Juniper Networks EX4600, QFX5100 and QFX5200 Ethernet Switches with JUNOS 18.1R1

Non-Proprietary FIPS 140-2 Cryptographic Module Security Policy

Version 1.0

12 July 2019

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA

408.745.2000
1.888 JUNIPER
www.juniper.net

Table of Contents

List of Figures

List of Figures

1 Introduction

The Juniper Networks QFX series switches are high performance, high density data center switches. The QFX switches provide high performance, wire speed switching with low latency and jitter. The QFX series switches provide the universal building blocks for multiple data center fabric architectures.

This Security Policy covers the following Ethernet switch models:

- QFX5100
- EX4600
- QFX5200

This is a non-proprietary Cryptographic Module Security Policy for the Juniper Networks EX4600, QFX5100 and QFX5200 Ethernet Switches Cryptographic Modules from Juniper Networks. It provides detailed information relating to each of the FIPS 140-2 security requirements relevant to Juniper Networks EX4600, QFX5100 and QFX5200 Ethernet Switches Cryptographic Modules along with instructions on how to run the modules in a secure FIPS 140-2 mode.

All three models run Junos OS firmware. The validated version of firmware is Junos OS 18.1 R1; the image for the hardware platforms is:

- jinstall-host-qfx-5e-x86-64-18.1R1.9-secure-signed.tgz

The Juniper Networks EX4600, QFX5100 and QFX5200 Ethernet Switches are cryptographic modules defined as multiple-chip standalone modules that execute JUNOS 18.1 R1 firmware on the EX4600, QFX5100 and QFX5200 Ethernet Switches listed in Table 1. The cryptographic boundary is defined as the outer edge of the switch. The cryptographic modules' operational environment is a limited operational environment.

Table 1 provides a list of the hardware versions that are part of the module validation and the basic configuration of the hardware.

Table 1 – Cryptographic Module Configurations

Model	Hardware Versions	Network Ports
QFX5100	QFX5100-24Q-AFO/AFI QFX5100-24Q-DC-AFO/AFI	24x40GE QSFP+ ports
	QFX5100-48S-AFO/AFI QFX5100-48S-DC-AFO/AFI	48x10GE SFP+ ports 6x40GE QSFP+ ports
	QFX5100-48SH-AFO/AFI QFX5100-48SH-DC-AFO/AFI	48x10GE SFP+ ports 6x40GE QSFP+ ports

Model	Hardware Versions	Network Ports
	QFX5100-48T-AFO/AFI QFX5100-48T-DC-AFO/AFI	48x10GBASE-T ports 6xQSFP+ ports
	QFX5100-48TH-AFO/AFI QFX5100-48TH-DC-AFO/AFI	48x10GBASE-T ports 6xQSFP+ ports
	QFX5100-96S-AFO/AFI QFX5100-96S-DC-AFO/AFI	96x10GE SFP+ ports 8x40GE QSFP+ ports
EX4600	EX4600-40F-AFO/AFI EX4600-40F-DC-AFO/AFI	24x10GE SFP/SFP+ ports 4x40GE QSFP+ ports
QFX5200	QFX5200-32C-AFO/AFI QFX5200-32C-DC-AFO/AFI	32x100GE QSFP28 ports
	QFX5200-48Y-AFO/AFI QFX5200-48Y-DC-AFO/AFI	48x25GE SFP28 ports 6x100GE QSFP28 ports

The modules are designed to meet FIPS 140-2 Level 1 overall:

Table 2 - Security Level of Security Requirements

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

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

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

1.1 Hardware and Physical Cryptographic Boundary

The physical forms of the module's various models and supported uplink modules are depicted in Figure 1 to Figure 7 below. The module is completely enclosed in a rectangular nickel or clear zinc coated, cold rolled steel, plated steel and brushed aluminium enclosure. For all models, the cryptographic boundary is defined as the outer edge of the switch chassis. The modules do not rely on external devices for input and output.

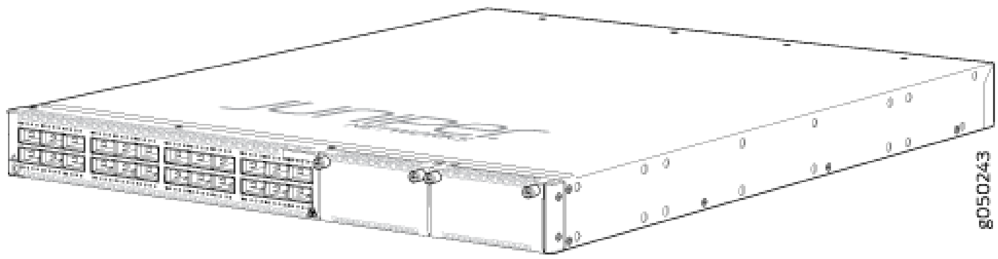

Figure 1 QFX5100-24Q

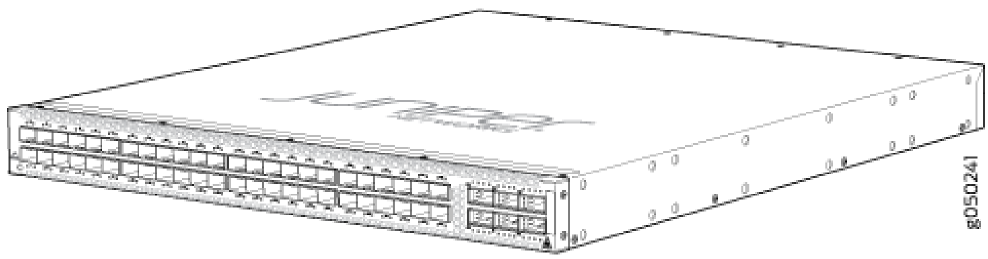

Figure 2 QFX5100-48S and QFX5100-48SH

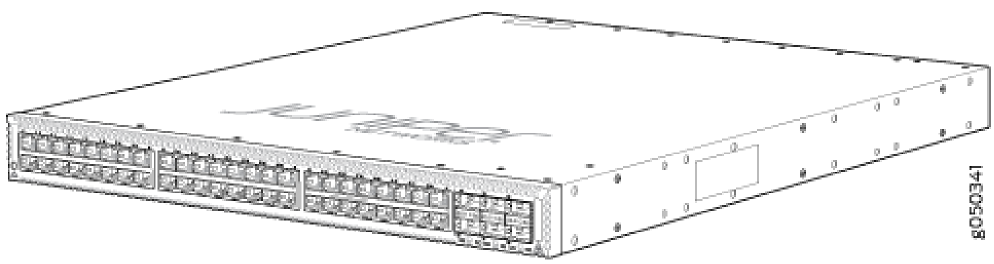

Figure 3 QFX5100-48T and QFX5100-TH

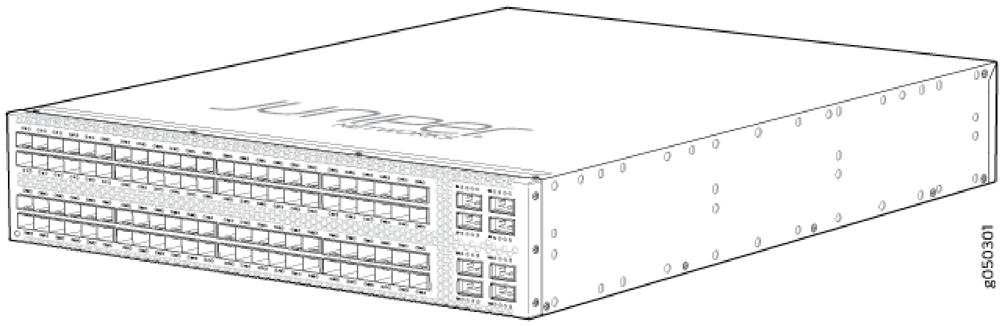

Figure 4 QFX5100-96S

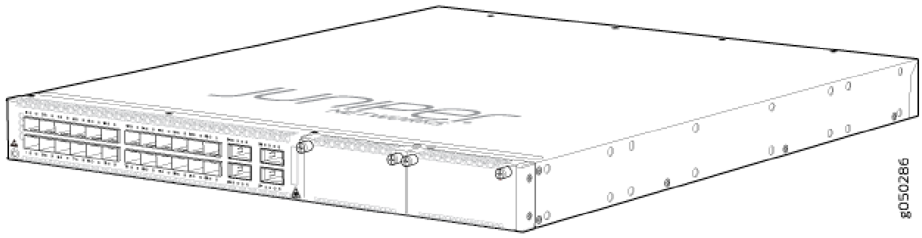

Figure 5 EX4600-40F

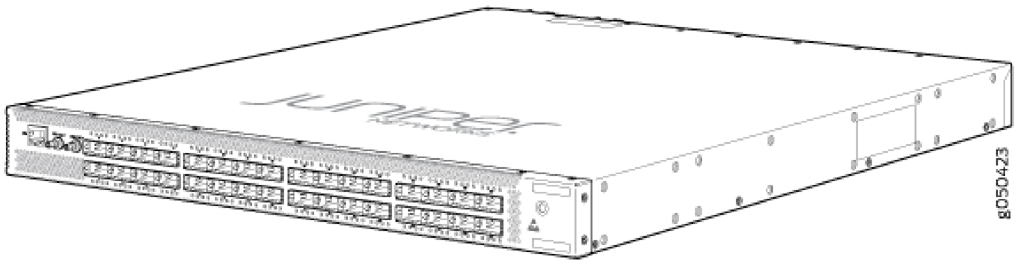

Figure 6 QFX5200-32C

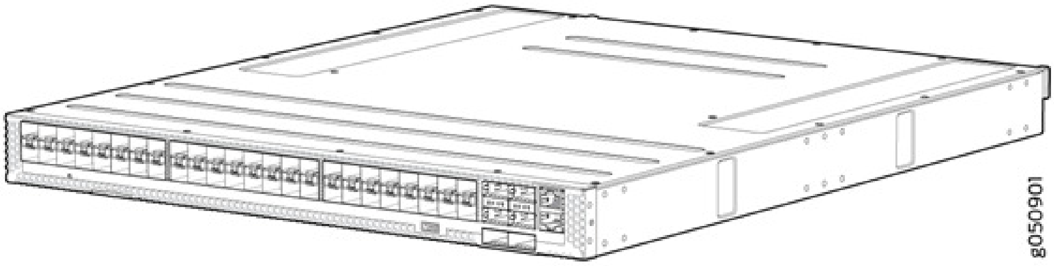

Figure 7 QFX5200-48y

The following table maps each logical interface type defined in the FIPS 140-2 standard to one or more physical interfaces.

Table 3 - Ports and Interfaces

Port	Description	Logical Interface Type
Ethernet	LAN Communications	Control in, Data in, Data out, Status out
Serial	Console serial port	Control in, Status out
MGMT	Out-of-band management port	Control in, Data in, Data out, Status out
Power	Power connector	Power in
Reset	Reset button	Control in
LED	Status indicator lighting	Status out
USB	Firmware load port	Control in, Data in

The following table provides a detailed description of the ports and interfaces available for each model.

Table 4 - Ports and Interfaces

Router model	Power supply port	Fan modules	Console port	Management port	USB port	Built-In Ports	Pluggable
EX4600-40F-AFO/AFI	2	5	1	2	1	32	2 expansion slots
EX4600-40F-DC-AFO/AFI	2	5	1	2	1	32	2 expansion slots
QFX5100-24Q-AFO/AFI	2	5	1	2	1	24	2 expansion slots
QFX5100-24Q-DC-AFO/AFI	2	5	1	2	1	24	2 expansion slots
QFX5100-48S-AFO/AFI	2	5	1	2	1	54	0
QFX5100-48S-DC-AFO/AFI	2	5	1	2	1	54	0
QFX5100-48SH-AFO/AFI	2	5	1	2	1	54	0
QFX5100-48SH-DC-AFO/AFI	2	5	1	2	1	54	0
QFX5100-48T-AFO/AFI	2	5	1	2	1	54	0
QFX5100-48T-DC-AFO/AFI	2	5	1	2	1	54	0
QFX5100-48TH-AFO/AFI	2	5	1	2	1	54	0
QFX5100-48TH-DC-AFO/AFI	2	5	1	2	1	54	0
QFX5200-32C	2	5	1	2	1	32	0
QFX5200-48Y	2	5	1	2	1	48	0

1.2 Mode of Operation

The cryptographic module provides a non-Approved mode of operation in which non-Approved cryptographic algorithms are supported. The module supports non-Approved algorithms when operating in the non-Approved mode of operation as described in Sections 2.4 and 3.4. When transitioning between the non-Approved mode of operation and the Approved mode of operation, the CO must zeroize all CSPs by following the instructions in Section 1.3.

Then, the CO must run the following commands to configure the module into the Approved mode of operation:

```
co@fips-qfx# set system fips level 1
```

```
co@fips-qfx# commit
```

Once the JUNOS firmware image is installed on the device, and configured into Approved mode and rebooted, and integrity and self-tests have run successfully on initial power-on, the module is operating in the Approved mode. This prevents access to non FIPS approved functionality. Transitioning back to non-approved mode is only possible via zeroising the module.

The operator can verify the module is operating in the Approved mode by verifying the following:

- The "show version local" command indicates that the module is running the Approved firmware (i.e. JUNOS Software Release 18.1R1).
- The command prompt ends in ":fips", which indicates the module has been configured in the Approved mode of operation.

1.3 Zeroization

The following command allows the Cryptographic Officer to zeroize CSPs contained within the module:

```
co@fips-qfx> request system zeroize
```

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

2 Cryptographic Functionality

The module implements the FIPS Approved, vendor affirmed, and non-Approved-but-Allowed cryptographic functions listed in Table 5 through Table 8 below. Table 9 summarizes the high level protocol algorithm support.

2.1 Approved Algorithms

References to standards are given in square bracket [ ]; see the References table.

Items enclosed in curly brackets { } are CAVP tested but not used by the module in the Approved mode.

Table 5 – Kernel Cryptographic Functions

CAVP Cert.	Algorithm	Mode	Description	Functions
#5388 #5518	{AES [197]}	CBC [38A]	Key Sizes: 128, 192, 256	Encrypt, Decrypt
#3569 #3674	HMAC [198]	SHA-1	λ = 160	Message Authentication
		SHA-256	λ = 256	
#4322 #4429	SHS [180]	SHA-1 SHA-256 SHA-384 SHA-512		Message Digest Generation
#2715 #2780	{Triple-DES [67]}[1]	TCBC [38A]	Key Size: 192	Encrypt, Decrypt
#2086 #2182	DRBG [90A]	HMAC	SHA-256	Random Bit Generation

[1] The module enforces a limit of 2^{20} transforms per Triple-DES key.

Table 6 – OpenSSL Approved Cryptographic Functions

CAVP Cert.	Algorithm	Mode	Description	Functions
#5389 #5520	AES [197][2]	CBC [38A]	Key Sizes: 128, 192, 256	Encrypt, Decrypt
		CTR [38A]	Key Sizes: 128, 192, 256	Encrypt, Decrypt
N/A[3]	CKG	[133] Section 6.1 [133] Section 6.2		Asymmetric key generation using unmodified DRBG output
#1424	ECDSA [186]		P-256 (SHA-256) P-384 (SHA-384) P-521 (SHA-512)	KeyGen, SigGen, SigVer
#1484			P-256 (SHA-256) P-384 (SHA-384) P-521 (SHA-512)	KeyGen, SigGen
#3571 #3677	HMAC [198]	SHA-1	λ = 160	SSH Message Authentication DRBG Primitive
		SHA-256	λ = 256	
		{SHA-224}	λ = 224	
		SHA-384	λ = 384	
		SHA-512	λ = 512	
N/A	KTS	AES Certs. #5389 and #5520 and HMAC Certs. #3571 and #3677		Key establishment methodology provides between 128 and 256 bits of encryption strength
		Triple-DES Certs. #2716 and #2782 and HMAC Certs. #3571 and #3677		Key establishment methodology provides 112 bits of encryption strength
#2882 #2961	RSA [186]		n=2048 (SHA 256, 384, 512) n=3072 (SHA 256, 384, 512)	SigGen

[2] ECB and GCM modes are also included in the scope of certificates #5389 and 5520 but are not available to users of the module.

[3] Vendor Affirmed.

			n=2048 (SHA 256, 384, 512) n=3072 (SHA 256, 384, 512)	SigVer
			n=2048 n=3072	KeyGen
#4324 #4432	SHS [180]	SHA-1 {SHA224} SHA-256 {SHA-384} SHA-512		Message Digest Generation, SSH KDF Primitive
#2716 #2782	Triple-DES[4] [67]	TCBC [38A]	Key Size: 192	Encrypt, Decrypt
#2087 #2184	DRBG [90A]	HMAC	SHA {1}, 256, {384},{512}	Random Bit Generation
#1852 #1965	CVL	SSH [135]	SHA 1, 256, 384,512	Key Derivation

Table 7 – LibMD Approved Cryptographic Functions

Cert	Algorithm	Mode	Description	Functions
#4323 #4430	SHS [180]	SHA-1 SHA-256 SHA-512		Message Digest Generation
#3570 #3675	HMAC [198]	SHA-1	$\lambda = 160$	Message Authentication
		SHA-256	$\lambda = 256$	Message Authentication

[4] The module enforces a limit of 2^{20} transforms per Triple-DES key.

2.2 Allowed Algorithms

Table 8 - Allowed Cryptographic Functions

Algorithm	Caveat	Use
Diffie-Hellman [IG] D.8	Provides 112 bits of encryption strength.	Key agreement; key establishment
Elliptic Curve Diffie-Hellman [IG] D.8	Provides 128 or 192 bits of encryption strength.	Key agreement; key establishment
NDRNG [IG] 7.14 Scenario 1a	Provides 256 bits of entropy.	Seeding the DRBG

2.3 Allowed Protocols

Table 9 - Protocols Allowed in FIPS Mode

Protocol	Key Exchange	Auth	Cipher	Integrity
SSHv2	Diffie-Hellman (L = 2048, N = 2047) EC Diffie-Hellman P-256, P-384	ECDSA P-256 ECDSA P-384 ECDSA P-521 RSA 2048 RSA 3072	Triple-DES CBC[5] AES CBC 128/192/256 AES CTR 128/192/256	HMAC-SHA-1 HMAC-SHA-256 HMAC-SHA-512

No parts of the SSHv2 protocol, other than the KDF, have been tested or reviewed by the CAVP or CMVP.

The SSH protocol allows independent selection of key exchange, authentication, cipher and integrity. In Table 9 - Protocols Allowed in FIPS Mode above, each column of options for a given protocol is independent, and may be used in any viable combination. These security functions are also available in the SSH connect (non-compliant) service.

2.4 Disallowed Algorithms

These algorithms are non-Approved algorithms that are disabled when the module is operated in an Approved mode of operation.

- ARCFOUR;

[5] The Triple-DES key for the IETF SSHv2 protocol is generated according to RFCs 4253 and 4344.

- Blowfish;
- CAST;
- DSA (SigGen, SigVer; non-compliant);
- HMAC-MD5;
- HMAC-RIPEMD160; and
- UMAC.

2.5 Critical Security Parameters

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

Table 10 - Critical Security Parameters (CSPs)

Name	Description and usage	Length	CKG
DRBG_Seed	Seed material used to seed or reseed the DRBG	N/A	N/A
DRBG_State	V and Key values for the HMAC_DRBG	N/A	N/A
Entropy Input	Entropy input string for the HMAC_DRBG	N/A	N/A
SSH PHK	SSH Private host key. 1st time SSH is configured, the keys are generated. ECDSA P-256 by default, but also supports ECDSA P-384, ECDSA P-521, RSA 2048 and RSA 3072. Used to identify the host.	Key length is dependent on chosen algorithm (see Table 5, Table 6 and/or Table 7).	[133] Section 6.1
SSH DH	SSH Diffie-Hellman private component. Ephemeral Diffie-Hellman private key used in SSH. Diffie-Hellman (N = 2047[6]), EC Diffie-Hellman P-256, EC Diffie-Hellman P-384 or EC Diffie-Hellman P-521	Key length is dependent on chosen algorithm (see Table 5, Table 6 and/or Table 7).	[133] Section 6.2
SSH-SEK	SSH Session Key; Session keys used with SSH. Triple-DES (3key), AES, HMAC.	Key length is dependent on chosen algorithm (see Table 5, Table 6 and/or Table 7).	[133] Section 7.3

[6] SSH generates a Diffie-Hellman private key that is 2x the bit length of the longest symmetric or MAC key negotiated.

Name	Description and usage	Length	CKG
CO-PW	ASCII Text used to authenticate the CO.	N/A	N/A
User-PW	ASCII Text used to authenticate the User.	N/A	N/A

Table 11 – Public keys

Name	Description and usage	CKG
SSH-PUB	SSH Public Host Key used to identify the host. ECDSA P-256 by default, but also supports ECDSA P-384, ECDSA P-521, RSA 2048 and RSA 3072	[133] Section 6.1
SSH-DH-PUB	Diffie-Hellman public component. Ephemeral Diffie-Hellman public key used in SSH key establishment. DH (L = 2048 bit), EC Diffie-Hellman P-256, EC Diffie-Hellman P-384 or EC Diffie-Hellman P-521	[133] Section 6.2
Auth-UPub	SSH User Authentication Public Keys. Used to authenticate users to the module. ECDSA P-256, ECDSA P-384, ECDSA P-521, RSA 2048 or RSA 3072	N/A
Auth-COPub	SSH CO Authentication Public Keys. Used to authenticate CO to the module. ECDSA P-256, ECDSA P-384, ECDSA P-521, RSA 2048 or RSA 3072	N/A
Root CA	Juniper Root CA. ECDSA P-256 or P-384 X.509 Certificate; Used to verify the validity of the Juniper Package CA at software load.	N/A
Package CA	Package CA. ECDSA P-256 X.509 Certificate; Used to verify the validity of Juniper Images at software load and boot.	N/A

3 Roles, Authentication and Services

3.1 Roles and Authentication of Operators to Roles

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

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

The User role monitors the switch via the console or SSH. The user role may not change the configuration.

3.2 Authentication Methods

The module implements two forms of Identity-based authentication - username and password over the Console and SSH as well as username and public key over SSH.

Password authentication: The module enforces 10-character passwords (at minimum) chosen from the 96 human readable ASCII characters. The maximum password length is 20 characters.

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

This leads to a maximum of nine (9) possible attempts in a one-minute period for each getty. The best approach for the attacker would be to disconnect after 4 failed attempts and wait for a new getty to be spawned. This would allow the attacker to perform roughly 9.6 attempts per minute; this would be rounded down to 9 per minute, because there is no such thing as 0.6 attempts. Thus the probability of a successful random attempt is $1/96^{10}$, which is less than 1/1 million. The probability of a success with multiple consecutive attempts in a one-minute period is $9/(96^{10})$, which is less than 1/100,000.

ECDSA signature verification: SSH public-key authentication. Processing constraints allow for a maximum of 5.6e7 ECDSA attempts per minute. The module supports ECDSA (P-256, P-384, and P-521), which has a minimum equivalent computational resistance to attack of either 2^{128} depending on the curve. The probability of a successful random attempt is $1/(2^{128})$, which is less than 1/1,000,000. Processing speed (partial establishment of an SSH session) limits the number of failed authentication attempts in a one minute period to 5.6e7 attempts. The probability of a success with multiple consecutive attempts in a one-minute period is $5.6e7/(2^{128})$, which is less than 1/100,000

3.3 Services

All services implemented by the module are listed in the tables below. Table 14 - CSP Access Rights within Services lists the access to CSPs by each service.

Table 12 - Authenticated Services

Service	Description	CO	User
Configure security	Security relevant configuration	x	
Configure	Non-security relevant configuration	x	
Status	Show status	x	x
Zeroize	Destroy all CSPs	x	
SSH connect	Initiate SSH connection for SSH monitoring and control (CLI)	x	x
Console access	Console monitoring and control (CLI)	x	x
Remote reset	Software initiated reset	x	
Software load	Firmware update	x	

Table 13 – Unauthenticated Traffic

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

Table 14 - CSP Access Rights within Services

Service	CSPs							
	DRBG_Seed	DRBG_State	Entropy Input	SSH PHK	SSH DH	SSH-SEK	CO-PW	User-PW
Configure security	--[7]	E	--	GWR	--	--	W	W
Configure	--	--	--	--	--	--	--	--
Status	--	--	--	--	--	--	--	--
Zeroize	--	Z	--	Z	--	--	Z	Z
SSH connect	--	E	--	E	GE	GE	E	E
Console access	--	--	--	--	--	--	E	E
Remote reset	GZE	GZ	GZE	--	Z	Z	Z	Z
Local reset	GZE	GZ	GZE	--	Z	Z	Z	Z
Traffic	--	--	--	--	--	--	--	--
Software load	--	--	--	--	--	--	--	--

[7] G = Generate: The module generates the key.

R = Read: The key is read from the module (e.g. the key is output).

E = Execute: The module executes using the key.

W = Write: The key is written to persistent storage in the module.

Z = Zeroize: The module zeroizes the key.

Table 15: Public Key Access Rights within Services

Service	Public key					
	SSH-PUB	SSH-DH-PUB	Auth-UPub	Auth-COPub	Root-CA	Package-CA
Configure security	GWR[8]	--	W	W	--	--
Configure	--	--	--	--	--	--
Secure traffic	--	--	--	--	--	--
Status	--	--	--	--	--	--
Zeroize	Z	--	Z	Z	--	--
SSH connect	E	GE	E	E	--	--
Console access	--	--	--	--	--	--
Remote reset	--	Z	Z	Z	--	E
Local reset	--	Z	Z	Z	--	E
Traffic	--	--	--	--	--	--
Software load	--	--	--	--	EW	EW

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

3.4 Non-Approved Services

The following services are available in the non-Approved mode of operation. The security functions provided by the non-Approved services are identical to the Approved counterparts with the exception of SSH Connect (non-compliant).

SSH Connect (non-compliant) supports the security functions identified in Section Disallowed Algorithms and Table 9

Table 16 - Authenticated Services

Service	Description	CO	User
Configure security (non-compliant)	Security relevant configuration	x	
Configure (non-compliant)	Non-security relevant configuration	x	
Status (non-compliant)	Show status	x	x
Zeroize (non-compliant)	Destroy all CSPs	x	
SSH connect (non-compliant)	Initiate SSH connection for SSH monitoring and control (CLI)	x	x
Console access (non-compliant)	Console monitoring and control (CLI)	x	x
Remote reset (non-compliant)	Software initiated reset	x	

Table 17 - Unauthenticated traffic

Service	Description
Local reset (non-compliant)	Hardware reset or power cycle
Traffic (non-compliant)	Traffic requiring no cryptographic services

4 Self-Tests

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

On power up or reset, the module performs the self-tests described below (reset can be forced with the "*request system reboot*" command). All KATs must be completed successfully prior to any other use of cryptography by the module. If one of the KATs fails, the module enters the Critical Failure error state.

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

- Firmware Integrity check using ECDSA P-256 with SHA-256
- Kernel KATs
  - AES-CBC (128/192/256) Encrypt KAT
  - AES-CBC (128/192/256) Decrypt KAT
  - Triple-DES-CBC Encrypt KAT
  - Triple-DES-CBC Decrypt KAT
  - HMAC-SHA-1 KAT
  - HMAC-SHA-256 KAT
  - SHA-384 KAT
  - SHA-512 KAT
  - SP 800-90A HMAC DRBG KAT
    - Health-tests initialize, re-seed, and generate.
- OpenSSL KATs
  - RSA 2048 w/ SHA-256 Sign KAT
  - RSA 2048 w/ SHA-256 Verify KAT
  - ECDSA P-256 w/ SHA-256 Sign/Verify PCT
  - Triple-DES-CBC Encrypt KAT
  - Triple-DES-CBC Decrypt KAT
  - HMAC-SHA-1 KAT

- HMAC-SHA-224 KAT
- HMAC-SHA-256 KAT
- HMAC-SHA-384 KAT
- HMAC-SHA-512 KAT
- AES-CBC (128/192/256) Encrypt KAT
- AES-CBC (128/192/256) Decrypt KAT
- SP 800-90A HMAC DRBG KAT
    - Health-tests initialize, re-seed, and generate.
- KDF-SSH KAT

- Libmd KATs
    - HMAC-SHA-1 KAT
    - HMAC-SHA-256 KAT
    - SHA-512 KAT
- Critical Function Test
    - The cryptographic module performs a verification of a limited operational environment.

Upon successful completion of self-tests, the module outputs “FIPS self-tests completed.” to the local console. If a self-test fails, the module outputs “<self-test name>: Failed” to the local console and automatically reboots.

The module also performs the following conditional self-tests:

- Continuous RNG Test on the SP 800-90A HMAC-DRBG
- Continuous RNG test on the NDRNG
- Pairwise consistency test when generating ECDSA and RSA key pairs.
- Firmware Load Test (ECDSA P-256 with SHA-256 signature verification)

5 Security Rules and Guidance

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. When the module has not been placed in a valid role, the operator does not have access to any cryptographic services.
3. Power up self-tests do not require any operator action.
4. Data output is inhibited during key generation, self-tests, zeroization, and error states.
5. Status information does not contain CSPs or sensitive data that if misused could lead to a compromise of the module.
6. There are no restrictions on which keys or CSPs are zeroized by the zeroization service.
7. The module does not support a maintenance interface or role.
8. The module does not support manual key entry.
9. The module does not output intermediate key values.
10. The module requires two independent internal actions to be performed prior to outputting plaintext CSPs.
11. The cryptographic officer must determine whether firmware being loaded is a legacy use of the firmware load service (legacy being those Junos firmware images signed with RSA signatures instead of ECDSA).
12. The cryptographic officer must retain control of the module while zeroization is in process.

6 References and Definitions

The following standards are referred to in this Security Policy.

Table 18– References

Abbreviation	Full Specification Name
[FIPS140-2]	Security Requirements for Cryptographic Modules, May 25, 2001
[SP800-131A]	Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths, January 2011
[IG]	Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program
[133]	NIST Special Publication 800-133, Recommendation for Cryptographic Key Generation, December 2012
[135]	National Institute of Standards and Technology, Recommendation for Existing Application-Specific Key Derivation Functions, Special Publication 800-135rev1, December 2011.
[186]	National Institute of Standards and Technology, Digital Signature Standard (DSS), Federal Information Processing Standards Publication 186-4, July, 2013.
[186-2]	National Institute of Standards and Technology, Digital Signature Standard (DSS), Federal Information Processing Standards Publication 186-2, January 2000.
[197]	National Institute of Standards and Technology, Advanced Encryption Standard (AES), Federal Information Processing Standards Publication 197, November 26, 2001
[38A]	National Institute of Standards and Technology, Recommendation for Block Cipher Modes of Operation, Methods and Techniques, Special Publication 800-38A, December 2001
[38D]	National Institute of Standards and Technology, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, Special Publication 800-38D, November 2007
[198]	National Institute of Standards and Technology, The Keyed-Hash Message Authentication Code (HMAC), Federal Information Processing Standards Publication 198-1, July, 2008
[180]	National Institute of Standards and Technology, Secure Hash Standard, Federal Information Processing Standards Publication 180-4, August, 2015
[67]	National Institute of Standards and Technology, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, Special Publication 800-67, May 2004

Abbreviation	Full Specification Name
[90A]	National Institute of Standards and Technology, Recommendation for Random Number Generation Using Deterministic Random Bit Generators, Special Publication 800-90A, June 2015.

Table 19 – Acronyms and Definitions

Acronym	Definition
AES	Advanced Encryption Standard
DSA	Digital Signature Algorithm
EC Diffie-Hellman	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
EMC	Electromagnetic Compatibility
ESP	Encapsulating Security Payload
FIPS	Federal Information Processing Standard
HMAC	Keyed-Hash Message Authentication Code
ICV	Integrity Check Value (i.e. Tag)
IOC	Input/Output Card
MD5	Message Digest 5
NPC	Network Processing Card
RE	Routing Engine
RSA	Public-key encryption technology developed by RSA Data Security, Inc.
SHA	Secure Hash Algorithms
SPC	Services Processing Card
SSH	Secure Shell
Triple-DES	Triple - Data Encryption Standard

Table 20 – Datasheets

Model	Title	URL
EX4600	EX4600 Ethernet Switch	https://www.juniper.net/us/en/local/pdf/datasheets/1000511-en.pdf
QFX5100	QFX5100 Ethernet Switch	https://www.juniper.net/us/en/local/pdf/datasheets/1000480-en.pdf
QFX5200	QFX5200 Switch	https://www.juniper.net/assets/uk/en/local/pdf/datasheets/1000560-en.pdf