing: 10/(96^10) 1/100,000.
Signature authentication	User/CO authentication via SSH.	ECDSA SigVer (SSH)	Strength of signature algorithm, minimum 112-bits. Probability of success for random attempt: 1/(2^112) 1/1,000,000.	A rate of 1 CPU cycle per failed authentication for the Intel Xeon E3-1200 v2 processor (4 cores, 3.1 GHz) allows for the probability of success by brute-force attack: 60 x 4 x 3.1 x 10^9 x 1/(2^112) 1/100,000.

Table 12: Authentication Methods

The module enforces the separation of roles using either password-based authentication or signature-based authentication.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
User	Role	Monitor	Password authentication Signature authentication
Cryptographic Officer	Role	CO	Password authentication Signature authentication

Table 13: Roles

The module supports two roles: Cryptographic Officer (CO) and User. The module supports role-based operator authentication for assuming these roles, using methods specified in Section 4.1. The module supports concurrent operators but does not support a maintenance role and/or bypass capability.

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.

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Configure security	Security relevant configuration	:fips suffix in CLI prompt	CLI commands	Status	SHA (LibMD) MAC (LibMD) DRBG (Kernel) SHA (Kernel) ECDSA KeyGen (PKID) RSA KeyGen (PKID) ENT	Cryptographic Officer - HMAC DRBG V value: E - HMAC DRBG Key value: E - HMAC DRBG Entropy Input: E - HMAC DRBG Seed: E - CO-PW: W - User-PW: W - SSH-Priv: G,R,W
Configure	Non-security relevant configuration	None	CLI commands	Status	None	Cryptographic Officer
Show status	Show status	None	CLI command	Status	None	Cryptographic Officer User
Zeroize	Zeroize / destroy all CSPs	None	CLI command	None (completion indicator is implicitly provided by the module rebooting)	None	Cryptographic Officer - HMAC DRBG V value: Z - HMAC DRBG Key value: Z - HMAC DRBG Seed: Z - HMAC DRBG Entropy Input: Z - SSH-DH-Shared-Secret: Z - SSH-Priv: Z - SSH-SEKs: Z - CO-PW: Z - User-PW: Z - SSH-PUB: Z - Auth-User Pub:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Z - Root-CA: Z - Package-CA: Z - SSH-DH-PUB (self): Z - SSH-DH-PUB (peer): Z
SSH connect	Initiate SSH connection for SSH monitoring and control (CLI)	:fips suffix in CLI prompt	SSH packets	SSH packets, status	Enc/Dec (SSH) KAS-SSC (SSH) ECDSA SigGen (SSH) ECDSA SigVer (SSH) MAC (SSH) KDF (SSH) DRBG (Kernel) RSA SigGen (SSH) RSA SigVer (SSH) Full KAS (SSH) KAS-ECC KeyGen (SSH) ENT KTS (SSH)	Cryptographic Officer - HMAC DRBG V value: E - HMAC DRBG Key value: E - HMAC DRBG Entropy Input: E - HMAC DRBG Seed: E - SSH-DH-Shared-Secret: G,E - SSH-DH-priv: G,E - SSH-SEKs: G,E - Auth-CO Pub: E - SSH-Priv: E - CO-PW: E - SSH-DH-PUB (self): G - SSH-DH-PUB (peer): E User - HMAC DRBG V value: E - HMAC DRBG Key value: E - HMAC DRBG Entropy Input: E - HMAC DRBG Seed: E - SSH-Priv: E - User-PW: E - SSH-DH-Shared-Secret: G,E - SSH-DH-priv: G,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- SSH-SEKs: G - SSH-DH-PUB (self): G - SSH-DH-PUB (peer): E - Auth-User Pub: E
Console access	Console monitoring and control (CLI)	None	CLI command	Status	None	Cryptographic Officer - CO-PW: E User - User-PW: R,E
Remote reset	Software initiated reset	None	CLI command	Status	None	Cryptographic Officer - HMAC DRBG V value: Z - HMAC DRBG Key value: Z - HMAC DRBG Entropy Input: Z - HMAC DRBG Seed: Z - SSH-DH-Shared-Secret: Z - SSH-DH-priv: Z - SSH-SEKs: Z - SSH-DH-PUB (self): Z - SSH-DH-PUB (peer): Z
Local reset	Hardware reset or power cycle	None	Manual power cycle	Status	None	Unauthenticated - HMAC DRBG V value: Z - HMAC DRBG Key value: Z - HMAC DRBG Entropy Input: Z - HMAC DRBG Seed: Z - SSH-DH-Shared-Secret: Z - SSH-SEKs: Z - SSH-DH-PUB (self): Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- SSH-DH-PUB (peer): Z
Traffic	Traffic requiring no cryptographic services	None	Traffic in	Traffic out	None	Unauthenticated
Load Image	Loading of firmware image	:fips suffix in CLI prompt	CLI command	status	Verify image	Cryptographic Officer - Root-CA: E - Package-CA: E
Perform self-tests	On-demand self-tests of all pre-operational and conditional algorithm self-tests	None	Local or remote reset	status	None	Cryptographic Officer User Unauthenticated
Show version	Show firmware version	None	CLI command	Status	None	Cryptographic Officer User

Table 14: Approved Services

4.4 Non-Approved Services

N/A for this module.

4.5 External Software/Firmware Loaded

The module includes a firmware load service to support necessary updates. Only the CO can install the new image using the CLI as described in Section 11.1. The loaded firmware is a complete image replacement and constitutes an entirely new module and version of Junos OS which would require a separate FIPS 140-3 validation.

5 Software/Firmware Security

5.1 Integrity Techniques

The cryptographic module implements an approved firmware integrity self-test that uses ECDSA P-256 with SHA2-256 to ensure the integrity of all Junos OS firmware components. The self-test is automatically run on power-up. It can also be run on demand by the module's operator by power cycling the module. When the integrity check fails, the module enters an error state (kernel panic) which can only be exited by power-cycling the module.

5.2 Initiate on Demand

The self-test is automatically run on power-up. It can also be run on demand by the module's operator by power cycling the module.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Non-Modifiable

How Requirements are Satisfied:

The module consists of hardware containing a non-modifiable operational environment as per the FIPS 140-3 definitions. It includes a firmware load service to support necessary updates. The loaded firmware is a complete image replacement and constitutes an entirely new module and version of Junos OS which would require a separate FIPS 140-3 validation.

6.2 Configuration Settings and Restrictions

There are no security rules, settings, or restrictions to the configuration of the operational environment beyond the initialization instructions to set the module in approved mode.

7 Physical Security

7.1 Mechanisms and Actions Required

Mechanism	Inspection Frequency	Inspection Guidance
Tamper seals (part # JNPR-FIPS-TAMPER-LBLS)	Once per month by the Cryptographic Officer	Seals should be free of any tamper evidence.
Opaque metal enclosure.	n/a	n/a

Table 15: Mechanisms and Actions Required

The module's 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 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.

Tamper-evident seals allow the operator to tell if the enclosure has been breached. These seals are not factory-installed and must be applied by the Cryptographic Officer. (Seals are available for order from Juniper using part number JNPR-FIPS-TAMPER-LBLS.) The tamper-evident seals shall be installed for the module to operate in approved mode.

The Cryptographic Officer is responsible for securing and having control at all times of any unused seals and the direct control and observation of any changes to the module such as reconfigurations where the tamper-evident seals or security appliances are removed or installed to ensure the security of the module is maintained during such changes and the module is returned to an approved mode of operation.

If the Cryptographic Officer observes tamper evidence, it shall be assumed that the device has been compromised. The Cryptographic Officer shall retain control of the module and perform zeroization of the module's CSPs by following the steps in Section 9.3 and then follow the steps in Section 11.1 to place the module back into approved mode of operation.

7.2 User Placed Tamper Seals

The number of seals that need to be applied depends on the module model, as follows:

- SRX1500: 10 seals
- SRX4100 and SRX 4200: 13 seals
- SRX4600: 15 seals
- SRX5400: 20 seals
- SRX5600: 19 seals

- SRX5800: 42 seals

Placement:

SRX1500

Six tamper evident labels (TEL) must be applied to the following location:

- The front of the SRX1500 has two slot covers. The slot covers should be secured with two screws each and then tamper evident labels (TEL #1 & #2) applied as shown by the red boxes in following two figures. The TEL go from the front of the SRX1500 to the top (Figures 4 & 5).

2 Tamper labels (#5 & #6) are used to cover the USB port and two tamper labels (#3 & #4) are used to cover the High Availability port (Figure 4).

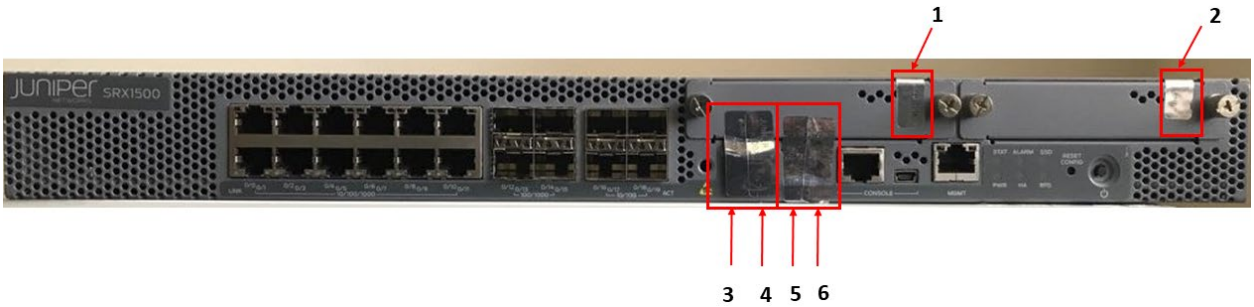

Figure 15 - SRX1500 Front View: TEL 1 - 6

- The rear of the SRX1500 has two TELs (TEL #7 and TEL #8). The TEL #7, at the top of the rear-view wraps to the top of the device and covers the fourth screw from side containing the power supply (see Figure 7). TEL #8 wraps from the rear of the SRX1500, on the SSD slot cover, to the bottom of the SRX1500 (see Figure 8).

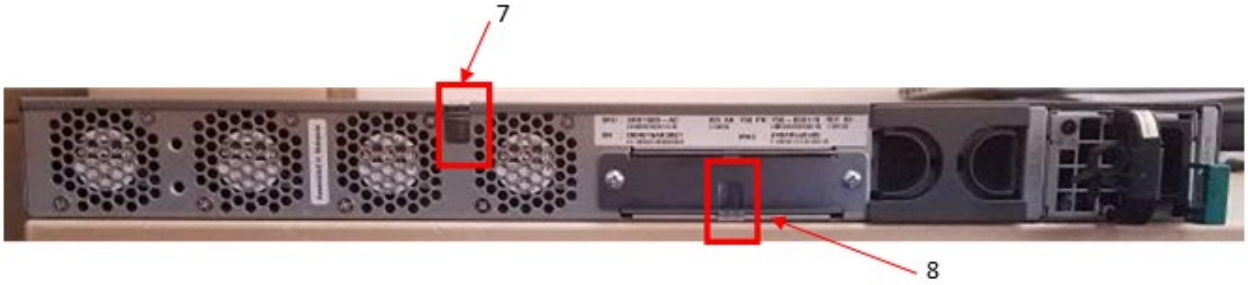

Figure 16 - SRX1500 Rear View: TEL 7 & 8

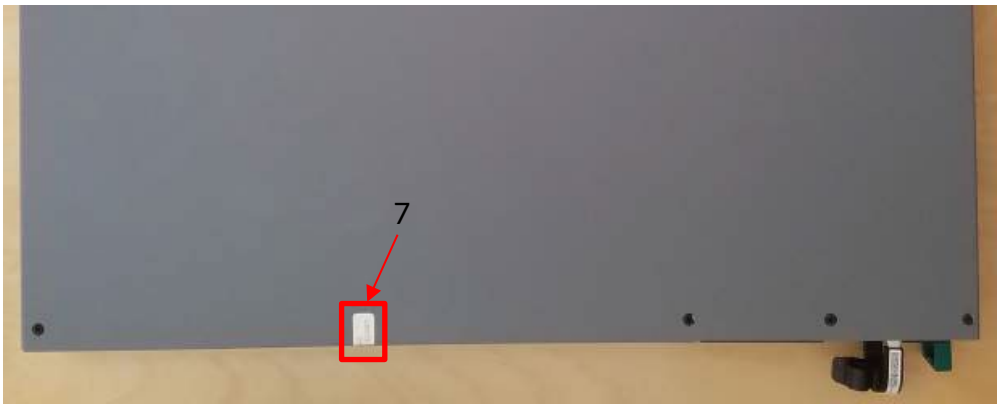

Figure 17 - SRX1500 Top - Rear View: TEL 7

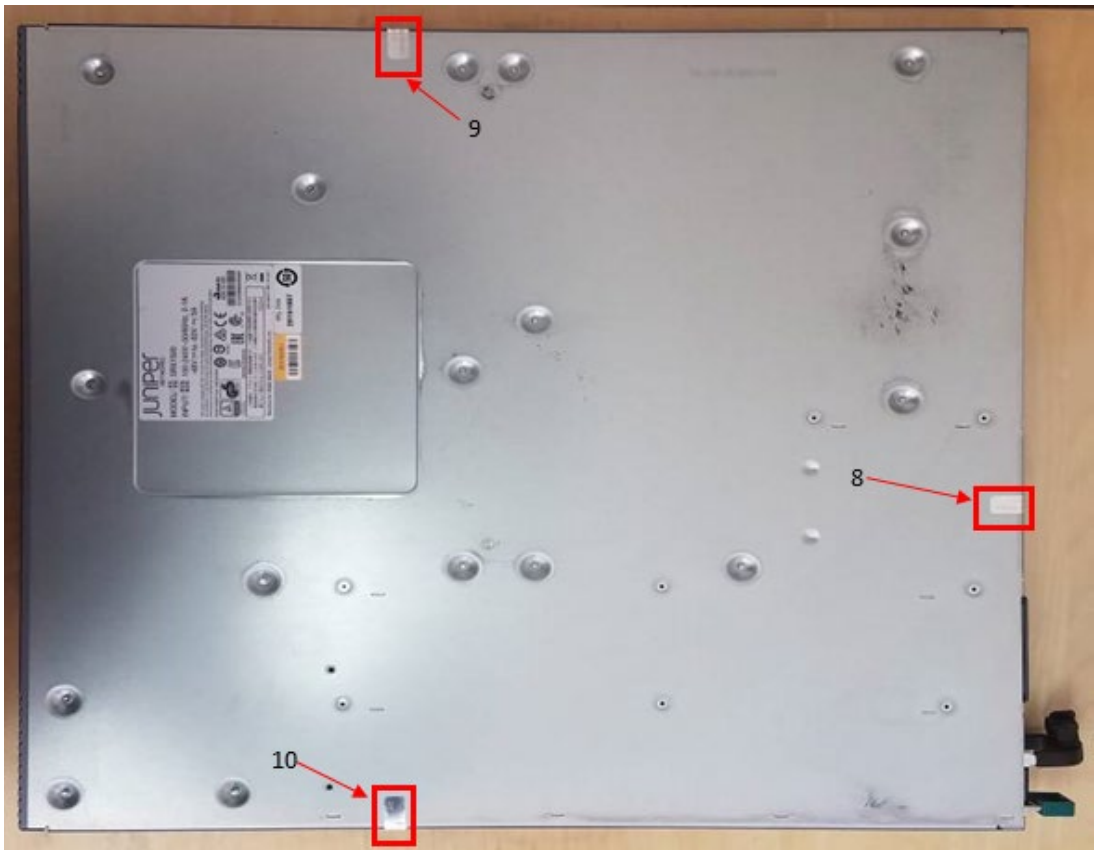

Figure 18 - SRX1500 Bottom View: TEL 8, 9 & 10

- TEL #9 and TEL #10 cover the indicated screw on the left and right side of the SRX1500 (Figure 19 and Figure 20) and wrap to the bottom of the SRX1500 as shown in Figure 8.

Figure 19 - SRX1500 Right Side View: TEL 9

Figure 20 - SRX1500 Left Side View: TEL 10

SRX4100 & SRX4200

The placement of the tamper evident labels for the SRX4100 and SRX4200 are the same in that the outside of the devices is identical. Thirteen tamper-evident seals must be applied to the following locations:

- The top of the chassis, covering one screw on the top-back left and one screw on the top-back right (TEL #1 and TEL #2). The TELs cover the screws on the top of the chassis and wrap down each side of the chassis.

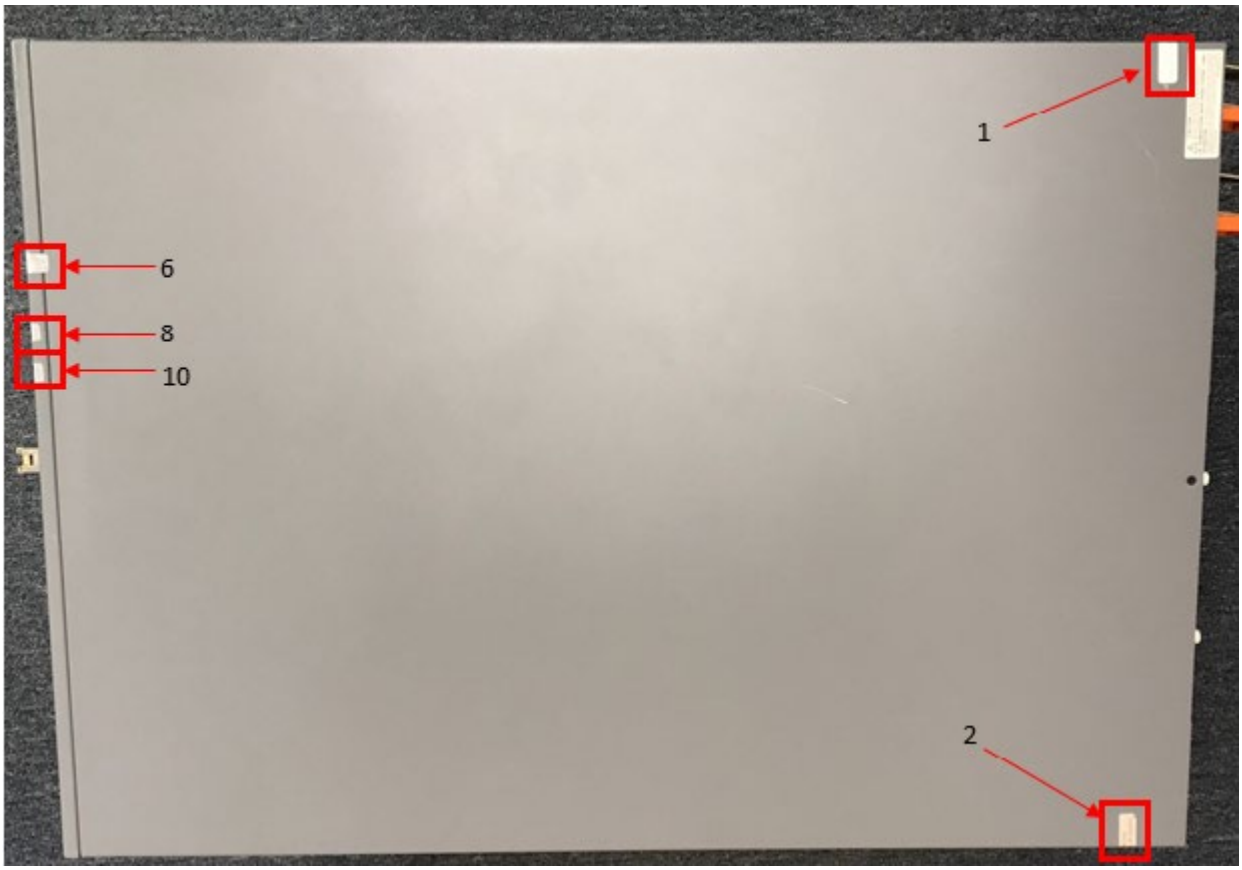

Figure 21 - SRX4100 & SRX4200 Top View: TEL 1, 2, 6, 8 & 10

Figure 22 - SRX4100 & SRX4200 Left-Side View: TEL 1

Figure 23 - SRX4100 & SRX4200 Right-Side View: TEL 2

- Bottom chassis, covering 3 screws that secure the faceplates on the front of the chassis. TEL #3, #4, #5 are entirely on the bottom of the chassis they do not wrap around to any other portion of the chassis.

Figure 24 - SRX4100 & SRX4200 Bottom View: TEL 3, 4, 5

- Tamper evident seals 6 & 7 cover the two USB ports on the front of the SRX4100 and the SRX4200
- Two tamper evident labels cover each HA port. Tamper evident labels #8 & #9 cover one HA port and tamper evident labels #10 & #11 cover the second HA port.

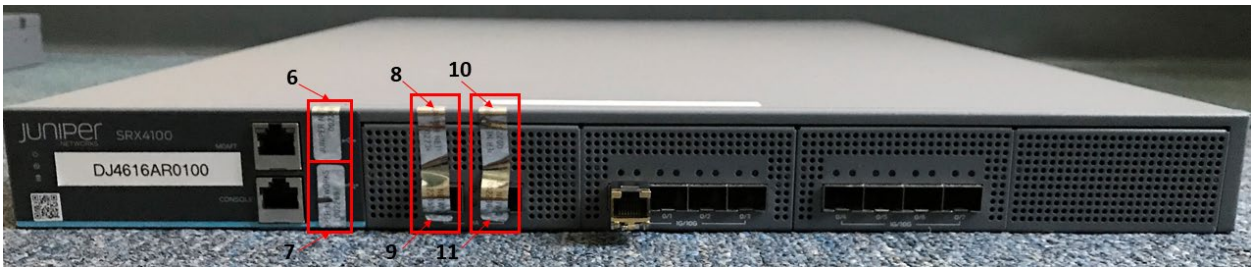

Figure 25 - SRX4100 & SRX4200 Front View: TEL 6-11

- The rear of the SRX4100 & SRX4200 require 2 tamper evident labels (TEL #12 and #13) as shown in Figure 26. Each label is applied on the power supply plate and wraps around the bottom.

Figure 26 - SRX4100 & SRX4200 Rear View: TEL 12-13

SRX4600

Fifteen tamper evident labels (TEL) must be applied to the following location:

- The front of the SRX4600 has 4 HA ports, 1 USB port, and 2 Solid State Drives (SSDs) that must be protected with 8 tamper evident labels.
- Referring to Figure 27 and Figure 28, the front panel requires 4 tamper evident labels (#1 - #4) cover the HA ports, 2 tamper evident labels cover the USB port and top screw (#5, #6), 1 tamper evident label (#7) to cover the first SSD, and 1 tamper evident label (#8) to cover the second SSD.

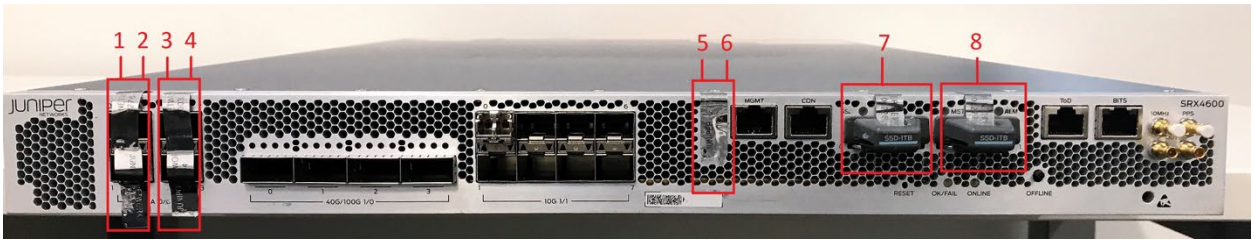

Figure 27 - SRX4600 Front View: TEL 1 – 8

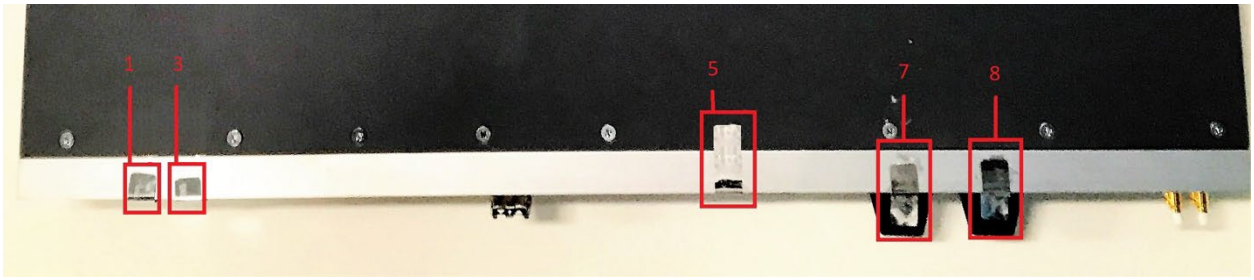

Figure 28 - SRX4600 Top Front View: TEL 1, 3, 5, 7, 8

- The rear of the SRX4600 requires 7 tamper evident labels (TEL #9 -#15) as shown in Figure 29 TEL #9 and TEL #10 wrap over the top and cover the back plate of each power supply and the adjacent chassis edge. Each of TEL#11 to TEL #15 = wraps over the top and attaches to a fan cover.

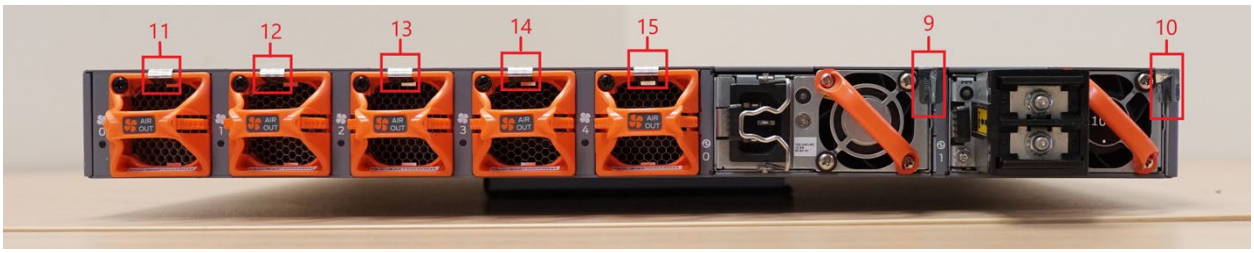

Figure 29 - SRX4600 Rear View: TEL 9-15

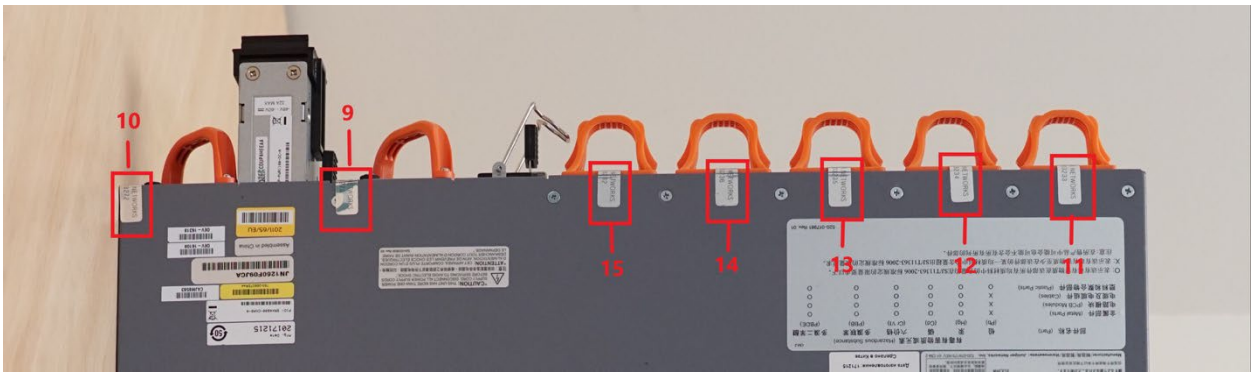

Figure 30 - SRX4600 Top Rear View: TEL 9 – 10

- The right and left sides have 1 TEL each (TEL 11 and 12) over the 4th screw from the front and wrapping around to the bottom. Figures 6 & 7 show the placement of the side TELs.

Figure 31 - SRX4600 Right Side View: TEL 12

Figure 32 - SRX4600 Left Side View: TEL 11

- The bottom view (Figure 8) shows the TELs wrapping around from the front and sides of the SRX4600.

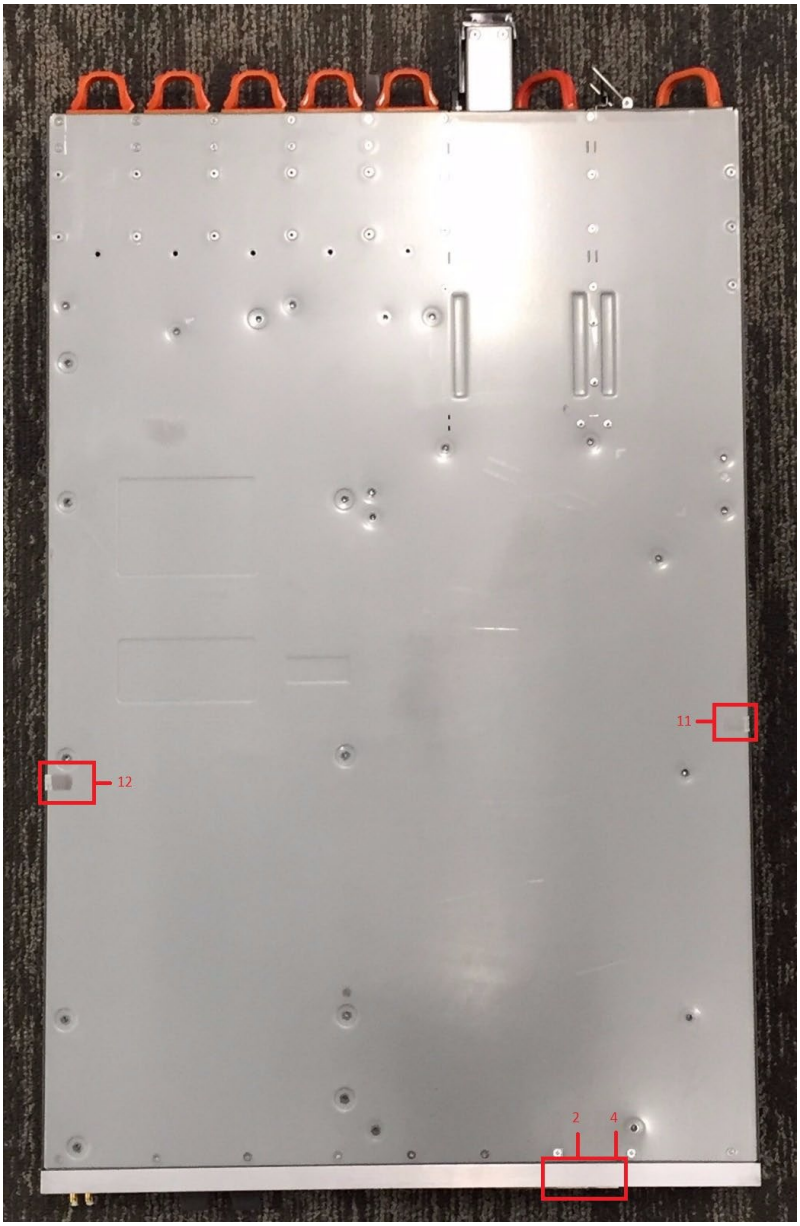

Figure 33 - SRX4600 Bottom View: TEL 2, 4, 11, 12

SRX5400

Tamper-evident seals shall be applied to the following locations:

- Front Pane (10 seals, TEL 1-10).

Two seals (2), Vertical Covering the screws on the information panel at the top of the device.

Seven (7) seals, vertical, connection each expansion plane to its neighbors and the top and button plane to the chassis.

One (1) seal, vertical covering the USB port.

- Back Pane (8 seals TEL 11-18)

Four (4) seals, vertical: one on each of the top four sub-panes, extending to the large chassis plate below.

Two (2) seals, vertical: on the horizontal screwed-in plate resting on the large central chassis. Placed over screws.

Two (2) seals, horizontal: placed on the low side sub-panes, extending to the large central chassis area and wrapping around to the neighbouring side panes.

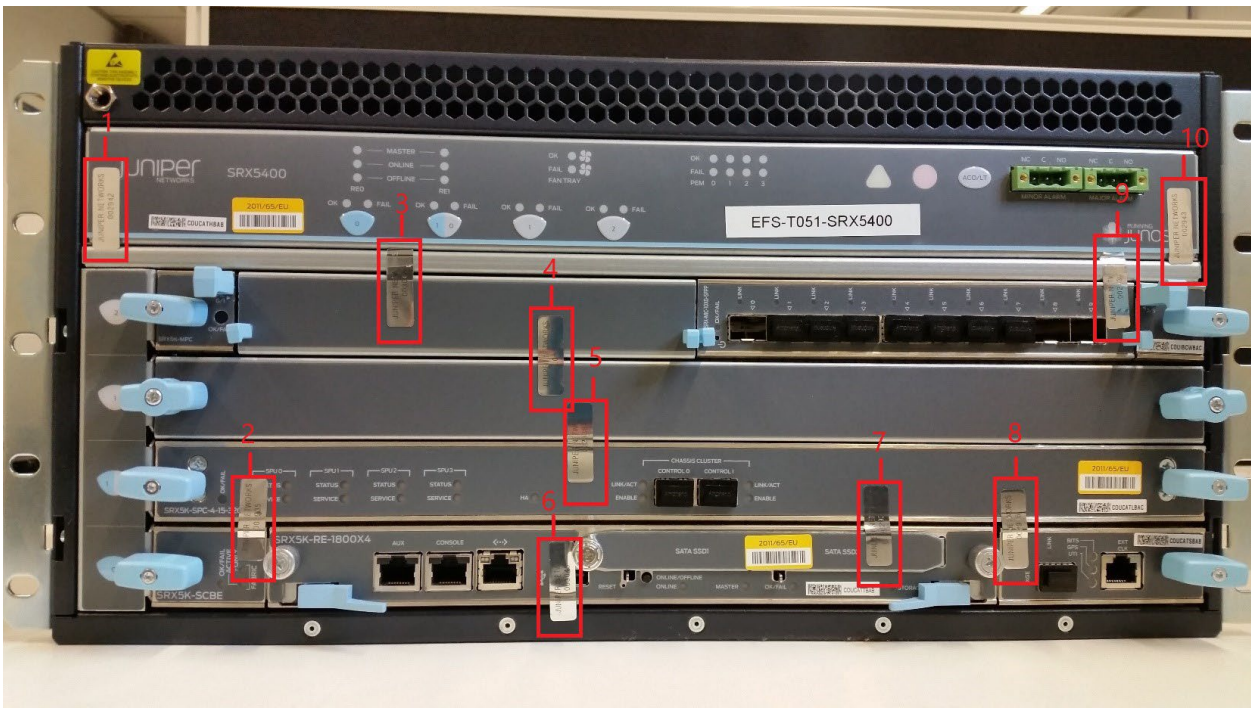

Figure 34 - SRX5400 Front View: TEL 1-10

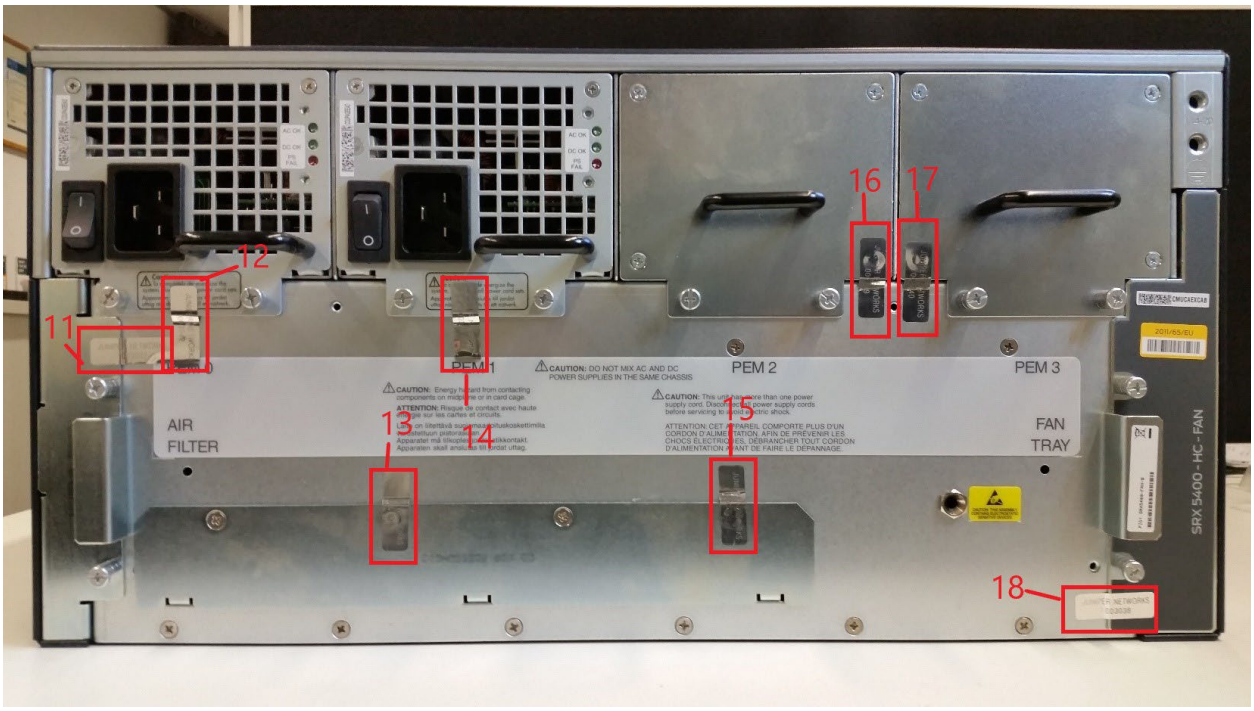

Figure 35 - SRX5400 Rear View: TEL 11-18

SRX5600

Tamper-evident seals must be applied to the following locations:

- Front Pane (11 seals, TEL 1-11)

Nine (9) seals, vertical: one for each horizontal sub-pane (excluding the honeycomb plate on the top and the thin sub-pane a little below). The seals should attach to vertically adjacent sub-panes. The extra on the bottom attaches to the lowermost sub-pane and wraps around attaching to the bottom pane. It should be ensured that one of the seals spans across the thin plate with ample extra distance on each side.

Two (2) seals, horizontal attach to the uppermost sub-pane and wraps around attaching to the side panels of the chassis.

- USB Port (1 seal, TEL 12)

One (1) Seal, Vertical Covering the front USB port.

- Back Pane (7 seals, TEL 13-18)

Four (4) seals, vertical: one on each of the upper four sub-panes, attaching to the large plate below.

Two (2) seals, horizontal: one on each of the vertical side sub-panes, extending to side the panes.

One (1) Seal, Horizontal: connecting the small access pane on the lower right of the chassis to the main panel.

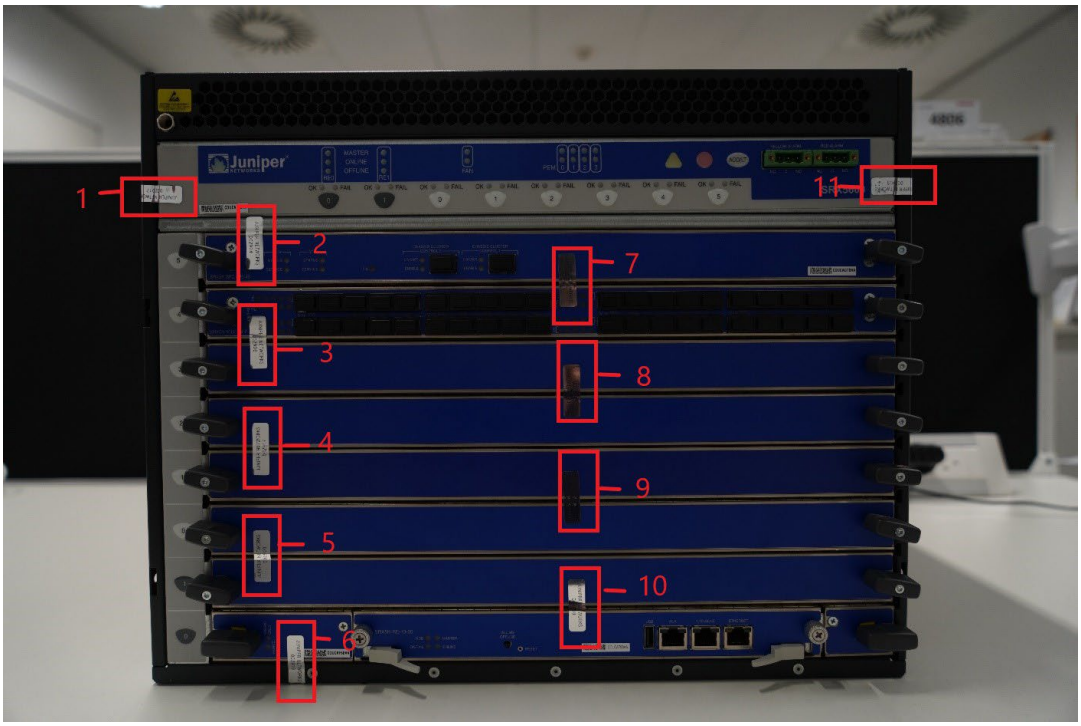

Figure 36 - SRX5600 Front View: TEL 1-11

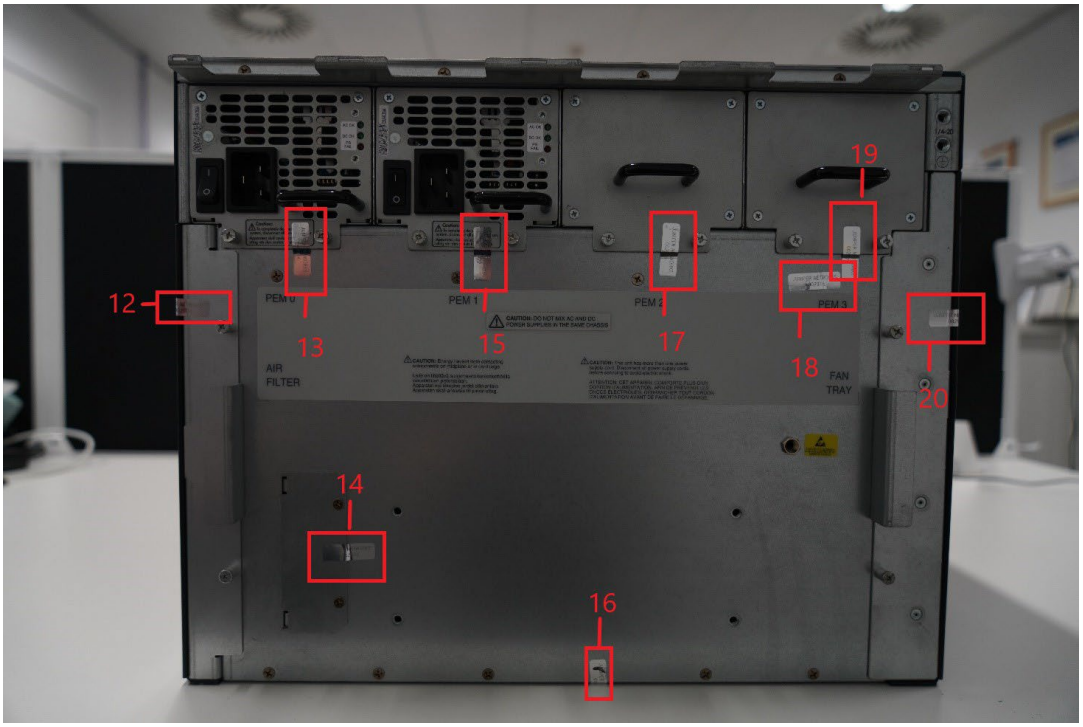

Figure 37 - SRX5600 Rear View: TEL 12-20

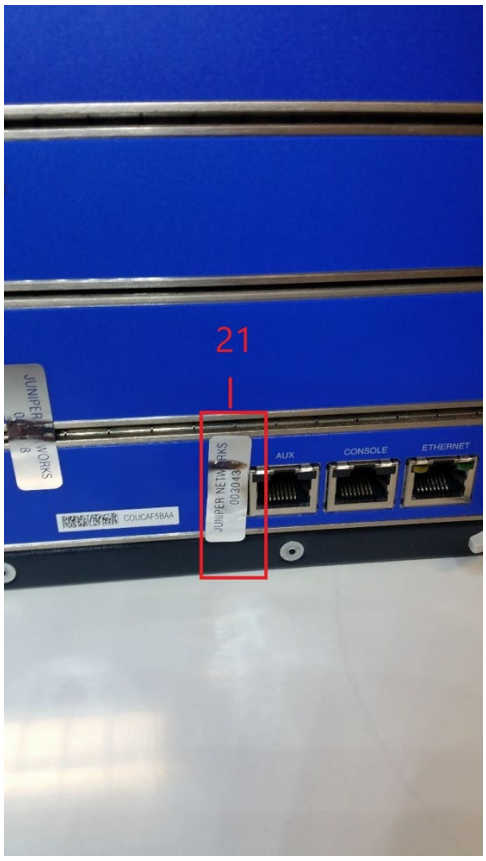

Figure 38 - SRX5600 USB Port: TEL 21

SRX5800

Tamper-evident seals shall be applied to the following locations:

- Front Panel (36 seals: TEL 1-36)

Thirty (30) seals, horizontal: two on each of the long vertical sub-panes, extending to the neighboring two. If on an end sub-pane, seal should wrap around to the side.

Three (3) seals, vertical: One over each of the thin panes – two near the bottom, one near the top of the lower half.

Two (2) seals, vertical: both on the console area at the top of the module, one extending to the top and the other extending to the chassis area below.

One (1) Seal, Diagonal covering the front USB port.

- Back Pane (6 seals: TEL 37-42)

Five (5) seals, horizontal: Three spanning the gaps between the vertical sub-panels, and then two more, one each on the far edges of the left and right panels. (These last two should wrap around to the sides.)

One (1) Seal, Vertical: At the top of the case connecting the DC system pane to the mesh fan cover.

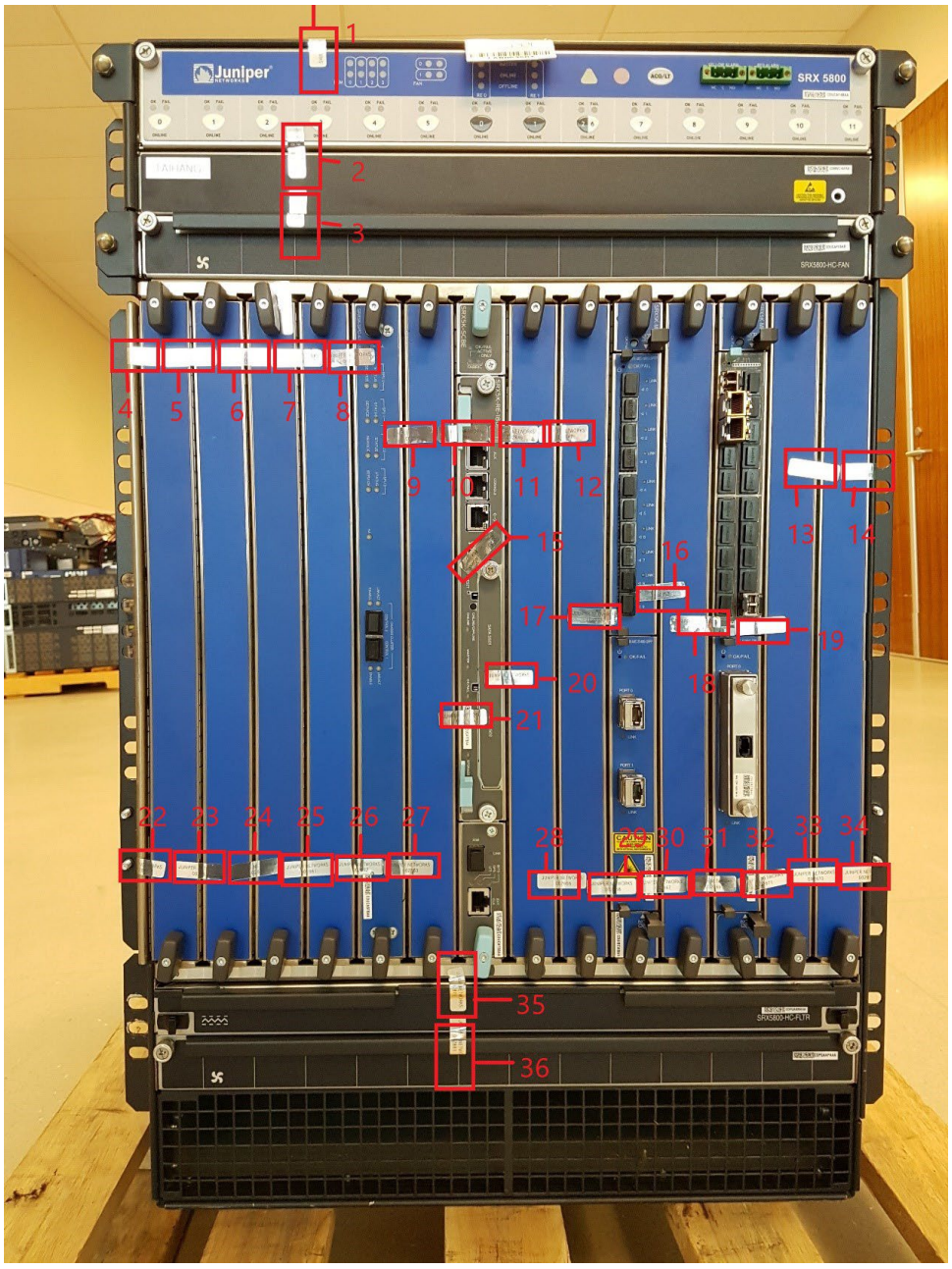

Figure 39 - SRX5800 Front View: TEL 1-36

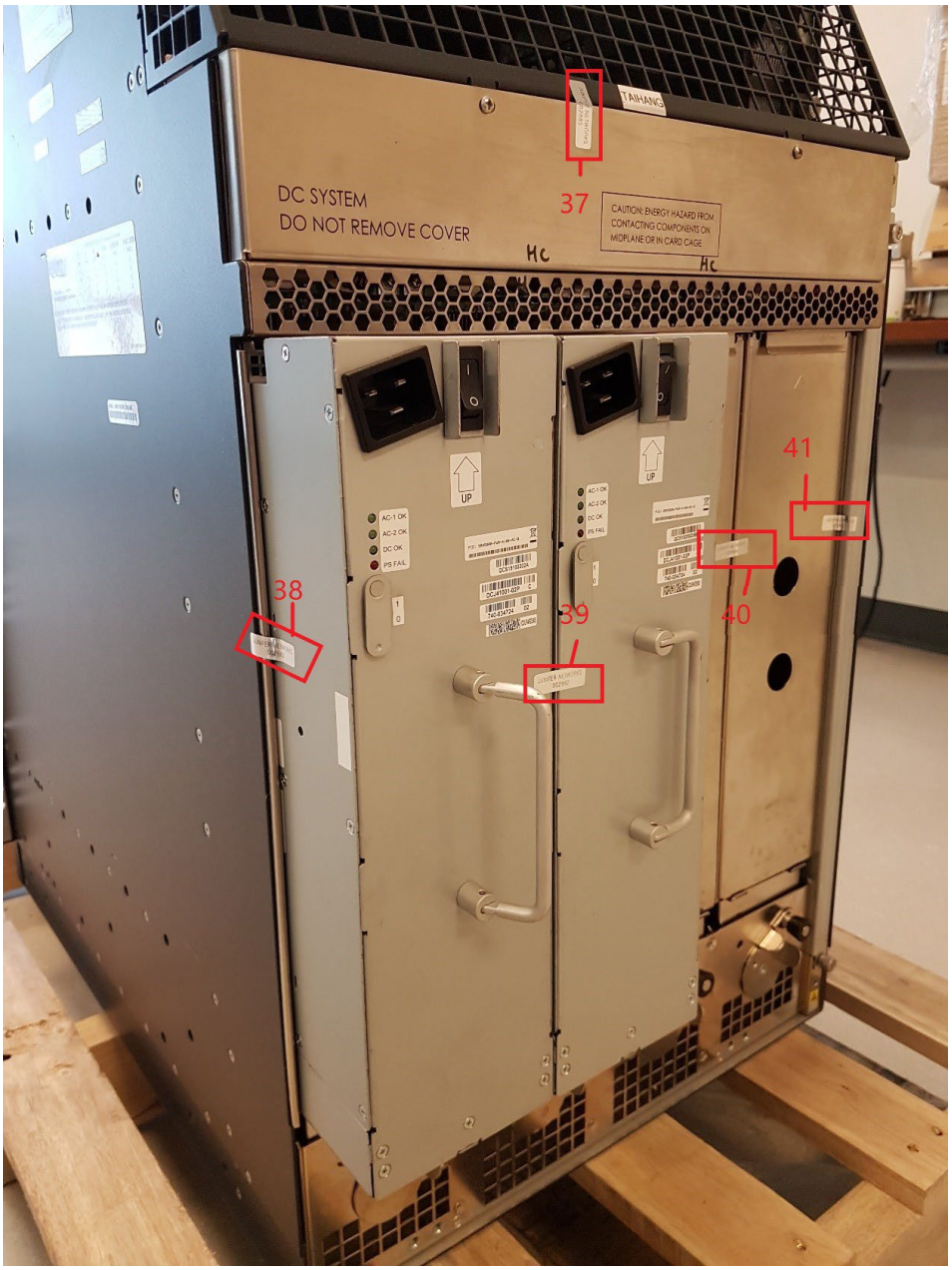

Figure 40 - SRX5800 Rear View: TEL 37-41

Figure 41 - SRX5800 Rear View: TEL 37,39,40-42

Surface Preparation:

For all seal applications, the Cryptographic Officer should observe the following instructions:

- Handle the seals with care. Do not touch the adhesive side.
- Before applying a seal, ensure the location of application is clean, dry, and clear of any residue.
- Place the seal on the module, applying firm pressure across it to ensure adhesion. Allow at least 1 hour for the adhesive to cure.

Operator Responsible for Securing Unused Seals:

The Cryptographic Officer is responsible for securing and having control at all times of any unused seals.

Part Numbers:

Tamper seals have part number JNPR-FIPS-TAMPER-LBLS.

8 Non-Invasive Security

This section is not applicable, as there are currently no approved non-invasive mitigation techniques specified in ISO/IEC 19790:2012

9 Sensitive Security Parameters Management

9.1 Storage Areas

The table below lists the areas within the module's cryptographic boundary where SSPs can be stored.

Storage Area Name	Description	Persistence Type
RAM	Random Access Memory	Dynamic
SSD	Solid-Stated Drive	Dynamic

Table 16: Storage Areas

9.2 SSP Input-Output Methods

The table below lists the method used by the module for the input and output of SSPs.

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Entry via SSH	Remote CO	RAM	Encrypted	Automated	Electronic	KTS (SSH)
Manual CLI entry	Local CO	RAM	Plaintext	Manual	Direct	
Entry via console	Local CO	RAM	Plaintext	Manual	Electronic	
Output via SSH	RAM	Remote CO	Encrypted	Automated	Electronic	KTS (SSH)
Output via console	RAM	Local CO	Plaintext	Manual	Direct	
Entry as part of KAS	Remote peer	RAM	Plaintext	Automated	Electronic	
Output as part of KAS	RAM	Remote peer	Plaintext	Automated	Electronic	
Pre-loaded	Manufacturer	SSD	Plaintext	Manual	Direct	

Table 17: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Reset	Zeroisation of SSPs in RAM via invocation of local or remote reset service	RAM is volatile and all data is lost when power is taken off. Zeroisation is practically instantaneous.	Yes, both User and CO, via invocation of Local Reset or Remote Reset services

Zeroization Method	Description	Rationale	Operator Initiation
Zeroize CLI command	These command wipe clean all the SSPs/configs as well as the disk and installs a factory default firmware image	This command overwrites all data on disk and forces a power cycle	Yes, CO via invocation of zeroize CLI command
Explicit zeroize function	Zeroisation of SSPs in memory when no longer needed	Use of explicit zeroisation function destroys SSP information immediately by overwriting memory area with zeroes	No. The operator cannot directly initiate this method.

Table 18: SSP Zeroization Methods

The CO can run the following commands to zeroize the approved mode SSPs:

- For the SRX1500, SRX4100, SRX4200 and SRX4600 models:

  ```
  user@host> request system zeroize hypervisor
  ```

- For the SRX5400, SRX5600 and SRX5800 models:

  ```
  user@host> request vmhost zeroize
  ```

These command wipe clean all the SSPs/configs as well as the disk and install a factory default firmware image. After zeroizing the system, the module is no longer in a FIPS compliant state. Installation and configuration as per section 11.1 is required to enter the FIPS compliant state and enable the Approved mode of operation.

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

Zeroization commands, as described above, and power cycling are initiated by the operator. The module automatically zeroizes all SSPs when no longer required by calling explicit delete commands. Session termination is initiated by the operator or by environmental errors.

The completion of zeroization is indicated implicitly. If the zeroization is initiated using a zeroization command or explicit delete command, completion of the command indicates that zeroization has successfully completed. If the zeroization is initiated by power cycling the module, then successful reboot of the module indicates that zeroization has completed successfully. In the case of zeroization initiated by session termination, SSPs are zeroized when the session terminates, and session termination is indicated in the log.

9.4 SSPs

All SSPs used by the module are described in this section

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
HMAC DRBG V value	A critical value of the internal state of DRBG per IG D.L	256 - 256	DRBG internal state - CSP	DRBG (Kernel)		DRBG (Kernel)
HMAC DRBG Key value	A critical value of the internal state of DRBG per IG D.L	256 - 256	DRBG internal state - CSP	DRBG (Kernel)		DRBG (Kernel)
HMAC DRBG Entropy Input	A critical value of the internal state of DRBG provided by entropy source	256 - 256	Entropy source output - CSP	ENT		DRBG (Kernel)
HMAC DRBG Seed	Seed material used to seed or reseed the HMAC DRBG	256 - 256	DRBG internal state - CSP	DRBG (Kernel)		DRBG (Kernel)
SSH-DH-Shared-Secret	Shared DH value computed from the ephemeral DH key-pairs as part of SSH and used to derive session keys. P-256, P-384 and P-521	256, 384, 521 - 128, 192, 256	DH shared value - CSP		KAS-SSC (SSH)	KDF (SSH)
SSH-Priv	SSH host authentication key (ECDSA or RSA)	2048, 256, 4096, 384, 521 - 112,128, 152, 192, 256	Asymmetric private key - CSP	ECDSA KeyGen (PKID) RSA KeyGen (PKID)		ECDSA SigGen (SSH) RSA SigGen (SSH)
SSH-DH-priv	SSH DH private key used in SSH. P-256, P-384 and P-521	256, 384, 521 - 128, 192, 256	Asymmetric private key - CSP	KAS-ECC KeyGen (SSH)		KAS-SSC (SSH)
SSH-SEKs	Session keys used with SSH-2.	128, 192, 256 - 128, 192, 256	Symmetric Key - CSP	KDF (SSH)		Enc/Dec (SSH) MAC (SSH)
CO-PW	Password used to authenticate the CO	n/a - n/a	Authentication password - CSP		KTS (SSH)	SHA (LibMD)
User-PW	Password used to authenticate the User.	n/a - n/a	Authentication password - CSP		KTS (SSH)	
SSH-PUB	SSH Public Host Key	2048, 256, 4096, 384, 521 -	Asymmetric key - PSP	ECDSA KeyGen (PKID)		

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		112,128, 152, 192, 256		RSA KeyGen (PKID)		
Auth-User Pub	SSH User Authentication Public Key	2048, 256, 4096, 384, 521 - 112,128, 152, 192, 256	Asymmetric key - PSP		KTS (SSH)	ECDSA SigVer (SSH) RSA SigVer (SSH)
Root-CA	JuniperRootCA. Used to verify the validity of the PackagCA	256, 384 - 128, 196	Asymmetric key - PSP			Verify image
Package-CA	Certificate that holds the public key of the signing key that was used to generate all the signatures used on the packages and signatures lists.	256 - 128	Asymmetric key - PSP			Verify image
SSH-DH-PUB (self)	SSH DH public key used for key establishment. P-256, P-384 and P-521	256, 384, 521 - 128, 192, 256	Asymmetric key - PSP	KAS-ECC KeyGen (SSH)		KAS-SSC (SSH)
SSH-DH-PUB (peer)	SSH DH public keys provided by protocol peer device and used with SSH for key establishment. P-256, P-384 and P-521.	2048, 256, 4096, 384, 521 - 112,128, 152, 192, 256	Asymmetric key - PSP			KAS-SSC (SSH)
Auth-CO Pub	SSH CO Authentication Public Key	2048, 256, 4096, 384, 521 - 112,128, 152, 192, 256	Asymmetric key - PSP			ECDSA SigVer (SSH) RSA SigVer (SSH)

Table 19: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
HMAC DRBG V value		RAM:Plaintext	Until updated by HMAC_DRBG_Update()	Reset	
HMAC DRBG Key value		RAM:Plaintext	Until updated by HMAC_DRBG_Update()	Reset	
HMAC DRBG Entropy Input		RAM:Plaintext	Until HMAC_Instantiate_Update() or HMAC_DRBG_Reseed() complete	Reset Explicit zeroize function	
HMAC DRBG Seed		RAM:Plaintext	Until HMAC_Instantiate_Update() or HMAC_DRBG_Reseed() complete	Reset Explicit zeroize function	
SSH-DH-Shared-Secret		RAM:Plaintext	Until SSH session termination	Reset Explicit zeroize function	
SSH-Priv		RAM:Plaintext SSD:Plaintext	Until SSH session termination	Reset Zeroize CLI command Explicit zeroize function	
SSH-DH-priv		RAM:Plaintext	Until SSH session termination	Reset Explicit zeroize function	
SSH-SEKs		RAM:Plaintext	Until SSH session termination	Reset Explicit zeroize function	
CO-PW	Manual CLI entry Entry via SSH Entry via console	SSD:Encrypted RAM:Plaintext	Until authentication session termination	Zeroize CLI command	
User-PW	Manual CLI entry Entry via SSH Entry via console	RAM:Plaintext SSD:Obfuscated	Until authentication session termination	Zeroize CLI command	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
SSH-PUB	Output via SSH Output via console Output as part of KAS	SSD:Plaintext		Zeroize CLI command	
Auth-User Pub	Entry via SSH Entry via console	SSD:Plaintext		Zeroize CLI command	
Root-CA	Pre-loaded	SSD:Plaintext		Zeroize CLI command	
Package-CA	Pre-loaded	SSD:Plaintext		Zeroize CLI command	
SSH-DH-PUB (self)	Output as part of KAS	RAM:Plaintext	Until SSH session termination	Reset Explicit zeroize function	
SSH-DH-PUB (peer)	Entry as part of KAS	RAM:Plaintext	Until SSH session termination	Reset Explicit zeroize function	
Auth-CO Pub	Entry via SSH Entry via console			Zeroize CLI command	

Table 20: SSP Table 2

9.5 Transitions

The following transitions apply to algorithms used by this module:

SHA-1: The SHA-1 hash algorithm will be non-Approved for all cryptographic purposes after December 31, 2030.

10 Self-Tests

On power up or reset, the module performs the pre-operational self-tests and the indicated conditional cryptographic algorithm self-tests described below. All KATs must be completed successfully prior to any other use of cryptography by the module. The algorithms utilized in the pre-operational firmware integrity test must pass their own CASTs prior to the Integrity Test.

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
Firmware Integrity check	ECDSA P-256 with SHA2-256	KAT	SW/FW Integrity	PASS/FAIL console output	ECDSA Verify
Critical functions test	SHA2-256	KAT	Critical Function	PASS/FAIL console output	The module implements a critical function that checks that any file that is executed is registered in a manifest of executable files that comes with the firmware. A pre-operational critical function test is implemented that verifies the integrity of the operational environment is being enforced by having the kernel attempt to run a specific executable file that does not contain a hash in the manifest file. The test is successful if it verifies that the specific file cannot be executed.

Table 21: Pre-Operational Self-Tests

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Entropy Source (start-up)	n/a	APT, RCT	CAST	Console output / output of entropy source	Start-up	On power-up
Entropy Source (continuous)	n/a	APT, RCT	CAST	Console output / output of entropy source	Continuous	On power-up
AES-CBC (A3693) Encrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Encrypt	On power-up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CBC (A3693) Decrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Decrypt	On power-up
AES-CTR (A3693) Encrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Encrypt	On power-up
AES-CTR (A3693) Decrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Decrypt	On power-up
HMAC DRBG (A3693)	SHA2-256	KAT	CAST	PASS/FAIL console output	Instantiate, reseed, and generate.	On power-up
KAS-ECC-SSC Sp800-56Ar3 (A3610)	P-256 (SHA 256) P-384 (SHA 384) P-521 (SHA 512)	KAT	CAST	PASS/FAIL console output	Derivation of the expected shared secret	On power-up
ECDSA SigGen (FIPS186-4) (A3693)	P-256, P-384, P-521	KAT	CAST	PASS/FAIL console output	Sign	On power-up
ECDSA SigVer (FIPS186-4) (A3693)	P-256, P-384, P-521	KAT	CAST	PASS/FAIL console output	Verify	On power-up
HMAC-SHA-1 (A3693)	Key size: 160 bits, = 160	KAT	CAST	PASS/FAIL console output	MAC	On power-up
HMAC-SHA2-256 (A3693)	Key size: 256 bits, = 256	KAT	CAST	PASS/FAIL console output	MAC	On power-up
HMAC-SHA2-512 (A3693)	Key size: 512 bits, = 512	KAT	CAST	PASS/FAIL console output	MAC	On power-up
RSA SigGen (FIPS186-4) (A3693)	RSA 2048 w/ SHA2-256, RSA 4096 w/ SHA2-256	KAT	CAST	PASS/FAIL console output	Sign	On power-up
RSA SigVer (FIPS186-4) (A3693)	RSA 2048 w/ SHA2-256, RSA 4096 w/ SHA2-256	KAT	CAST	PASS/FAIL console output	Verify	On power-up
SHA-1 (A3693)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA2-256 (A3693)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-384 (A3693)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-512 (A3693)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
KDF SSH (A4271)	SHA-1, SHA2-256, SHA2-384	KAT	CAST	PASS/FAIL console output	Key derivation	On power-up
SHA-1 (A3367)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-256 (A3367)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-512 (A3367)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
HMAC-SHA-1 (A3367)	Key size: 160 bits, = 160	KAT	CAST	PASS/FAIL console output	MAC	On power-up
HMAC-SHA2-256 (A3367)	Key size: 256 bits, = 256	KAT	CAST	PASS/FAIL console output	MAC	On power-up
HMAC DRBG (A3493)	SHA2-256	KAT	CAST	PASS/FAIL console output	Health-tests initialise, re-seed, and generate	On power-up
AES-CBC (A3493) Encrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Encrypt	On power-up
AES-CBC (A3493) Decrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Decrypt	On power-up
HMAC-SHA-1 (A3493)	Key size:160 bits, = 160	KAT	CAST	PASS/FAIL console output	MAC	On power-up
HMAC-SHA2-256 (A3493)	Key size:256 bits, = 256	KAT	CAST	PASS/FAIL console output	MAC	On power-up
SHA-1 (A3493)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-256 (A3493)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA2-512 (A3361)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
ECDSA KeyGen (FIPS186-4) (A3693)	P-256, P-384, P-521	PCT	PCT	Returned key/transition soft error state	Generation and Verification of ECDSA signature	On key generation
RSA KeyGen (FIPS186-4) (A3693)	RSA 2048, RSA 4096	PCT	PCT	Returned key/transition soft error state	Generation and Verification of signature	On key generation
FW load	ECDSA P-256 with SHA2-256	KAT	SW/FW Load	PASS/FAIL console output	Verification of ECDSA signature on FW	On FW load
Manual SSP entry	-	Duplicate entry	Manual Entry	PASS/FAIL console output	Duplicate entry	On manual, direct entry of SSP

Table 22: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Firmware Integrity check	KAT	SW/FW Integrity	On demand	Manually
Critical functions test	KAT	Critical Function	On demand	Manually

Table 23: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Entropy Source (start-up)	APT, RCT	CAST	On demand	Manually
Entropy Source (continuous)	APT, RCT	CAST	Continuous	Automatically
AES-CBC (A3693) Encrypt	KAT	CAST	On demand	Manually
AES-CBC (A3693) Decrypt	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CTR (A3693) Encrypt	KAT	CAST	On demand	Manually
AES-CTR (A3693) Decrypt	KAT	CAST	On demand	Manually
HMAC DRBG (A3693)	KAT	CAST	On demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A3610)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-4) (A3693)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-4) (A3693)	KAT	CAST	On demand	Manually
HMAC-SHA-1 (A3693)	KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A3693)	KAT	CAST	On demand	Manually
HMAC-SHA2-512 (A3693)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-4) (A3693)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-4) (A3693)	KAT	CAST	On demand	Manually
SHA-1 (A3693)	KAT	CAST	On demand	Manually
SHA2-256 (A3693)	KAT	CAST	On demand	Manually
SHA2-384 (A3693)	KAT	CAST	On demand	Manually
SHA2-512 (A3693)	KAT	CAST	On demand	Manually
KDF SSH (A4271)	KAT	CAST	On demand	Manually
SHA-1 (A3367)	KAT	CAST	On demand	Manually
SHA2-256 (A3367)	KAT	CAST	On demand	Manually
SHA2-512 (A3367)	KAT	CAST	On demand	Manually
HMAC-SHA-1 (A3367)	KAT	CAST	On power-up	Manually
HMAC-SHA2-256 (A3367)	KAT	CAST	On power-up	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC DRBG (A3493)	KAT	CAST	On power-up	Manually
AES-CBC (A3493) Encrypt	KAT	CAST	On power-up	Manually
AES-CBC (A3493) Decrypt	KAT	CAST	On power-up	Manually
HMAC-SHA-1 (A3493)	KAT	CAST	On power-up	Manually
HMAC-SHA2-256 (A3493)	KAT	CAST	On power-up	Manually
SHA-1 (A3493)	KAT	CAST	On power-up	Manually
SHA2-256 (A3493)	KAT	CAST	On power-up	Manually
SHA2-512 (A3361)	KAT	CAST	On power-up	Manually
ECDSA KeyGen (FIPS186-4) (A3693)	PCT	PCT	On condition trigger	Automatic
RSA KeyGen (FIPS186-4) (A3693)	PCT	PCT	On condition trigger	Automatic
FW load	KAT	SW/FW Load	On FW load request	Automatic
Manual SSP entry	Duplicate entry	Manual Entry	On condition trigger	Automatic

Table 24: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Critical Failure state	The cryptographic module ceases to perform cryptographic operations, inhibits all data output, and provides status of the error via syslog messages and console status output	On pre-operational self-test or CAST failure	Power cycle	Console status output
Soft Error State	A non-critical self-test failure occurs, causing a failure of the triggering operation	PCT, firmware load test, continuous entropy health test failure	The module processes the error, and resumes normal operation	Console displays error

Table 25: Error States

The module enters Critical Failure State upon failure of any pre-operational self-tests or CAST, causing the kernel to 'panic' and all execution to halt. The only way to exit from this state is to reboot the module, which causes the self-tests to be repeated and pass successfully before the corresponding algorithms are usable.

10.5 Operator Initiation of Self-Tests

Self–tests that are performed at power-up are available on demand by power cycling the module.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

Before installation of module firmware, CO must first zeroize any module SSPs by following the instructions in Section 9.3.

Once zeroization is complete, the CO must install the JUNOS firmware image on the device using the following CLI command:

```
CO@host> request system software add /<image-path>/<image-filename>
no-copy no-validate reboot.
```

The image-filenames for the validated firmware are as follows:

- SRX1500: junos-srxentedge-x86-64-22.2R3-S1.9.tgz;
- SRX4100 and SRX4200: junos-srxmr-x86-64-22.2R3-S1.9.tgz;
- SRX4600: junos-srxhe-x86-64-22.2R3-S1.9.tgz; and
- SRX5400, SRX5600 and SRX5800: junos-vmhost-install-srx-x86-64-22.2R3-S1.9.tgz.

Once the image is installed, the CO shall follow the instructions in Section 7.2 to apply the tamper seals to the module.

Next, the CO shall proceed as follows:

1. Enable the approved mode on the device.

   ```
   CO@host> set system fips level 2
   ```

2. Set the root password.

   ```
   user@host# set system root-authentication plain-text-password
   ```

   ```
   New password: <type password here>
   ```

3. Commit and reboot the device.

   ```
   CO@host# commit
   ```

Once the module is rebooted and the integrity and self-tests have run successfully on initial power-on in, the module is operating in the approved mode of operation. The CO must create a backup image of the firmware to ensure it is also a approved mode Junos OS image by issuing the `request system snapshot` command.

The show version command will display the version of the Junos OS on the device so that the CO can confirm it is the FIPS validated version. The CO should also verify the presence of the suffix string ":fips" in the cli prompt, indicating the module is operating in approved mode.

IPsec, High Availability and TLS features are not enabled by default and must not be enabled for FIPS compliant usage of the module. .

11.2 Administrator Guidance

The Cryptographic Officer is the person responsible for enabling, configuring, monitoring, and maintaining the module in approved mode. The Cryptographic Officer securely installs Junos OS on the device, enables the approved 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.

11.3 Non-Administrator Guidance

No specific non-administrator guidance is required to operate the module.

11.4 Design and Rules

The module design corresponds to the security rules below. The term *must* 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. Power up self-tests do not require any operator action.
3. Data output is inhibited during key generation, self-tests, zeroization, and error states.
4. Status information does not contain CSPs or sensitive data that if misused could lead to a compromise of the module.
5. There are no restrictions on which SSPs are zeroized by the zeroization service.
6. The module does not support a maintenance interface or role.
7. The module does not output intermediate key values.
8. The module requires two independent internal actions to be performed prior to outputting plaintext CSPs.
9. The cryptographic officer must determine whether firmware being loaded is a legacy use of the firmware load service.
10. The cryptographic officer must retain control of the module while zeroization is in process.
11. IPsec, High Availability and TLS features must not be enabled.

11.5 Maintenance Requirements

No special maintenance requirements apply.

11.6 End of Life

When disposing of the cryptographic module, the CO shall perform the zeroize command described in Section 9.3.

12 Mitigation of Other Attacks

The module does not implement mechanisms to mitigate other attacks beyond what is described in this security policy.