Juniper Networks NFX250 Network Services Platform

Non-Proprietary FIPS 140-2 Cryptographic Module Security Policy

Document Version: 1.0

Date: April 20, 2021

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

Table of Contents

List of Tables

List of Figures

1 Introduction

The NFX250 Network Services Platform are Juniper Network's secure, automated, software-driven customer premises equipment (CPE) devices that deliver virtualized network and security services on demand. The NFX250 Network Services Platform delivers capacity, performance, and scale to larger enterprise organizations and service providers who are looking to run multiple services on one platform. The NFX250 family provides greater VNF capacity and is integrated with Juniper Networks vSRX Virtual Firewall for the secure delivery of SD-WAN, secure router, and a broad portfolio of managed services. The Juniper Networks NFX250 Network Services Platform cryptographic modules, hereafter referred to as the NFX250 or the modules, run Juniper's Junos firmware Junos OS 20.1R1.

This Security Policy covers the NFX250-S1, NFX250-S1E, and NFX250-S2 models. The cryptographic modules are defined as multiple-chip standalone modules that execute Junos firmware as detailed in the table below. The cryptographic modules provide for an encrypted connection, using SSH, between the management station and the modules. The cryptographic modules also provide for an encrypted connection, using IPSec protocol, between the modules and other IPSec peers. All other data input or output from the NFX250 is considered plaintext for this FIPS 140-2 validation.

Table 1 – Cryptographic Module Configurations

Model	Hardware Versions	Firmware	Routing Engine (RE)	Power	Distinguishing Features
NFX250	NFX250-S1	Junos OS 20.1R1	Built-in RE (RE-NFX250-S1)	Fixed PSU 100-240 VAC	16GB of memory and 100 GB of solid-state drive (SSD) storage; 8 x 10/100/1000BASE-T RJ-45 LAN ports; 2 x 10/100/1000BASE-T RJ-45 LAN/WAN ports;2 x 100/1000BASE-X small form-factor pluggable transceiver (SFP) WAN ports; 2 x 1GbE/10GbE SFP+ WAN ports; 1 x 10/100/1000BASE-T RJ-45 management port; ADSL2/VDSL2 SFP (pluggable into any SFP port)
NFX250	NFX250-S1E	Junos OS 20.1R1	Built-in RE (RE-NFX250-S1E)		16 GB of memory and 200 GB of solid-state drive (SSD)

Model	Hardware Versions	Firmware	Routing Engine (RE)	Power	Distinguishing Features
				Fixed PSU 100-240 VAC	storage; 8 x 10/100/1000BASE-T RJ-45 LAN ports; 2 x 10/100/1000BASE-T RJ-45 LAN/WAN ports; 2 x 100/1000BASE-X small form-factor pluggable transceiver (SFP) WAN ports; 2 x 1GbE/10GbE SFP+ WAN ports; 1 x 10/100/1000BASE-T RJ-45 management port; ADSL2/VDSL2 SFP (pluggable into any SFP port)
NFX250	NFX250-S2	Junos OS 20.1R1	Built-in RE (RE-NFX250-S2)	Fixed PSU 100-240 VAC	32 GB of memory and 400 GB of solid-state drive (SSD) storage; 8 x 10/100/1000BASE-T RJ-45 LAN ports; 2 x 10/100/1000BASE-T RJ-45 LAN/WAN ports; 2 x 100/1000BASE-X SFP WAN ports; 2 x 1GbE/10GbE SFP+ WAN ports; 1 x 10/100/1000BASE-T RJ-45 management port; ADSL2/VDSL2 SFP (pluggable into any SFP port)

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 modules have a limited operational environment as per the FIPS 140-2 definitions. It includes a firmware load service to support necessary updates. Any firmware versions other than Junos OS 20.1R1, loaded into the modules are out of the scope of this validation and require a separate FIPS 140-2 validation.

The modules do not implement any mitigations of other attacks as defined by FIPS 140-2.

1.1 Cryptographic Boundary

The physical form of the modules is depicted in Figures 1,2, 3 and 4 below. The cryptographic boundary is defined as the outer edge of the chassis containing the Routing Engine and Junos firmware image defined in Section 1. The modules exclude the Junos Device Manager (JDM) component of the firmware which comprises of the non-Junos OS User Space applications. The modules also exclude the power supplies from the requirements of FIPS 140-2. The power supplies do not contain any security relevant components and cannot affect the security of the modules.

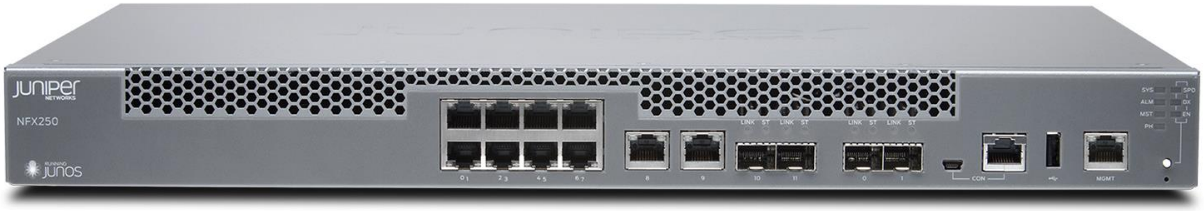

Figure 1 – NFX250-S1/S1E/S2 Front View

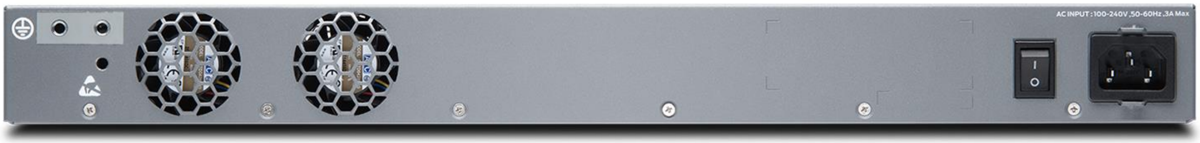

Figure 2 – NFX250-S1/S1E/S2 Back View

Table 3 – Ports and Interfaces

Port	Device (# of ports)	Description	Logical Interface Type
Ethernet	NFX250-S1 (8), NFX250-S1E (8), NFX250-S2 (8)	RJ - 45 LAN Communications	Control in, Data in, Data out, Status out
	NFX250-S1 (2), NFX250-S1E (2), NFX250-S2 (2)	RJ - 45 LAN/WAN Communications	
Ethernet	NFX250-S1 (2), NFX250-S1E (2), NFX250-S2 (2)	1-Gigabit SFP network/uplink ports	
	NFX250-S1 (2), NFX250-S1E (2), NFX250-S2 (2)	1/10-Gigabit SFP+ uplink ports	
Ethernet	NFX250-S1 (1), NFX250-S1E (1), NFX250-S2 (1)	Management port	Control in, Status out
Serial	NFX250-S1 (1), NFX250-S1E (1), NFX250-S2 (1)	Console serial port	Control in, Status out
Mini-USB	NFX250-S1 (1), NFX250-S1E (1), NFX250-S2 (1)	Console mini-USB port	Control in, Status out
USB	NFX250-S1 (1), NFX250-S1E (1), NFX250-S2 (1)	Firmware load port	Control in, Data in
Power	NFX250-S1 (1), NFX250-S1E (1), NFX250-S2 (1)	Power connector	Power
Mode Button	NFX250-S1(1), NFX250-S1E (1), NFX250-S2 (1)	Reset	Control in
LED	NFX250-S1 (15), NFX250-S1E (15), NFX250-S2 (15)	Status indicator lighting	Status out

1.2 Mode of Operation

The NFX250 modules have both a FIPS Approved mode of operation and a non-Approved mode of operation. The NFX250 modules are in a non-FIPS Approved mode by default. The Crypto Officer enables the FIPS-Approved mode of operation and sets up keys and passwords for the system and other FIPS users. The Crypto Officer must put the NFX250 into a FIPS Approved mode by following the steps listed in Section 6.1.2.

1.3 Zeroization

The cryptographic modules provide a non-Approved mode of operation in which non-approved cryptographic algorithms are supported. When transitioning between the Approved mode of operation and the non-Approved mode of operation, the Cryptographic Officer must run the following commands to zeroize the Approved mode CSPs:

```
crypto-officer:fips> request system zeroize
```

Once the NFX250 modules are put into a FIPS Approved mode they remain in the FIPS Approved mode. The only way the modules can leave the FIPS mode is to perform “request system zeroize” which will zeroize the system.

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

2 Cryptographic Functionality

The modules implement the FIPS Approved and Non-Approved but Allowed cryptographic functions listed in Tables 4, 5, 6, 7, 8 and 9 below. The Allowed Protocols in Table 11 summarizes the high-level protocol algorithm support. There may be some algorithm modes that were tested but not implemented by the modules. Only the algorithms, modes, and key sizes that are implemented by the modules are shown in this/these table(s).

2.1 Approved Algorithms

There is a limit of 2^20 encryptions with the same Triple-DES key. The user is responsible for ensuring the module does not surpass this limit. References to standards are given in square bracket [ ]; see Table 7.

Table 4 – Data Plane Approved Cryptographic Functions

CAVP Cert.	Algorithm	Standard	Mode	Key Lengths, Curves, or Moduli	Functions
C1979	AES	PUB 197-38A	CBC	Key Sizes: 128, 192, 256	Encrypt, Decrypt
		SP 800-38D	GCM	Key Sizes: 128, 192, 256	Encrypt, Decrypt, AEAD
C1979	HMAC	PUB 198	SHA-1	Key size: 160 bits, λ = 160	Message Authentication
			SHA-256	Key size: 256 bits, λ = 256	
C1979	SHS	PUB 180-4	SHA-1 SHA-256		Message Digest Generation
C1979	Triple-DES	SP 800-67	TCBC [SP 800-38A]	Key Size: 192	Encrypt, Decrypt

Table 5 – Control Plane QuickSec Approved Cryptographic Functions

CAVP Cert.	Algorithm	Standard	Mode	Key Lengths, Curves, or Moduli	Functions
C2042	AES	PUB 197-38A	CBC	Key Sizes: 128, 192, 256	Encrypt, Decrypt
		SP 800-38D	GCM	Key Sizes: 128, 256	Encrypt, Decrypt, AEAD
N/A[1]	CKG	SP 800 -133rev2	Section 5.1 Section 5.2		Asymmetric seed generation using unmodified DRBG output
			Section 6.2.1		Derivation of symmetric keys
C2042	CVL	SP 800-135	IKEv1	SHA 256, 384	Key Derivation

[1] Vendor Affirmed.

CAVP Cert.	Algorithm	Standard	Mode	Key Lengths, Curves, or Moduli	Functions
			IKEv2	SHA 256, 384	
C2042	DRBG	SP 800-90A	HMAC	SHA-256	Random Bit Generation
C2042	ECDSA	PUB 186-4		P-256 (SHA 256) P-384 (SHA 384)	KeyGen, SigGen, SigVer
C2042	HMAC	PUB 198	SHA-256	Key size: 256 bits, λ = 256	Message Authentication, KDF Primitive
			SHA-384	Key size: 384 bits, λ = 384	
N/A	KTS		AES Cert. #C2042 and HMAC Cert. #C2042		key establishment methodology provides between 128 and 256 bits of encryption strength (AES Mode: CBC; Key Sizes: 128, 192, 256 bits)
			AES Cert. #C2042 and HMAC Cert. #C2042		key establishment methodology provides 128 or 256 bits of encryption strength (AES Mode: GCM; Key Sizes: 128, 256 bits)
			Triple-DES Cert. #C2042 and HMAC Cert. #C2042		key establishment methodology provides 112 bits of encryption strength (Triple-DES Mode: CBC; Key Size: 192 bits)
C2042	RSA	PUB 186-4	PKCS1_V1_5	n=2048 (SHA 256) n=4096 (SHA 256)	SigGen, SigVer[2]
C2042	SHS	PUB 180-4	SHA-256 SHA-384		Message Digest Generation
C2042	Triple-DES	SP 800-67	TCBC	Key Size: 192	Encrypt, Decrypt

Table 6 – OpenSSL Approved Cryptographic Functions

CAVP Cert.	Algorithm	Standard	Mode	Key Lengths, Curves, or Moduli	Functions
C1981	AES	PUB 197-38A	CBC, CTR[3]	Key Sizes: 128, 192, 256	Encrypt, Decrypt
C1981	DRBG	SP 800-90A	HMAC	SHA-256	Random Bit Generation

[2] RSA 4096 SigVer was not tested by the CAVP; however, it is Approved for use per CMVP guidance, because RSA 2048 SigVer was tested and testing for RSA 4096 SigVer is not available.

[3] AES ECB has been tested for testing AES CTR.

CAVP Cert.	Algorithm	Standard	Mode	Key Lengths, Curves, or Moduli	Functions
N/A[4]	CKG	SP 800 - 133rev2	Section 5.1 Section 5.2		Asymmetric seed generation using unmodified DRBG output
			Section 6.2.1		Derivation of symmetric keys
C1981	ECDSA	PUB 186-4		P-256 (SHA 256) P-384 (SHA 384) P-521 (SHA 512)	SigGen, KeyGen, SigVer, PKV
C1981	HMAC	PUB 198	SHA-1	Key size: 160 bits, λ = 160	Message Authentication
			SHA-512	Key size: 512 bits, λ = 512	
			SHA-256	Key size: 256, λ = 256	Message Authentication, DRBG Primitive
N/A[5]	KAS-SSC	SP 800-56Arev3	ECDH	SSHD, PKID, IKED P-256 (SHA 256) P-384 (SHA 384) P-521 (SHA 512)	Key Agreement Scheme - Key Agreement Scheme Shared Secret Computation (KAS-SSC) per SP 800-56Arev3, Key Derivation per SP 800-135 (CVL Cert. #C1981 for SSH and CVL Cert. #C2042 for IKE)
			DH	IKED P-256 (SHA 256) P-384 (SHA 384), Group 24	
N/A	KTS		AES Cert. #C1981 and HMAC Cert. #C1981		key establishment methodology provides between 128 and 256 bits of encryption strength (AES Mode: CBC; Key Sizes: 128, 192, 256 bits)
			Triple-DES Cert. #C1981 and HMAC Cert. #C1981		key establishment methodology provides

[4] Vendor Affirmed.

[5] Vendor Affirmed per IG D.1rev3 (per IG D.8 Scenario X1).

CAVP Cert.	Algorithm	Standard	Mode	Key Lengths, Curves, or Moduli	Functions
					112 bits of encryption strength (Triple-DES Mode: CBC; Key Size: 192 bits)
C1981	RSA	PUB 186-4	Appendix B.3.3	n=2048 (SHA 256, 512) n=4096 (SHA 256, 512)	KeyGen[6]
			PKCS1_V1_5		SigGen, SigVer[7]
C1981	SHS	PUB 180-4	SHA-1 SHA-256 SHA-384		Message Digest Generation, KDF Primitive
			SHA-512		Message Digest Generation
C1981	Triple-DES	SP 800-67	TCBC	Key Size: 192	Encrypt, Decrypt

Table 7 – OpenSSH Approved Cryptographic Functions

CAVP Cert.	Algorithm	Standard	Mode	Key Lengths, Curves, or Moduli	Functions
C1978	CVL	SP 800-135	SSH	SHA 1, 256, 384	Key Derivation

Table 8 – LibMD Approved Cryptographic Functions

CAVP Cert.	Algorithm	Standard	Mode	Key Lengths, Curves, or Moduli	Functions
C1982	HMAC	PUB 198	SHA-1	Key size: 160 bits, λ = 160	Password Hashing
			SHA-256	Key size: 256 bits, λ = 256	
C1982	SHS	PUB 180-4	SHA-1 SHA-256 SHA-512		Message Digest Generation

Table 9 – Kernel Approved Cryptographic Functions

CAVP Cert.	Algorithm	Standard	Mode	Key Lengths, Curves, or Moduli	Functions
C1983	DRBG	SP 800-90A	HMAC	SHA-256	Random Bit Generation

[6] RSA 4096 KeyGen was not tested by the CAVP; however, it is Approved for use per CMVP guidance, because RSA 2048 KeyGen was tested and testing for RSA 4096 KeyGen is not available.

[7] RSA 4096 SigVer was not tested by the CAVP; however, it is Approved for use per CMVP guidance, because RSA 2048 SigVer was tested and testing for RSA 4096 SigVer is not available.

CAVP Cert.	Algorithm	Standard	Mode	Key Lengths, Curves, or Moduli	Functions
C1983	HMAC	PUB 198	SHA-256	Key size: 256, λ = 256	Message Authentication, DRBG Primitive
C1983	SHS	PUB 180-4	SHA-1 SHA-256		Message Digest Generation

2.2 Allowed Algorithms

Table 10 – Allowed Cryptographic Functions

Algorithm	Caveat	Use
NDRNG [IG] 7.14 Scenario 1a	The modules generate a minimum of 256 bits of entropy for key generation.	Seeding the DBRG

2.3 Allowed Protocols

Table 11 – Protocols Allowed in FIPS Mode

Protocol	Key Exchange	Auth	Cipher	Integrity
IKEv1[8]	Diffie-Hellman (L = 2048, N = 256) EC Diffie-Hellman P-256, P-384	RSA 2048 RSA 4096 Pre-Shared Secret ECDSA P-256 ECDSA P-384	Triple-DES CBC AES CBC 128/192/256	SHA-256 SHA-384
IKEv2[9]	Diffie-Hellman (L = 2048, N =256) EC Diffie-Hellman P-256, P-384	RSA 2048 RSA 4096 Pre-Shared Secret ECDSA P-256 ECDSA P-384	Triple-DES CBC AES CBC 128/192/256 AES GCM[10] 128/256	SHA-256 SHA-384
IPsec ESP	IKEv1 with optional: • Diffie-Hellman (L = 2048, N = 256) • EC Diffie-Hellman P-256, P-384	IKEv1	3 Key Triple-DES CBC AES CBC 128/192/256 AES GCM[11] 128/192/256	HMAC-SHA-1-96 HMAC-SHA-256-128

[8] RFC 2409 governs the generation of the Triple-DES encryption key for use with the IKEv1 protocol.

[9] The module uses RFC 7296 compliant IKEv2 to establish the shared secret SKEYSEED, from which all keys are derived to include the AES-GCM and Triple-DES keys.

[10] The AES GCM IV is generated according to RFC5282 and is used only in the context of the IPSec protocol as allowed in IG A.5. Rekeying is triggered after 2^{32} AES GCM transformations.

[11] The AES GCM IV is generated according to RFC4106 and is used only in the context of the IPSec protocol as allowed in IG A.5. Rekeying is triggered after 2^{32} AES GCM transformations.

Protocol	Key Exchange	Auth	Cipher	Integrity
	IKEv2 with optional: • Diffie-Hellman (L = 2048, N = 256) • EC Diffie-Hellman P-256, P-384	IKEv2	3 Key Triple-DES CBC AES CBC 128/192/256 AES z[9]128/192/256	
SSHv2[10]	EC Diffie-Hellman P-256, P-384, P-521	RSA 2048 ECDSA P-256	Triple-DES CBC AES CBC 128/192/256 AES CTR 128/192/256	HMAC-SHA-1 HMAC-SHA-256 HMAC-SHA-512

No part of these protocols, other than the KDF, has been tested by the CAVP and CMVP. The IKE and SSH algorithms allow independent selection of key exchange, authentication, cipher and integrity. In reference to the Allowed Protocols in Table 11 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) and IPSec connect (non-compliant) service.

2.4 Disallowed Algorithms and Protocols

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

Algorithms:

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

Protocols:

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

2.5 Critical Security Parameters

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

Table 12 – Critical Security Parameters (CSPs)

Name	Description and usage
DRBG_Seed	Seed material used to seed or reseed the DRBG
DRBG_State	V and Key values for the HMAC_DRBG
Entropy Input String	256 bits entropy (min) input used to instantiate the DRBG
ECDH Shared Secret	The Diffie-Hellman shared secret used in EC Diffie-Hellman (ECDH) exchange. Created per the EC Diffie-Hellman protocol. Provides between 128-256 bits of security.
DH Shared Secret	The shared secret used in Diffie Hellman (DH) key exchange. 128 bits. Established per the Diffie-Hellman key agreement.
SSH PHK	SSH Private host key. 1st time SSH is configured, the keys are generated. RSA 2048, ECDSA P-256. Used to identify the host.
SSH ECDH	SSH Elliptic Curve Diffie-Hellman private component. Ephemeral Diffie-Hellman private key used in SSH. ECDH P-256, ECDH P-384 or ECDH P-521
SSH-SEKs	SSH Session Keys; SSH Session Encryption Key: TDES (3key) or AES; SSH Session Integrity Key: HMAC
ESP-SEKs	IPSec ESP Session Keys: IKE Session Encryption Key: TDES (3key) or AES; IKE Session Integrity Key: HMAC.
IKE-PSK	Pre-Shared Key used to authenticate IKE connections.
IKE-Priv	IKE Private Key. RSA 2048, RSA 4096, ECDSA P-256, or ECDSA P-384
IKE-SKEYID	IKE SKEYID. IKE secret used to derive IKE and IPsec ESP session keys.
IKE-SEKs	IKE Session Keys: IKE Session Encryption Key: TDES (3key) or AES; IKE Session Integrity Key: HMAC
IKE-DH-PRI	IKE Diffie-Hellman private component. Ephemeral Diffie-Hellman private key used in IKE. DH (L=2048, N = 256), ECDH P-256, or ECDH P-384
HMAC Key	The LibMD HMAC keys: message digest for hashing password and critical function test.
CO-PW	ASCII Text used to authenticate the CO.
User-PW	ASCII Text used to authenticate the User.

Table 13 – Public Keys

Name	Description and usage
SSH-PUB	SSH Public Host Key used to identify the host. RSA 2048, ECDSA P-256.
SSH-ECDH-PUB	Diffie-Hellman public component. Ephemeral Diffie-Hellman public key used in SSH key establishment. ECDH P-256, ECDH P-384 or ECDH P-521.
IKE-PUB	IKE Public Key RSA 2048, RSA 4096, ECDSA P-256, or ECDSA P-384.
IKE-DH-PUB/IKE-ECDH-PUB	IKE Ephemeral Diffie-Hellman public component. Ephemeral Diffie-Hellman public key used in IKE key establishment. DH (L = 2048, N = 256), ECDH P-256, or ECDH P-384.
Auth-UPub	User Authentication Public Keys. Used to authenticate users to the modules. RSA 2048, 4096 or ECDSA P-256, P-384 and P-521.
Auth-COPub	CO Authentication Public Keys. Used to authenticate CO to the modules. RSA 2048, 4096 or ECDSA P-256, P-384 and P-521.
Root-CA	JuniperRootCA. ECDSA P-256 or P-384 X.509 Certificate; Used to verify the validity of the Juniper Package-CA at software load.
Package-CA	PackageCA. ECDSA P-256 X.509 Certificate; Used to verify the validity of Juniper Images at software load and also at runtime integrity.

3 Roles, Authentication and Services

3.1 Roles and Authentication of Operators to Roles

The modules support two roles: Cryptographic Officer (CO) and User. The modules support concurrent operators but do not support a maintenance role and/or bypass capability. The modules enforce the separation of roles using either of the identity-based operator authentication methods in Section 3.2.

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

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

3.2 Authentication Methods

The modules implement 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 modules enforce 10-character passwords (at minimum) chosen from the 96 human readable ASCII characters. The maximum password length is 20-characters.

The modules enforce 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 modules enforce a 5-second delay. Each failed attempt thereafter results in an additional 5-second delay above the previous (e.g. 4th failed attempt = 10-second delay, 5th failed attempt = 15-second delay, 6th failed attempt = 20-second delay, 7th 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 (576 attempts per hour/60 mins); this would be rounded down to 9 per minute, because there is no such thing as 0.6 attempts. 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. The modules support ECDSA (P-256, P-384, and P-521), which has a minimum equivalent computational resistance to attack of either 2^{128} , 2^{192} or 2^{256} depending on the curve. Thus, the probability of a successful random attempt is $1/ (2^{128})$, which is less than 1/1,000,000. Configurable SSH connection establishment rate limits the number of connection attempts, and thus failed authentication attempts in a one-minute period to a maximum of 15,000 attempts. The probability of a success with multiple consecutive attempts in a one-minute period is $15,000/(2^{128})$, which is less than 1/100,000.

RSA signature verification: SSH public-key authentication. The modules support RSA (2048, 4096), which has a minimum equivalent computational resistance to attack of 2^{112} (2048). Thus, the probability of a successful random attempt is $1/ (2^{112})$, which is less than 1/1,000,000. Configurable SSH connection establishment rate limits the number of connection attempts, and thus failed authentication attempts in a one-minute period to a maximum of 15,000 attempts. The probability of a success with multiple consecutive attempts in a one-minute period is $15,000/ (2^{112})$, which is less than 1/100,000.

3.3 Services

All services implemented by the modules are listed in the tables below. Table 16 lists the access to CSPs by each service.

Table 14 – Authenticated Services

Service	Description	CO	User
Configure security	Security relevant configuration.	X	
Configure	Non-security relevant configuration.	X	
Secure Traffic	IPsec protected connection (ESP).	X	
Status	Show status.	X	X
Zeroize	Destroy all CSPs.	X	
SSH connect	Initiate SSH connection for SSH monitoring and control (CLI).	X	X
IPsec connect	Initiate IPsec connection (IKE).	X	
Console access	Console monitoring and control (CLI).	X	X
Remote reset	Software initiated reset conducted over SSH connection to the management port. The remote reset service is used to perform self-tests on demand.	X	
Load Image	Verification and loading of a validated firmware image onto the modules.	X	

Table 15 – Unauthenticated traffic

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

Table 16 – CSP Access Rights within Services

Service	CSPs																
	DRBG_Seed	DRBG_State	Entropy Input String	DH Shared Secret	ECDH Shared Secret	SSH PHK	SSH DH	SSH-SEK	ESP-SEK	IKE-PSK	IKE-Priv	IKE-SKEYID	IKE-SEK	IKE-DH-PRI	HMAC Key	CO-PW	User-PW
Configure security	--	E	--	GW	GW	GWR	--	--	--	WR	GWR	--	--	--	G	W	W
Configure	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Secure traffic	--	--	--	--	--	--	--	--	E	--	--	--	E	--	--	--	--
Status	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Zeroize	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
SSH connect	--	E	--	E	E	E	GE	GE	--	--	--	--	--	--	--	E	E
IPSec connect	--	E	--	--	--	--	--	--	G	E	E	GE	G	GE	--	--	--
Console access	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	E	E
Remote reset	GEZ	GZ	GZ	Z	Z	--	Z	Z	Z	--	--	Z	Z	Z	--	Z	Z
Load Image	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Local reset	GEZ	GZ	GZ	Z	Z	--	Z	Z	Z	--	--	Z	Z	Z	--	Z	Z
Traffic	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

G = Generate: The module generates the CSP
R = Read: The CSP is read from the module (e.g. the CSP is output)
E = Execute: The module executes using the CSP
W = Write: The CSP is updated or written to the module
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) and IPSec Connect (non-compliant). SSH Connect (non-compliant) supports the security functions identified in Section 2.4 and the SSHv2 row of Table 11. The IPsec (non-compliant) supports the DSA in Section 2.4 and the IKEv1, IKEv2 and IPSec rows of Table 11.

Table 17 – Authenticated Services

Service	Description	CO	User
Configure security (non-compliant)	Security relevant configuration.	X	
Configure (non-compliant)	Non-security relevant configuration.	X	
Secure Traffic (non-compliant)	IPsec protected connection (ESP).	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
IPsec connect (non-compliant)	Initiate IPsec connection (IKE).	X	
Console access (non-compliant)	Console monitoring and control (CLI).	X	X
Remote reset (non-compliant)	Software initiated reset conducted over SSH connection to the management port. The remote reset service is used to perform self-tests on demand.	X	
Load Image (non-compliant)	Verification and loading of a validated firmware image into the modules.	X	

Table 18 – 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 modules are powered up, they test that the cryptographic algorithms still operate correctly, and that sensitive data have not been damaged. Power-up self–tests are available on demand by power cycling the modules (Remote reset service).

On power up or reset, the modules perform the self-tests described below. All KATs must be completed successfully prior to any other use of cryptography by the modules. If one of the KATs fails, the module on which the failure has occurred, enters the Critical Failure error state.

The modules each perform the following power-up self-tests:

- **Firmware Integrity** check using ECDSA P-256 with SHA-256
- **Data Plane 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
  - AES-GCM (128/192/256) Encrypt KAT
  - AES-GCM (128/192/256) Decrypt KAT
- **Control Plane QuickSec KATs**
  - SP 800-90A HMAC DRBG KAT
    - Health-tests initialize, re-seed, and generate
  - 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-256 KAT
  - HMAC-SHA-384 KAT
  - AES-CBC (128/192/256) Encrypt KAT
  - AES-CBC (128/192/256) Decrypt KAT
  - AES-GCM (128/256) Encrypt KAT
  - AES-GCM (128/256) Decrypt KAT
  - KDF-IKE-V1 KAT
  - KDF-IKE-V2 KAT
- **OpenSSL KATs**
  - SP 800-90A HMAC DRBG KAT
    - Health-tests initialize, re-seed, and generate.
  - ECDSA P-256 Sign/Verify PCT
  - ECDH P-256 KAT
    - Derivation of the expected shared secret.
  - RSA 2048 w/ SHA-256 Sign KAT
  - RSA 2048 w/ SHA-256 Verify KAT
  - Triple-DES-CBC Encrypt KAT
  - Triple-DES-CBC Decrypt KAT

    - HMAC-SHA-1 KAT
    - HMAC-SHA-256 KAT
    - HMAC-SHA-512 KAT
    - AES-CBC (128/192/256) Encrypt KAT
    - AES-CBC (128/192/256) Decrypt KAT
    - KAS-ECC-EPHEM-UNIFIED-NOKC KAT
    - KAS-FFC-EPHEM-NOKC KAT
- **OpenSSH KATs**
    - KDF-SSH-SHA256 KAT
- **LibMD KATs**
    - HMAC SHA-1
    - HMAC SHA-256
    - SHA-512
- **Kernel KATs**
    - SP 800-90A HMAC DRBG KAT
        - Health-tests initialize, re-seed, and generate
    - HMAC-SHA-256 KAT
    - SHA-1

- **Critical Function Test**
    - The cryptographic modules perform a verification of a limited operational environment, and verification of optional non-critical packages.

The modules also perform the following conditional self-tests:

- Continuous RNG Test on the OpenSSL 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 signature verification).

5 Physical Security Policy

The modules' physical embodiment is that of a multi-chip standalone device that meets Level 1 Physical Security requirements. Each 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. The modules consist of production-grade components that use standard passivation techniques.

6 Security Rules and Guidance

The modules' 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 modules.

1. The modules clear previous authentications on power cycle.
2. When the modules have 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 modules.
6. There are no restrictions on which keys or CSPs are zeroized by the zeroization service.
7. The modules do not support a maintenance interface or role.
8. The modules do not support manual key entry.
9. The modules do not output intermediate key values.
10. The modules require two independent internal actions to be performed prior to outputting plaintext CSPs viz., verification of CO permissions and a specific show command requested by the authenticated CO. These two required internal actions prevent the CSPs from inadvertently being output.
11. The cryptographic officer must verify that the firmware image to be loaded on the NFX250 is a FIPS validated image. If any other non-validated image is loaded the modules will no longer be FIPS validated modules.
12. The cryptographic officer must retain control of the modules while zeroization is in process.
13. Virtualized Network Functions (VNFs) shall not be configured in FIPS-mode of operation.
14. The Triple-DES encryption key is generated as part of recognized IETF protocols (RFC 2409 IKEv1, RFC 4251 SSH, RFC 7296 IKEv2, and RFC 6071 IPSec). The operator is required to ensure that Triple-DES keys used in the SSH and IPsec, IKEv1/v2 protocols do not perform more than $2\^{20}$ encryptions.
15. If the modules lose power and then it is restored, then a new key shall be established for use with the AES GCM encryption/decryption processes.
16. When the IV in RFC 5282 exhausts the maximum number of possible values for a given security association, either party to the security association that encounters this condition triggers a rekeying with IKEv2 to establish a new encryption key for the security association per RFC 7296. The rekeying is triggered after 2^32 values have been used which is less than the (2^64)-1 requirement.
17. 3-key Triple-DES has been implemented in the module and is FIPS approved until December 31, 2023. Should the CMVP disallow the usage of Triple-DES post December 31, 2023, then users must not configure Triple-DES.

6.1 Cryptographic Officer Guidance

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

6.1.1 Installing the FIPS-Approved firmware image

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

Connect to the console port on the module from your management device and log in to the Junos OS CLI. Copy the firmware package to the module to the /var/tmp/ directory. Install the new package on the NFX250 device:

```
root> request vmhost software add  /var/tmp/package.tgz.
```

NOTE: If you need to terminate the installation, do not reboot your device; instead, finish the installation and then issue the request system software delete *package*.tgz command, where *package*.tgz is, for example, jinstall-ppc-20.1R1-signed.tgz. This is your last chance to stop the installation.

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

```
root > request vmhost reboot
```

6.1.2 Enabling FIPS-Approved Mode of Operation

The cryptographic officer shall follow the steps found in the *Junos OS FIPS Evaluated Configuration Guide for NFX250 Network Services Platform, Release 20.1R1* document Chapter 2 to place each module into a FIPS-Approved mode of operation. The steps from the aforementioned document are repeated below:

1. Zeroize the device by following instructions outlined in Section 1.3 to delete all CSPs before entering FIPS mode. Once device comes up in amnesiac mode post zeroize, connect to device using console port with username "root" , configure root authentication and then configure the Crypto Officer credentials:

   ```
   FreeBSD/amd64 (Amnesiac) (ttyu0)
   login: root
   --- JUNOS 20.1R1-20180131.0 Kernel 64-bit JNPR-11.0-20180123.155949_fbsdroot@:~
   # cli
   root>
   ```

   Configure root authentication:

```
root> edit
Entering configuration mode
[edit]
root# set system root-authentication plain-text-password
New password:
Retype new password:
[edit]
root# commit
commit complete
```

Configure Crypto Officer credentials:

```
[edit]
root# set system login user crypto officer class super-user authentication plain-text-password
New password:
Retype new password:
[edit]
root# commit
commit complete
```

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

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

3. Load configuration onto device and commit new configuration.

4. Install the fips-mode package needed for enabling FIPS mode:

```
crypto-officer@device > request system software add optional://fips-mode.tgz
Verified fips-mode signed by PackageDevelopmentEc_2020 method ECDSA256+SHA256
```

5. Install the jpfe-fips package needed for Routing Engine KATS:

```
crypto-officer@device > request system software add optional://jpfe-fips.tgz
Verified jpfe-fips signed by PackageDevelopmentEc_2020 method ECDSA256+SHA256
```

6. Configure the FIPS mode of operation by setting “set system fips level 1” and “set system fips chassis level 1”, followed by commit.

   The device might display that the encrypted password must be re-configured to use FIPS compliant hash warning, to delete older CSP in loaded configuration.

7. After deleting and reconfiguring CSPs, commit will go through and the device needs to be rebooted to enter FIPS mode:

```
[edit]
crypto-officer@device# commit
Generating RSA key /etc/ssh/fips_ssh_host_key
Generating RSA2 key /etc/ssh/fips_ssh_host_rsa_key
Generating ECDSA key /etc/ssh/fips_ssh_host_ecdsa_key
[edit]
system
reboot is required to transition to FIPS level 1
commit complete
root@hostname> request vmhost reboot
```

8. After rebooting the device, FIPS self-tests will run and the device enters FIPS mode:

   ```
   crypto-officer@device:fips>
   ```

9. After the reboot has completed, log in and use the show version command to verify the version:

   ```
   crypto-officer@device:fips > show version
   ```

The module boots up in the FIPS mode which allows only a restricted set of SSH Key algorithms. All Disallowed Algorithms listed in Section 2.4 are disabled.

Direct access to the Junos Device Manager (JDM), from external connections, is disabled in FIPS mode. All connections from external devices, to the modules, are via the Junos Control Plane (JCP).

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

As cryptographic officer, the operator may need to disable the FIPS-Approved mode of operation on the modules to return them to a non-Approved mode of operation. To disable FIPS-Approved mode on the modules, the modules must be zeroized.   Follow the steps found in Section 1.3 to zeroize the modules.

6.2 User Guidance

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

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

All FIPS users must:

- Keep all passwords confidential.
- Store routers or switches and documentation in a secure area.
- Deploy routers or switches in secure areas.
- Check audit files periodically.
- Conform to all other FIPS 140-2 security rules.
- Follow these guidelines:

- Users are trusted.
- Users abide by all security guidelines.
- Users do not deliberately compromise security.
- Users behave responsibly at all times.

7 References and Definitions

The following standards are referred to in this Security Policy.

Table 19 – References

Abbreviation	Full Specification Name
[FIPS 140-2]	*Security Requirements for Cryptographic Modules*, May 25, 2001.
[FIPS 140-2 IG]	*Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program,* June 29, 2020.
[FIPS 180-4]	*National Institute of Standards and Technology, Secure Hash Standard, Federal Information Processing Standards Publication 180-4,* August 2015.
[FIPS 186-4]	*National Institute of Standards and Technology, Digital Signature Standard (DSS), Federal Information Processing Standards Publication 186-4*, July 2013.
[FIPS 197]	*National Institute of Standards and Technology, Advanced Encryption Standard (AES), Federal Information Processing Standards Publication 197,* November 26, 2001.
[FIPS 198-1]	*National Institute of Standards and Technology, The Keyed-Hash Message Authentication Code (HMAC), Federal Information Processing Standards Publication 198-1,* July 2008.
[RFC 2409]	*The Internet Key Exchange (IKE),* November 1998.
[RFC 4106]	*The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP),* June 2005.
[RFC 4251]	*The Secure Shell (SSH) Protocol Architecture,* January 2006.
[RFC 4253]	*The Secure Shell (SSH) Transport Layer Protocol,* January 2006.
[RFC 5282]	*Using Authenticated Encryption Algorithms with the Encrypted Payload of the Internet Key Exchange version 2 (IKEv2) Protocol,* August 2008.
[RFC 6071]	*IP Security (IPsec) and Internet Key Exchange (IKE) Document Roadmap,* February 2011.
[RFC 7296]	*Internet Key Exchange Protocol Version 2 (IKEv2),* October 2014.
[SP 800-38A]	*National Institute of Standards and Technology, Recommendation for Block Cipher Modes of Operation, Methods and Techniques, Special Publication 800-38A,* December 2001.
[SP 800-38D]	*National Institute of Standards and Technology, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC,* November 2007.
[SP 800-56Arev3]	*National Institute of Standards and Technology, Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography,* April 2018.
[SP 800-67rev2]	*National Institute of Standards and Technology, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, Special Publication 800-67rev2,* November 2017.

Abbreviation	Full Specification Name
[SP 800-90Arev1]	*National Institute of Standards and Technology, Recommendation for Random Number Generation Using Deterministic Random Bit Generators, Special Publication 800-90Arev1*, June 2015.
[SP 800-131Arev2]	*National Institute of Standards and Technology, Transitioning the Use of Cryptographic Algorithms and Key Lengths*, March 2019.
[SP 800 133rev2]	*National Institute of Standards and Technology, Recommendation for Cryptographic Key Generation,* June 2020.
[SP 800-135rev1]	*National Institute of Standards and Technology, Recommendation for Existing Application-Specific Key Derivation Functions, Special Publication 800-135rev1,* December 2011.

Table 20 – Acronyms and Definitions

Acronym	Definition
AEAD	Authenticated Encryption with Additional Data
AES	Advanced Encryption Standard
ASCII	American Standard Code for Information Interchange
CA	Certificate Authority
CKG	Cryptographic Key Generation
CLI	Command-line Interface
CO	Cryptographic Officer
CPE	Customer Premises Equipment
CVL	Component Validation Listing
DH	Diffie-Hellman
DRBG	Deterministic Random Number Generator
DSA	Digital Signature Algorithm
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
EMC	Electromagnetic Compatibility
ESD	Electrostatic Discharge
ESP	Encapsulating Security Payload
FIPS	Federal Information Processing Standard
FTP	File Transfer Protocol
HMAC	Keyed-Hash Message Authentication Code
IETF	The Internet Engineering Task Force
IKE	Internet Key Exchange
IPSec	Internet Protocol Security
JCP	Junos Control Plane
JDM	Junos Device Manager
KAS-SSC	Key Agreement Scheme-Shared Secret Computation
KTS	Key-Transport Scheme

Acronym	Definition
MD5	Message Digest 5
NDRNG	Non-Deterministic RNG
NFV	Network Functions Virtualization
PHK	Private Host Key
PKCS	Public Key Cryptography Standards
PSK	Pre-Shared Key
PW	Password
RSA	Rivest–Shamir–Adleman
RFC	Request for Comments
RNG	Random Number Generator
SEK	Session Encryption Key
SFP	Small Form-Factor Pluggable
SHA	Secure Hash Algorithms
SHS	Secure Hash Standard
SSD	Solid-State Drive
SSH	Secure Shell
Triple-DES	Triple - Data Encryption Standard
VNF	Virtualized Network Function

Table 21 – Datasheets

Model	Title	URL
NFX250	NFX Series Network Services Platform	https://www.juniper.net/assets/us/en/local/pdf/datasheets/1000563-en.pdf