HPE Juniper Networking

HPE Juniper Networking SRX1600 Service Gateway

FIPS 140-3 Non-Proprietary Security Policy

Version: Junos OS 23.4R1.9

Prepared for:

HPE Juniper Networking,
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 HPE Juniper Networking SRX1600 Service Gateway running Junos OS 23.4R1.9, 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 HPE Juniper Networking SRX1600 Service Gateway is a secure router that provides 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.

This FIPS 140-3 validation includes the SRX Series Service Gateway model SRX1600.

The cryptographic module runs Junos OS, Juniper's reliable, high-performance, modular network operating system that is supported across all of Juniper's physical and virtual routing, switching, and security platforms.

The cryptographic module provides for an encrypted connection, using SSH, between the management station and the module. All other data input or output from the modules are considered plaintext for this FIPS 140-3 validation.

Module Type:

The cryptographic module is a Hardware cryptographic module.

Module Embodiment:
The cryptographic module is defined as a Multi-Chip Standalone module that executes Junos OS 23.4R1.9 firmware on any of the identified HPE Juniper Networking devices.

Cryptographic Boundary:

The cryptographic boundary is defined as the outer edge of the chassis. The chassis is a rigid sheet-metal structure that houses all components of the device.

The cryptographic module is FIPS-compliant when installed and configured with Junos OS 23.4R1.9 validated firmware as specified in section 11.1.

Tested Operational Environment's Physical Perimeter (TOEPP):

The physical form of the module is depicted in Figure 1 to Figure 2 below. The module does not rely on external devices for input and output of security sensitive parameters (SSPs).

Figure 1 - SRX1600 (front)

Figure 2 – SRX1600 (rear)

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

The following models of the module were tested.

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
SRX1600	SRX1600	Junos OS 23.4R1.9	Intel(R) Xeon(R) D-1713NT	16 x 1 GbE 10/100/1000 BASE-T ports; 4 x 1 GbE/10 GbE SFP+ ports; 4x10GbE SFP ports+; 2 x 1 GbE/10 GbE/25 GbE SFP28 ports; 2 x 1 GbE SFP ports

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.

N/A for this module.

Tested Module Identification – Hybrid Disjoint Hardware: N/A

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

N/A for this module.

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

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

N/A for this module.

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

There are no vendor-affirmed operational environments claimed.

N/A for this module.

2.3 Excluded Components

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

2.4 Modes of Operation

Modes List and Description:

The module supports an Approved mode only. The module enters Approved mode as a result of successful installation, initialization and configuration steps described in section 11. Until these procedures have been followed, the module is non-compliant.

Mode Name	Description	Type	Status Indicator
Approved	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

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 1.0.2 Cryptographic Library

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A6736	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CTR	A6736	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
ECDSA KeyGen (FIPS186-5)	A6736	Curve - P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A6736	Curve - P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A6736	Curve - P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A6736	Curve - P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A6736	Key Length - Key Length: 160	FIPS 198-1
HMAC-SHA2-256	A6736	Key Length - Key Length: 256	FIPS 198-1
HMAC-SHA2-512	A6736	Key Length - Key Length: 512	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A6736	Domain Parameter Generation Methods - P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3

Algorithm	CAVP Cert	Properties	Reference
KDF SSH (CVL)	A6736	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA-1, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
RSA KeyGen (FIPS186-5)	A6736	Key Generation Mode - probable Modulo - 2048, 3072, 4096 Primality Tests - 2powSecStr Private Key Format - standard	FIPS 186-5
RSA SigGen (FIPS186-5)	A6736	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5	FIPS 186-5
RSA SigVer (FIPS186-5)	A6736	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5	FIPS 186-5
SHA-1	A6736	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A6736	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A6736	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A6736	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 4: Approved Algorithms - OpenSSL 1.0.2 Cryptographic Library

Intel DRNG

Algorithm	CAVP Cert	Properties	Reference
Conditioning Component AES-CBC-MAC SP800-90B	A2518	Key Length - 128	SP 800-90B

Table 5: Approved Algorithms - Intel DRNG

OpenSSL 1.1.1 Cryptographic Library

Algorithm	CAVP Cert	Properties	Reference
ECDSA SigVer (FIPS186-5)	A5633	Curve - P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
SHA2-256	A5633	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 6: Approved Algorithms - OpenSSL 1.1.1 Cryptographic Library

Kernel Cryptographic Library

Algorithm	CAVP Cert	Properties	Reference
HMAC DRBG	A5108	Prediction Resistance - Yes Mode - SHA2-256	SP 800-90A Rev. 1
HMAC-SHA2-256	A5108	Key Length - Key Length: 256	FIPS 198-1
SHA2-256	A5108	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 7: Approved Algorithms - Kernel Cryptographic Library

LibMD Cryptographic Library

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA-1	A5107	Key Length - Key Length: 160	FIPS 198-1
HMAC-SHA2-256	A5107	Key Length - Key Length: 256	FIPS 198-1
SHA-1	A5107	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A5107	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A5107	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 8: Approved Algorithms - LibMD Cryptographic Library

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Key type:Asymmetric	N/A	SP 800-133 Rev. 2 Section 4, example 1 direct output from DRBG.

Table 9: 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: (A6736) AES-CTR: (A6736)
KAS-SSC (SSH)	KAS-SSC	Key Agreement Scheme Shared Secret Computation for SSH		KAS-ECC-SSC Sp800-56Ar3: (A6736)
ECDSA SigGen (SSH)	DigSig-SigGen	Signature Generation for peer authentication in SSH		ECDSA SigGen (FIPS186-5): (A6736) SHA2-256: (A6736) SHA2-384: (A6736)

Name	Type	Description	Properties	Algorithms
				SHA2-512: (A6736)
ECDSA SigVer (SSH)	DigSig-SigVer	Signature Verification for peer authentication in SSH		ECDSA SigVer (FIPS186-5): (A6736) SHA2-256: (A6736) SHA2-384: (A6736) SHA2-512: (A6736)
MAC (SSH)	MAC	Message Authentication for SSH		HMAC-SHA-1: (A6736) HMAC-SHA2-256: (A6736) HMAC-SHA2-512: (A6736) SHA-1: (A6736) SHA2-256: (A6736) SHA2-512: (A6736)
KDF (SSH)	KAS-135KDF	Key derivation Function for SSH		KDF SSH: (A6736) SHA-1: (A6736) SHA2-256: (A6736) SHA2-384: (A6736) SHA2-512: (A6736)
SHA (LibMD)	SHA	Message Digest Generation		SHA-1: (A5107) SHA2-256: (A5107) SHA2-512: (A5107)

Name	Type	Description	Properties	Algorithms
MAC (LibMD)	MAC	Message authentication		HMAC-SHA-1: (A5107) HMAC-SHA2-256: (A5107)
DRBG (Kernel)	DRBG	Random Bit Generation		HMAC DRBG: (A5108) HMAC-SHA2-256: (A5108) SHA2-256: (A5108)
AES-CBC-MAC (Intel)	ENT-Cond	Entropy source conditioning component		Conditioning Component AES-CBC-MAC SP800-90B: (A2518)
ECDSA KeyGen (PKID)	AsymKeyPair-KeyGen CKG	ECDSA Key Generation used for SSH when authentication keys are internally generated		ECDSA KeyGen (FIPS186-5): (A6736) CKG: () Key type: Asymmetric HMAC DRBG: (A5108)
RSA KeyGen (PKID)	AsymKeyPair-KeyGen CKG	RSA Key generation used for SSH when authentication keys are internally generated		RSA KeyGen (FIPS186-5): (A6736) CKG: () Key type: Asymmetric HMAC DRBG: (A5108)
RSA SigGen (SSH)	DigSig-SigGen	RSA Signature Generation for SSH		RSA SigGen (FIPS186-5): (A6736)

Name	Type	Description	Properties	Algorithms
RSA SigVer (SSH)	DigSig-SigVer	RSA Signature verification for SSH		RSA SigVer (FIPS186-5): (A6736)
Verify image	DigSig-SigVer	Verification of software image		ECDSA SigVer (FIPS186-5): (A5633) SHA2-256: (A5633)
Full KAS (SSH)	KAS-Full	Full Key Agreement for SSH	IG:IG D.F Scenario 2, path 2, split Key Confirmation:No Key derivation:KDA (separately tested) Caveat:Key establishment methodology provides between 128 and 256 bits of security strength	ECDSA KeyGen (FIPS186-5): (A6736) ECDSA KeyVer (FIPS186-5): (A6736) KAS-ECC-SSC Sp800-56Ar3: (A6736) KDF SSH: (A6736) SHA-1: (A6736) SHA2-256: (A6736) SHA2-384: (A6736) SHA2-512: (A6736)
KAS-ECC KeyGen (SSH)	AsymKeyPair-KeyGen CKG	KAS-ECC Key Pair Generation SSH		ECDSA KeyGen (FIPS186-5): (A6736)
ENT	ENT-P	Entropy Source		Conditioning Component AES-CBC-MAC SP800-90B: (A2518)

Name	Type	Description	Properties	Algorithms
KTS (SSH)	KTS-Wrap	Key transport using SSH as per IG D.G provisions	Standard:SP 800-38F IG D.G:Key wrapping key Caveat:Key establishment methodology provides between 128 and 256 bits of encryption strength	AES-CBC: (A6736) AES-CTR: (A6736) HMAC-SHA-1: (A6736) HMAC-SHA2-256: (A6736) HMAC-SHA2-512: (A6736)

Table 10: Security Function Implementations

2.7 Algorithm Specific Information

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

The tables below indicate the entropy source used by the module and their associated certificates.

Cert Number	Vendor Name
E141	HPE Juniper Networking

Table 11: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Junos OS Physical Entropy Source	Physical	Intel(R) Xeon(R) D-1713NT	128 bits	Full entropy	A2518 (AES-CBC-MAC)

Table 12: Entropy Sources

The module accumulates entropy from an entropy source that generates 128 bits of full entropy, using AES-CBC-MAC-128 vetting conditioning component, until 72 bytes of entropy has been collected. The entropy source is used to seed the module's HMAC-SHA2-256-DRBG with 576-bits of full entropy. HMAC-SHA2-256-DRBG is used for all random data required by the module, including key generation. The maximum security strength of keys generated by the module is 256-bits.

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. E141).

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	EC Diffie-Hellman P-256 EC Diffie-Hellman P-384 EC Diffie-Hellman P-521	ECDSA P-256 ECDSA P-384 ECDSA P-521 RSA 2048 RSA 3072 RSA 4096	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. The module does not have a Control Output Interface.

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
Power Button	Control Input	Power on/off
LED	Status Output	Status indicator lighting
Power	Power	Power
USB	Data Input Control Input	Firmware load port

Physical Port	Logical Interface(s)	Data That Passes
SFP28	Data Input Data Output Control Input Status Output	Two 1/10/25 GbE SFP28 MACsec ports for network traffic
SFP+	Data Input Data Output Control Input Status Output	Four 1/10 GbE SFP+ MACsec ports for network traffic

Table 13: Ports and Interfaces

4 Roles, Services, and Authentication

4.1 Authentication Methods

The module implements two forms of role-based authentication methods, as described in the following table.

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), RSA 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(R) Xeon(R) D-1713NT processor (4 cores, 2.2 GHz) allows for the probability of success by brute-force attack: 60 x 4 x 2.2 x 10^9 x 1/(2^112) < 1/100,000.

Table 14: Authentication Methods

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 15: 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 either of the role-based operator authentication methods in Section 4.1.

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,Configuration of the users (including permissions and passwords), ssh (including ciphers, macs, key-exchange, host-key algorithm), syslog, authentication login settings (retries, back-off factor and	Status	SHA (LibMD) MAC (LibMD) DRBG (Kernel) ECDSA KeyGen (PKID) RSA KeyGen	Cryptographic Officer - HMAC DRBG V value: E - HMAC DRBG Key value: E - HMAC DRBG Entropy Input (kernel):

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			threshold) . Configuration carried out by executing CLI commands and providing parameters. Configuration can also be done by providing configuration text files.		(PKID) ENT	E - HMAC DRBG Seed: E - CO-PW: W - User-PW: W - SSH-Priv: G,R,W - Auth-User Pub: W
Configure	Non-security relevant configuration	None	CLI commands, QoS features, VLAN addressing and integrated routing, Configuration of Network Interfaces, Firewall Services	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 (kernel): Z - SSH-DH-

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Shared-Secret: Z - SSH-Priv: Z - SSH-SEKs: Z - SSH-SEKs (Auth): Z - CO-PW: Z - User-PW: Z - SSH-PUB: Z - Auth-User Pub: 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) AES-CBC-MAC	Cryptographic Officer - HMAC DRBG V value: E - HMAC DRBG Key value: E - HMAC DRBG Entropy Input (kernel): E - HMAC DRBG Seed: E - SSH-DH-Shared-Secret: G,E - SSH-DH-

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					(Intel) RSA SigGen (SSH) RSA SigVer (SSH) Full KAS (SSH) KAS-ECC KeyGen (SSH) ENT KTS (SSH)	priv: G,E - SSH-SEKs: G,E - SSH-SEKs (Auth): G,E - Auth-CO Pub: E - SSH-Priv: E - CO-PW: E - SSH-DH-PUB (self): G,R - SSH-DH-PUB (peer): W,E User - HMAC DRBG V value: E - HMAC DRBG Key value: E - HMAC DRBG Entropy Input (kernel): E - HMAC DRBG Seed: E - SSH-Priv: E - User-PW: E - SSH-DH-Shared-Secret: G,E - SSH-DH-priv: G,E - SSH-SEKs: G,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- SSH-SEKs (Auth): G,E - SSH-DH-PUB (self): G,R - SSH-DH-PUB (peer): W,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 (kernel): Z - HMAC DRBG Seed: Z - SSH-DH-Shared-Secret: Z - SSH-DH-priv: Z - SSH-SEKs: Z - SSH-SEKs

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						(Auth): 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 (kernel): Z - HMAC DRBG Seed: Z - SSH-DH-Shared-Secret: Z - SSH-SEKs: Z - SSH-SEKs (Auth): Z - SSH-DH-PUB (self): Z - SSH-DH-PUB (peer): Z
Traffic	Traffic requiring no cryptographic services	None	Traffic in	Traffic out	None	Unauthenticated

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Load Image	Loading of firmware image	':fips' suffix in CLI prompt	CLI command, image location	status	Verify image	Cryptographic Officer - Root-CA: E - Package-CA: E
Perform self-tests	On demand execution 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 16: Approved Services

4.4 Non-Approved Services

The module does not offer any 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 a firmware integrity self-test that uses ECDSA P-256 with SHA2-256 to ensure the integrity of all Junos OS firmware components. The ECDSA P-256 public key used for signature verification is a non-SSP and stored persistently across reboots in the module's Non-Volatile memory and is exempt from zeroization. The self-test is automatically run on power-up.

5.2 Initiate on Demand

The firmware integrity test can 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: Limited

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 # Juniper Networks XXXXXX (barcode))	Once per month by the Cryptographic Officer	Seals should be free of any tamper evidence.
Opaque metal enclosure.	n/a	n/a

Table 17: 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. 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:

Number: 14 seals

Placement:

Front pane: TEL 1-3

- Apply one seal on the left side of the power supply inlet vent holes (left of the front pane -**TEL 1**). Position the seal so that it extends from the top cover approximately one quarter overhang, the overhanging part is sealed onto the non hex vent hole part of the front pane.
- Apply one seal on the middle of the front pane (**TEL 2**). Position the seal so that it extends from the top cover to the top of the front faceplate above the center without blocking the hex vent holes.
- Apply one seal on the right of the front pane (**TEL 3**). Position the seal so that it extends from the top cover to top of the front faceplate above the center of the cluster port and management port, without blocking the hex vent holes.

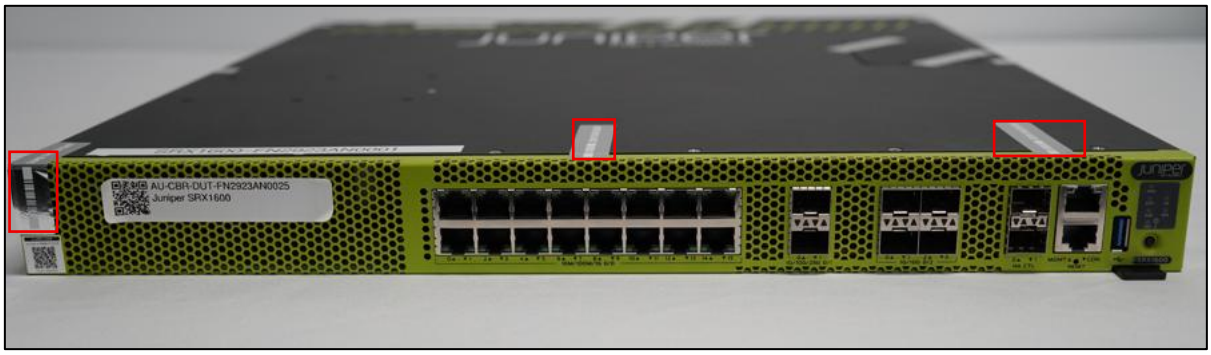

Figure 3 - SRX1600 Front View: TEL 1 – 3

Access door on top cover: TEL 6-7

- Apply two seals on top of the SSD access door on the top cover. Position the seal approximately at 45 degree angle so that two ends of the seal are applied on the top cover and the central portion of the seal is on the access door (**TEL-6**). Place the second seal in a parallel to cover the other diagonal of the access door (**TEL-7**).

Figure 4 - SRX1600 Top - Front View TEL 1-14

Sides of chassis: TEL 4-5 and 13-14

- Apply four seals on the sides of the system. Fold the seals twice so that the resulting three faces can be attached to the top cover, side of the chassis, and bottom of chassis. For each side, the two seals should cover the forth (**TEL 4**) and seventh top cover side screws counting from the front faceplate (**TEL 5**). Repeat the same for the other side (**TEL 13-14**).

Figure 5 - SRX1600 Right Side View: TEL 4 and 5

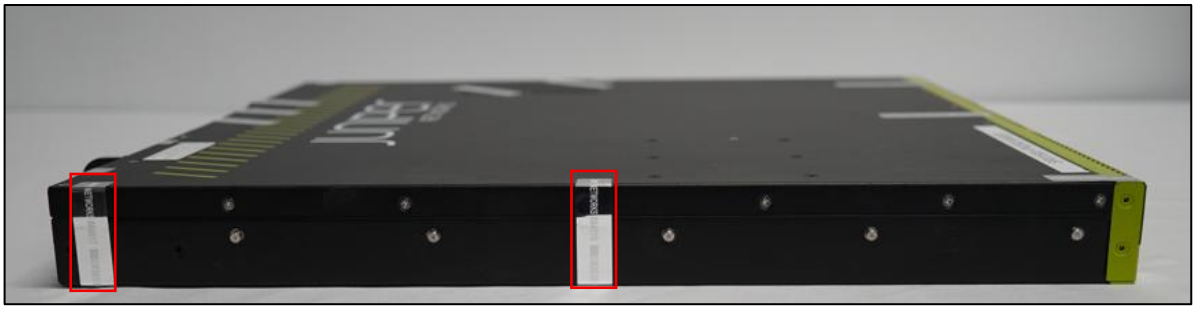

Figure 6 - SRX1600 Left Side View: TEL 13 and 14

Rear pane: TEL 8-12

- Apply a total of five seals on the rear faceplate.
- Fold the seals twice so that the seals so that the resulting three faces can be attached to the top cover, rear face of chassis, and bottom of chassis. Place the first seal (**TEL 12**) along the centerline of the Power Supply Unit (PSU) 1 blank. Place the second seal on the left edge of PSU0 such that it does not block the airflow (**TEL 11**).
- Attach the seal onto Fan0's screw (**TEL 8**) on the rear face and bend it once so that the remaining face is attached to the top cover. Repeat the same step for Fan1 (**TEL 9**) and Fan 2 (**TEL 10**).

Figure 7 - SRX1600 Rear View: TEL 8-12

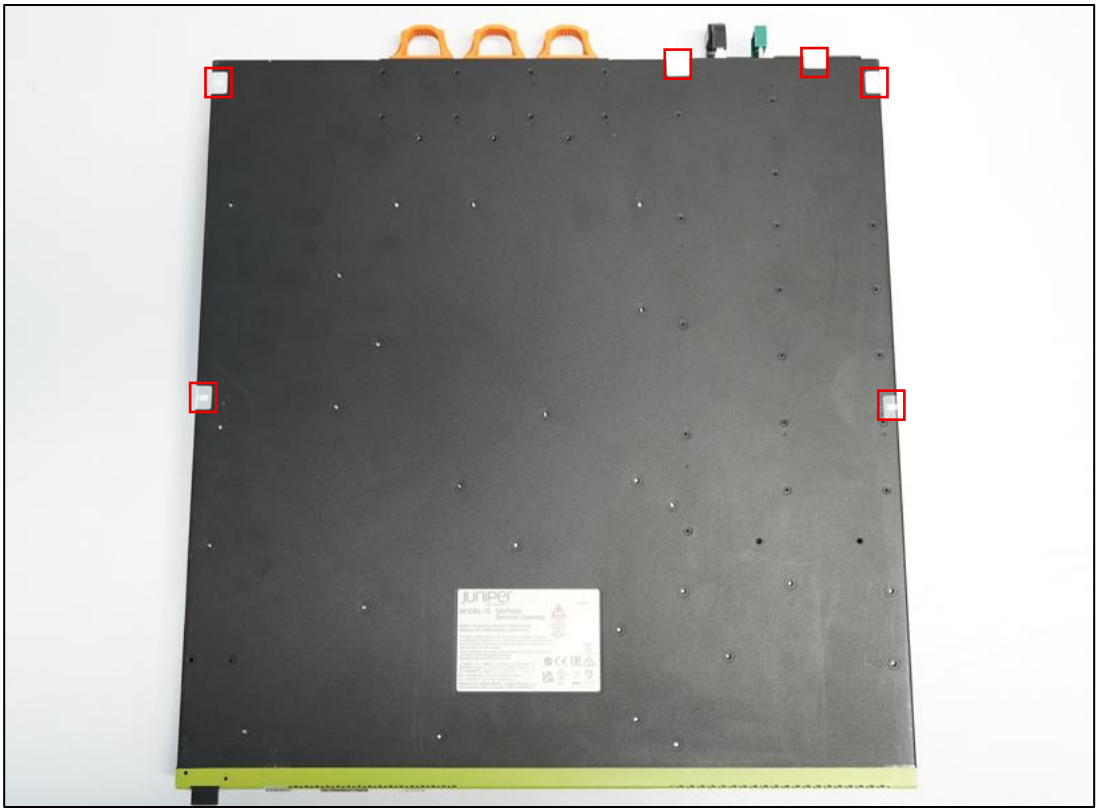

Figure 8 - SRX1600 Bottom View TEL 4-5 and 11-14.

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. Do not cut or otherwise resize a seal to make it fit.
- Make sure all surfaces to which the seals are applied are clean and dry and clear of any residue.
- Apply the seals with firm pressure across the seal to ensure adhesion. Allow at least 24 hours for the adhesive to cure.

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 six-digit part numbers with a barcode in the following format: **Juniper Networks XXXXXX (barcode)**.

8 Non-Invasive Security

This section is not applicable, as there is 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	Static

Table 18: 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	

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Output as part of KAS	RAM	Remote peer	Plaintext	Automated	Electronic	
Pre-loaded	Manufacturer	SSD	Plaintext	Manual	Direct	

Table 19: SSP Input-Output Methods

9.3 SSP Zeroization Methods

The table below describes the SSP zeroization methods employed by the module.

Zeroization Method	Description	Rationale	Operator Initiation
Reset	Zeroization 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. Zeroization is practically instantaneous.	Yes, both User and CO, via invocation of Local Reset or Remote Reset services
Zeroize CLI command	This command wipes 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	Zeroization of SSPs in memory when no longer needed	Use of explicit zeroization function destroys SSP information immediately by overwriting memory area with zeroes	No. The operator cannot directly initiate this method.

Table 20: SSP Zeroization Methods

The Zeroize CLI command method is detailed in section 11.2.3.

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 (kernel)	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	Asymmetric private key - CSP	ECDSA KeyGen (PKID) RSA		ECDSA SigGen (SSH) RSA

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		- 112,128, 152, 192, 256		KeyGen (PKID)		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 encryption/decryption keys used with SSH-2.	128, 192, 256 - 128, 192, 256	Symmetric Key - CSP		KDF (SSH)	Enc/Dec (SSH) KTS (SSH)
SSH-SEKs (Auth)	Session authentication keys used with SSH-2.	160, 256, 512 - 128-512	Auth Key - CSP		KDF (SSH)	Enc/Dec (SSH) KTS (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 - 112,128, 152, 192, 256	Asymmetric key - PSP	ECDSA KeyGen (PKID) RSA KeyGen (PKID)		ECDSA SigVer (SSH) RSA SigVer (SSH)
Auth-User Pub	SSH User Authentication Public Key	2048, 256, 4096, 384, 521	Asymmetric key - PSP		KTS (SSH)	ECDSA SigVer (SSH) RSA

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		- 112,128, 152, 192, 256				SigVer (SSH)
Root-CA	JuniperRootCA. Used to verify the validity of the PackageCA	256 - 128	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.	256, 384, 521 - 128, 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 21: 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 (kernel)		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-PUB:Paired With
SSH-DH-priv		RAM:Plaintext	Until SSH session termination	Reset Explicit zeroize function	SSH-DH-PUB (self):Paired With
SSH-SEKs		RAM:Plaintext	Until SSH session termination	Reset Explicit zeroize function	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
SSH-SEKs (Auth)		RAM:Plaintext	Until SSH session termination	Reset Explicit zeroize function	
CO-PW	Entry via SSH Manual CLI entry Entry via console	SSD:Plaintext RAM:Plaintext	Until authentication session termination	Zeroize CLI command	
User-PW	Entry via SSH Manual CLI entry Entry via console	RAM:Plaintext SSD:Plaintext	Until authentication session termination	Zeroize CLI command	
SSH-PUB	Output via SSH Output via console Output as part of KAS	SSD:Plaintext		Zeroize CLI command	SSH-Priv:Paired With
Auth-User Pub	Entry via SSH Entry via console	SSD:Plaintext		Zeroize CLI command	
Root-CA	Pre-loaded	SSD:Plaintext		Zeroize CLI command	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
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-priv:Paired With
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 22: SSP Table 2

9.5 Transitions

The SHA-1 hash algorithm will be non-Approved for cryptographic protection 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 CASTs for algorithms utilized in the pre-operational Firmware integrity check are performed prior to the Firmware integrity check.

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 23: 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

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CBC (A6736) Encrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Encrypt	On power-up
AES-CBC (A6736) Decrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Decrypt	On power-up
AES-CTR (A6736) Encrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Encrypt	On power-up
AES-CTR (A6736) Decrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Decrypt	On power-up
HMAC DRBG (A5108)	SHA2-256	KAT	CAST	PASS/FAIL console output	Instantiate, reseed, and generate	On power-up
KAS-ECC-SSC Sp800-56Ar3 (A6736)	P-256 (SHA 256)	KAT	CAST	PASS/FAIL console output	Derivation of the expected shared secret	On power-up
ECDSA SigGen (FIPS186-5) (A6736)	P-256, P-384	KAT	CAST	PASS/FAIL console output	Sign	On power-up
ECDSA SigVer (FIPS186-5) (A6736)	P-256, P-384	KAT	CAST	PASS/FAIL console output	Verify	On power-up
HMAC-SHA-1 (A6736)	Key size: 160 bits, = 160	KAT	CAST	PASS/FAIL console output	MAC	On power-up
HMAC-SHA2-256 (A6736)	Key size: 256 bits, = 256	KAT	CAST	PASS/FAIL console output	MAC	On power-up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-512 (A6736)	Key size: 512 bits, = 512	KAT	CAST	PASS/FAIL console output	MAC	On power-up
RSA SigGen (FIPS186-5) (A6736)	RSA 2048 w/ SHA2-256	KAT	CAST	PASS/FAIL console output	Sign	On power-up
RSA SigVer (FIPS186-5) (A6736)	RSA 2048 w/ SHA2-256	KAT	CAST	PASS/FAIL console output	Verify	On power-up
SHA-1 (A6736)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-256 (A6736)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-384 (A6736)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-512 (A6736)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
KDF SSH (A6736)	SHA-1, SHA2-256, SHA2-384	KAT	CAST	PASS/FAIL console output	Key derivation	On power-up
SHA-1 (A5107)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-256 (A5107)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-512 (A5107)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
HMAC-SHA-1 (A5107)	Key size: 160 bits, = 160	KAT	CAST	PASS/FAIL console output	MAC	On power-up
HMAC-SHA2-256 (A5107)	Key size: 256 bits, = 256	KAT	CAST	PASS/FAIL console output	MAC	On power-up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CBC (A5108) Encrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Encrypt	On power-up
AES-CBC (A5108) Decrypt	Key Sizes: 128, 192, 256	KAT	CAST	PASS/FAIL console output	Decrypt	On power-up
HMAC-SHA-1 (A5108)	Key size:160 bits, = 160	KAT	CAST	PASS/FAIL console output	MAC	On power-up
HMAC-SHA2-256 (A5108)	Key size:256 bits, = 256	KAT	CAST	PASS/FAIL console output	MAC	On power-up
SHA-1 (A5108)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-256 (A5108)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
SHA2-512 (A5108)	n/a	KAT	CAST	PASS/FAIL console output	Hash	On power-up
ECDSA KeyGen (FIPS186-5) (A6736)	P-256, P-384, P-521	PCT	PCT	Returned key/transition critical error state	Generation and Verification of ECDSA signature	On key generation
RSA KeyGen (FIPS186-5) (A6736)	RSA 2048, RSA 4096	PCT	PCT	Returned key/transition critical 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

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Manual SSP entry	-	Duplicate entry	Manual Entry	PASS/FAIL console output	Duplicate entry	On manual, direct entry of SSP
SHA2-256 (A5633)	N/A	KAT	CAST	PASS/FAIL console output	Hash	On power-up
ECDSA KeyVer (FIPS186-5) (A6736)	P-256, P-384, P-521	PCT	PCT	Returned key/transition critical error state	Generation and Verification of ECDSA signature	On key generation

Table 24: Conditional Self-Tests

10.3 Periodic Self-Test Information

The module does not implement periodic self-testing.

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 25: 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	On power-up	Manually
AES-CBC (A6736) Encrypt	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CBC (A6736) Decrypt	KAT	CAST	On demand	Manually
AES-CTR (A6736) Encrypt	KAT	CAST	On demand	Manually
AES-CTR (A6736) Decrypt	KAT	CAST	On demand	Manually
HMAC DRBG (A5108)	KAT	CAST	On demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A6736)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6736)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6736)	KAT	CAST	On demand	Manually
HMAC-SHA-1 (A6736)	KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A6736)	KAT	CAST	On demand	Manually
HMAC-SHA2-512 (A6736)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6736)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6736)	KAT	CAST	On demand	Manually
SHA-1 (A6736)	KAT	CAST	On demand	Manually
SHA2-256 (A6736)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
SHA2-384 (A6736)	KAT	CAST	On demand	Manually
SHA2-512 (A6736)	KAT	CAST	On demand	Manually
KDF SSH (A6736)	KAT	CAST	On demand	Manually
SHA-1 (A5107)	KAT	CAST	On demand	Manually
SHA2-256 (A5107)	KAT	CAST	On demand	Manually
SHA2-512 (A5107)	KAT	CAST	On demand	Manually
HMAC-SHA-1 (A5107)	KAT	CAST	On power-up	Manually
HMAC-SHA2-256 (A5107)	KAT	CAST	On power-up	Manually
AES-CBC (A5108) Encrypt	KAT	CAST	On power-up	Manually
AES-CBC (A5108) Decrypt	KAT	CAST	On power-up	Manually
HMAC-SHA-1 (A5108)	KAT	CAST	On power-up	Manually
HMAC-SHA2-256 (A5108)	KAT	CAST	On power-up	Manually
SHA-1 (A5108)	KAT	CAST	On power-up	Manually
SHA2-256 (A5108)	KAT	CAST	On power-up	Manually
SHA2-512 (A5108)	KAT	CAST	On power-up	Manually
ECDSA KeyGen (FIPS186-5) (A6736)	PCT	PCT	On condition trigger	Automatic

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA KeyGen (FIPS186-5) (A6736)	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
SHA2-256 (A5633)	KAT	CAST	On power-up	Manually
ECDSA KeyVer (FIPS186-5) (A6736)	PCT	PCT	On condition trigger	Automatic

Table 26: 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 self-test error, PCT	Power cycle	Console status output
Soft Error State	A non-critical self-test failure occurs, causing a failure of the triggering operation	Firmware load test, continuous entropy health test failure	The module processes the error, and resumes normal operation	Console displays error

Table 27: Error States

The module enters error state upon failure of any self-tests, 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

The module must be correctly installed and configured to enter a FIPS compliant state and operate in the Approved mode. The required procedures are as follows:

1. Install the Junos OS firmware image - the procedure is detailed in section 11.2.1
2. Configure device for the Approved mode - the procedure is section 11.2.2.

To continue using the module in a FIPS compliant way, the Module Operation Rules in section 11.4.2 must be followed.

11.2 Administrator Guidance

11.2.1 Installing the Junos OS firmware image

1. 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.

   The cryptographic module device uses the following firmware image

   SRX1600: junos-vmhost-install-srxmr2-x86-64-23.4R1.9.tgz

2. Connect to the console port on the device from your management device and log in to the Junos OS CLI.
3. Install the new package on the device (package may be a local file copied to the device, or a file on a remote server):

   SRX1600: `user@host> request vmhost software add <package>`
4. Reboot the device to load the installation:

   SRX1600: `user@host> request vmhost reboot`

5. After the reboot has completed, log in and use the show version command to verify that the new version of the software is successfully installed.

SRX1600:
```
user@host> show version
Hostname: <host name>
Model: srx1600
Junos: 23.4R1.9
```

11.2.2 Configure the device for the Approved mode

To configure the device for the Approved mode:

1. Zeroize the device to delete all CSPs before entering the Approved mode.

   SRX1600: `root@host# request vmhost zeroize no-forwarding`
2. After the device comes up, login using username “root” and password blank.
3. Configure root authentication with password at least 10 characters or more.

   SRX1600: `root@host# set vmhost root-authentication plain-text-password`
4. Load configuration onto device and commit new configuration.

   NOTE: SSH key-exchange configuration must not include ‘dh-group14-sha1’. It is not approved for this module.
5. Configure crypto-officer and login with crypto-officer credentials.
6. Set the fips level to 2.

   SRX1600: `crypto-officer@host# set system fips chassis level 2`
7. Commit and reboot the device.

   SRX1600:
   ```
   crypto-officer@host# commit
   crypto-officer@host# run request system reboot
   ```

11.2.3 Zeroizing the System

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

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).

NOTE: The Crypto-Officer must retain control of the module while zeroization is in progress.

To zeroize the device:

1. Login to the device as Crypto Officer and from CLI, enter

   SRX1600: `crypto-officer@host# request vmhost zeroize no-forwarding`

```
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
```

11.3 Non-Administrator Guidance

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

11.4 Design and Rules

11.4.1 Module Design Rules

The module design implements the following security rules:

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.

11.4.2 Module Operation Rules

The following are requirements for compliant usage of the module:

1. The cryptographic officer must retain control of the module while zeroization is in process.
2. The cryptographic officer shall verify that the firmware image to be loaded on the module is a FIPS validated image.
3. Before pushing the factory reset button on the device, the cryptographic officer shall perform the zeroize command as described in section 11.2.3.
4. The password minimum-length must be configured to be at least 10.
5. Virtual Chassis features must not be configured.
6. The module shall not be configured to use a radius server and the radius server capability shall be disabled.

7. SSH key-exchange must not be configured to include ‘dh-group14-sha1’.

11.6 End of Life

When disposing of the cryptographic module, the cryptographic officer shall perform the zeroize command as described in Section 11.2.3.

12 Mitigation of Other Attacks

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