Forcepoint

Forcepoint Next Generation Firewall for Desktop Appliances

FIPS 140-3 Non-Proprietary Security Policy

Forcepoint
10900-A Stonelake Blvd.
Austin, TX 78759
United States of America

Phone: 1-858-320-8000
Email: legal@forcepoint.com
www.forcepoint.com

Revision History

Revision	Date	Reason
A	May 7, 2025	Initial release.
B	December 14, 2025	Updates for CMVP Comments
C	May 14, 2026	Updates for CMVP Comments

Trademarks, Copyrights, and Third-Party Software

Preface

This is a non-proprietary Cryptographic Module Security Policy for the Forcepoint Next Generation Firewall for Desktop Appliances (Hardware Version: NGFW N60, NGFW N120, NGFW N120L, NGFW N352, and NGFW N355; Firmware Version: 6.10.13.26655.fips.2) from Forcepoint. This Security Policy describes how the Forcepoint Next Generation Firewall for Desktop Appliances (referred as NGFW appliances, modules, and firewalls) meet the security requirements of Federal Information Processing Standards (FIPS) Publication 140-3, which details the U.S. and Canadian government requirements for cryptographic modules. More information about the FIPS 140-3 standard and validation program is available on the National Institute of Standards and Technology (NIST) and the Canadian Centre for Cyber Security (CCCS) Cryptographic Module Validation Program (CMVP) website at https://csrc.nist.gov/projects/cryptographic-module-validation-program

This document also describes how to run the modules in a secure Approved mode of operation. This policy was prepared as part of the Level 1 FIPS 140-3 validation of the module. The Forcepoint Next Generation Firewall for Desktop Appliances are referred to in this document as the NGFW appliances, crypto modules, or modules.

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

The NGFW appliances are high-performance network security appliances that add a broad range of built-in security features, including VPN, IPS, anti-evasion, TLS inspection, SD-WAN, and mission-critical application proxies, to a traditional firewall and provides end-to-end protection across the entire enterprise network. All appliances can be deployed as either a Layer 2 or Layer 3 firewall or a next generation IPS. However, in the FIPS 140-3 approved mode, the appliances are deployed in Firewall/VPN mode of operation, which provides access control and VPN connectivity. Each of the appliances run NGFW firmware version 6.10.13.26655.fips.2 based on the NGFW OS 10 operating system with Linux Kernel version 4.19.

1.2 Security Levels

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

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The Forcepoint NGFW 60, 120, and 350 series bring together a wide range of capabilities in compact desktop appliances that are easy to install, even in smaller locations. They integrate multi-ISP SD-WAN connectivity, site-to-site Multi-Link VPN, and high-availability clustering with next-generation firewall (NGFW) and intrusion prevention (IPS) security. The appliances enable remote offices, branch

offices, and stores to securely connect directly to the cloud. All Forcepoint NGFW appliances are centrally managed using the Forcepoint Security Management Center (SMC).

Module Type: Hardware

Module Embodiment: Multi-Chip Standalone

Module Characteristics:

Cryptographic Boundary:

The cryptographic boundary of the module is defined as the outer edges of the NGFW N60, NGFW N120, NGFW N120L, NGFW N352, and NGFW N355 chassis. This includes all ports and physical interfaces.

Figure 1: NGFW N60 Front

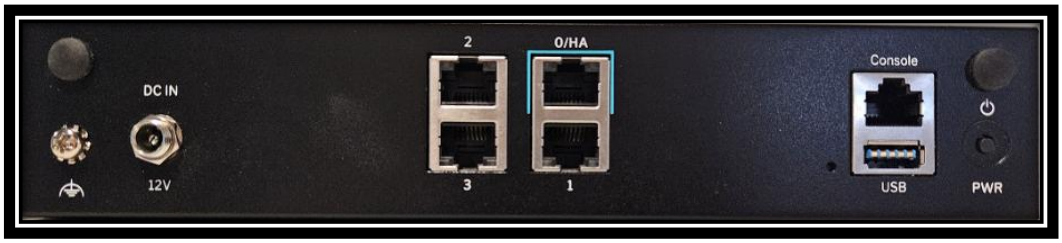

Figure 2: NGFW N60 Rear

Figure 3: NGFW N120 Front

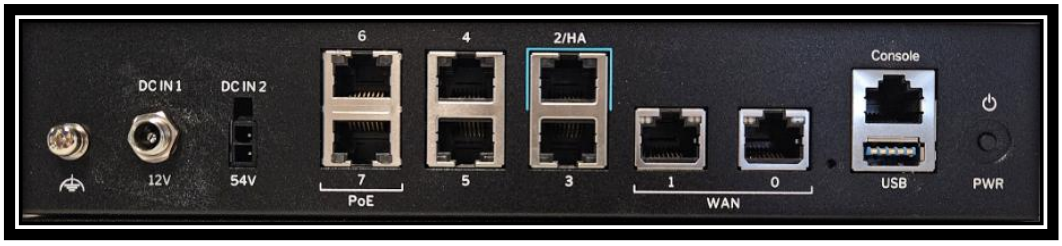

Figure 4: NGFW N120 Rear

Figure 5: NGFW N120L Front

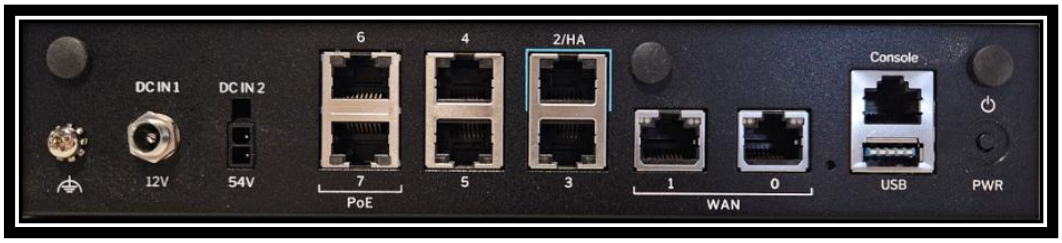

Figure 6: NGFW N120L Rear

Figure 7: NGFW N352/N355 Front

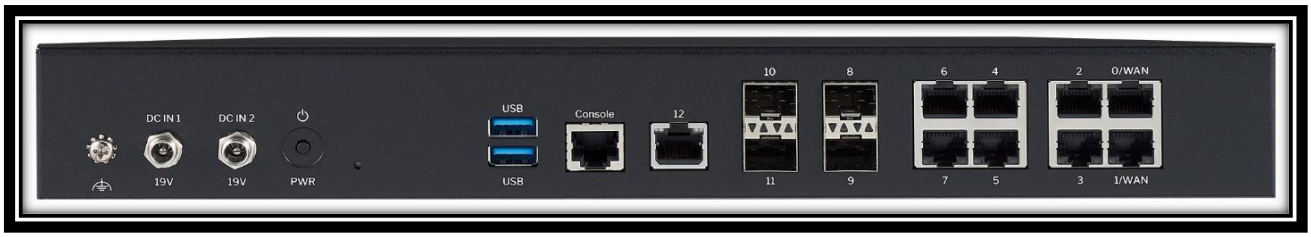

Figure 8: NGFW N355/N355 Rear

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
NGFW N120, Part Number: 120-C3	N120	6.10.13.26655.fips.2	Intel Atom C3338R (Goldmont)	8x1 Gbps Ethernet interfaces Desktop form factor
NGFW N120L, Part Number: 120-C4	N120L	6.10.13.26655.fips.2	Intel Atom C3338R (Goldmont)	8x1 Gbps Ethernet interfaces Desktop form factor
NGFW N352, Part Number: 352-C1	N352	6.10.13.26655.fips.2	Intel Atom C5315 (Tremont)	8x1 Gbps Ethernet interfaces 1x2.5 Gbps Ethernet interface 4x10 Gbps SFP+ interfaces Desktop form factor

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
NGFW N355, Part Number: 355-C1	N355	6.10.13.26655.fips.2	Intel Atom C5325 (Tremont)	8x1 Gbps Ethernet interfaces 1x2.5 Gbps Ethernet interface 4x10 Gbps SFP+ interfaces Desktop form factor
NGFW N60, Part Number: 60-C1	N60	6.10.13.26655.fips.2	Intel Atom C3338R (Goldmont)	4x1 Gbps Ethernet interfaces Desktop form factor

Table 2: Tested Module Identification – Hardware

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

N/A for this module.

Tested Module Identification – Hybrid Disjoint Hardware:

N/A for this module.

Tested Operational Environments - Software, Firmware, Hybrid:

N/A for this module.

Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid:

N/A for this module.

2.3 Excluded Components

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved Mode	This module only supports an Approved Mode of Operation where only approved security services are provided by the module.	Approved	As per IG 2.4.C Option 1, the module returns a separate indicator for each approved security service. This indicator is a log message that outputs "Started in FIPS 140 operating mode."

Table 3: Modes List and Description

The module supports the Approved mode of operation only. When installed, configured, and operated according to this Security Policy, the module does not support a non-Approved mode of operation.

2.5 Algorithms

Approved Algorithms:

The following cryptographic library and associated CAVP certificates are used by the cryptographic module:

- Forcepoint NGFW FIPS Cryptographic Module (Cert. #A4793)
- Forcepoint NGFW FIPS Library (Cert. #A4550)
- Forcepoint NGFW Cryptographic Kernel Module (Cert. #A2166)
- Forcepoint NGFW Entropy Library (Cert. #A4426)

Only the algorithms specified in this section are supported by the module in approved mode of operation.

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A2166	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CBC	A4793	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A4793	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A2166	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 192, 256	SP 800-38D
AES-GCM	A4793	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 192, 256	SP 800-38D
AES-KWP	A4793	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
Counter DRBG	A4793	Prediction Resistance - No Mode - AES-256 Derivation Function Enabled - Yes	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-5)	A4793	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
ECDSA KeyVer (FIPS186-5)	A4793	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A4793	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A4793	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A2166	Key Length - Key Length: 112-512 Increment 8	FIPS 198-1
HMAC-SHA-1	A4793	Key Length - Key Length: 112-2048 Increment 8	FIPS 198-1
HMAC-SHA2-256	A2166	Key Length - Key Length: 112-512 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4550	Key Length - Key Length: 112-2048 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4793	Key Length - Key Length: 112-2048 Increment 8	FIPS 198-1
HMAC-SHA2-384	A2166	Key Length - Key Length: 112-1024 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4793	Key Length - Key Length: 112-2048 Increment 8	FIPS 198-1
HMAC-SHA2-512	A2166	Key Length - Key Length: 112-1024 Increment 8	FIPS 198-1
HMAC-SHA2-512	A4793	Key Length - Key Length: 112-2048 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4793	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KAS-FFC-SSC Sp800-56Ar3	A4793	Domain Parameter Generation Methods - FB, FC, ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, modp-2048, modp-3072, modp-4096, modp-6144, modp-8192 Scheme - dhEphem - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDF IKEv1 (CVL)	A4793	Authentication Method - Digital Signature, Pre-shared Key Diffie-Hellman Shared Secret Length - Diffie-Hellman Shared Secret Length: 256, 2048, Diffie-Hellman Shared Secret Length: 256, 3072, Diffie-Hellman Shared Secret Length: 384, 4096, 6144, Diffie-Hellman Shared Secret Length: 528, 8192	SP 800-135 Rev. 1

Algorithm	CAVP Cert	Properties	Reference
		Hash Algorithm - SHA-1, SHA2-256, SHA2-384, SHA2-512 Preshared Key Length - Preshared Key Length: 112, 1136	
KDF IKEv2 (CVL)	A4793	Diffie-Hellman Shared Secret Length - Diffie-Hellman Shared Secret Length: 256, 2048, Diffie-Hellman Shared Secret Length: 256, 3072, Diffie-Hellman Shared Secret Length: 384, 4096, 6144, Diffie-Hellman Shared Secret Length: 528, 8192 Derived Keying Material Length - Derived Keying Material Length: 320, 1312, Derived Keying Material Length: 320, 1792, Derived Keying Material Length: 320, 2432, Derived Keying Material Length: 320, 3072 Hash Algorithm - SHA-1, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
KDF SP800-108	A4550	KDF Mode - Counter, Feedback Supported Lengths - Supported Lengths: 128, 3456, 4096, 72, 776, 8	SP 800-108 Rev. 1
PBKDF	A4550	Iteration Count - Iteration Count: 1-10000 Increment 1 Password Length - Password Length: 8-128 Increment 1	SP 800-132
RSA KeyGen (FIPS186-5)	A4793	Key Generation Mode - probable, probableWithProbableAux Hash Algorithm - SHA2-256 Modulo - 2048, 3072, 4096 Primality Tests - 2pow100 Private Key Format - standard	FIPS 186-5
RSA SigGen (FIPS186-5)	A4793	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-5)	A4793	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
Safe Primes Key Generation	A4793	Safe Prime Groups - ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, modp-2048, modp-3072, modp-4096, modp-6144, modp-8192	SP 800-56A Rev. 3
Safe Primes Key Verification	A4793	Safe Prime Groups - ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, modp-2048, modp-3072, modp-4096, modp-6144, modp-8192	SP 800-56A Rev. 3
SHA-1	A2166	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA-1	A4793	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-224	A4793	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A2166	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A4550	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A4793	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A2166	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A4793	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A2166	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A4793	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA3-256	A4426	Message Length - Message Length: 0-65536 Increment 8	FIPS 202

Algorithm	CAVP Cert	Properties	Reference
TLS v1.2 KDF RFC7627 (CVL)	A4793	Hash Algorithm - SHA2-256, SHA2-384	SP 800-135 Rev. 1

Table 4: Approved Algorithms -

Legacy

Algorithm	CAVP Cert	Properties	Reference
RSA SigVer (FIPS186-4)	A4793	Signature Type - PKCS 1.5 Modulo - 1024	FIPS 186-4

Table 5: Approved Algorithms - Legacy

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
Symmetric CKG	Key Type:Symmetric	Forcepoint NGFW FIPS Cryptographic Module	Section 4 (Example 1) and Section 6.1 (via the direct generation of keys using an approved Counter DRBG) of NIST SP 800-133 Rev. 2, and IG D.H.
Asymmetric CKG	Key Type:Asymmetric	Forcepoint NGFW FIPS Cryptographic Module	Section 4 (Example 1) , Section 5.1, and Section 5.2 of NIST SP 800-133rev2 and IG D.H

Table 6: 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:

The module does not offer non-Approved algorithms not allowed in the Approved mode of operation.

N/A for this module.

2.6 Security Function Implementations

The table below lists the security function implementations for this module.

Name	Type	Description	Properties	Algorithms
AES CBC Encryption/Decryption for IPsec/VPN	BC-UnAuth	Data Encryption/Decryption used for IPsec/VPN		AES-CBC: (A2166) Key Sizes: 128, 192, 256 bits
AES CBC Encryption/Decryption for TLS	BC-UnAuth	Data Encryption/Decryption as part of TLS.	Special Publication :SP 800-38A	AES-CBC: (A4793) Key Sizes: 128 and 256 bits
AES Decryption for Modify and Apply Configuration	BC-UnAuth	Key Decryption for Modify and Apply Configuration Service		AES-CBC: (A4793) Key Sizes: 128 and 256 bits
AES Encryption/Decryption for Configuration File Protection	BC-UnAuth	Data Encryption/Decryption for configuration files stored on disk		AES-CBC: (A4793) Key Sizes: 128 and 256 bits
AES Encryption/Decryption for Key Pair Management	BC-UnAuth	Data Encryption/Decryption for key pairs stored on disk		AES-CBC: (A4793) Key Sizes: 256 bits
AES-CBC Encryption/Decryption for SNMP	BC-UnAuth	Data Encryption/Decryption for SNMP		AES-CBC: (A4793) Key Sizes: 128 and 256 bits
AES-CBC Encryption/Decryption for State Synchronization	BC-UnAuth	Data Encryption/Decryption as part of State Synchronization for Peer Connection Service		AES-CBC: (A4793) Key Sizes: 128 bits
AES-GCM Encryption/Decryption for IPsec/VPN	BC-Auth	Data Encryption/Decryption used for IPsec/VPN		AES-GCM: (A2166) Key Sizes: 128, 192, 256 bits AES-CBC: (A2166)
AES-GCM Encryption/Decryption for TLS	BC-Auth	Data Encryption/Decryption as part of TLS.	Special Publication:SP 800-38D	AES-GCM: (A4793) Key Sizes: 128 and 256 bits AES-CBC: (A4793)
Authentication for HTTPS and Mobile VPN	SHA	Authentication for HTTPS and Mobile VPN using SHA2-512		SHA2-512: (A4793)

Name	Type	Description	Properties	Algorithms
Authentication for IPsec VPN	MAC	Authentication for IPsec VPN PSK		HMAC-SHA-1: (A2166) Key Sizes: 112-512 bits MAC Length: 32-160 bits HMAC-SHA2-256: (A2166) Key Sizes: 112-512 bits MAC Length: 32-256 bits HMAC-SHA2-384: (A2166) Key Sizes: 112-1024 bits MAC Length: 32-384 bits HMAC-SHA2-512: (A2166) Key Sizes: 112-1024 bits MAC Length: 32-512 bits SHA-1: (A2166) SHA2-256: (A2166) SHA2-384: (A2166) SHA2-512: (A2166)
CKG	CKG	As per section 6.1 and Section 4 of NIST SP 800-133, the module uses its Approved DRBG to generate cryptographic keys. The resulting symmetric or asymmetric key, or generated seed is an unmodified output from the DRBG.	Symmetric Keys:Configuration File Protection Passphrase, Configuration File Protection Key, and State Synchronization Key	Symmetric CKG : () ECDSA KeyGen (FIPS186-5): (A4793) Curves: P-224, P-256, P-384, and P-521 RSA KeyGen (FIPS186-5): (A4793) Modulus: 2048 and 3072 bits KAS-FFC-SSC Sp800-56Ar3: (A4793) KAS-FFC Keys for TLS: ffdhe2048/3072/4096/6144/8192 KAS-FFC Keys for IKE: MODP-2048/3072/4096/6144/8192 Counter DRBG: (A4793) Asymmetric CKG: () Key Type: Asymmetric
DRBG	DRBG	Get random bits from the Counter DRBG		Counter DRBG: (A4793) AES-ECB: (A4793) Key Sizes: 256 bits
ECDSA Key Generation	AsymKeyPair-KeyGen	Key Pair Generation for TLS/VPN		Asymmetric CKG: () Key Type: Asymmetric Counter DRBG: (A4793) ECDSA KeyGen (FIPS186-5): (A4793)

Name	Type	Description	Properties	Algorithms
				Curves: P-224, P-256, P-384 and P-521 curves
ECDSA Key Verification	AsymKeyPair-KeyVer	Key Pair Verification for TLS/VPN		ECDSA KeyVer (FIPS186-5): (A4793) Curves: P-224, P-256, P-384, and P-521
ECDSA Signature Generation	DigSig-SigGen	Signature Generation for TLS/VPN		ECDSA SigGen (FIPS186-5): (A4793) SHA2-256: (A4793) SHA2-384: (A4793) SHA2-512: (A4793)
ECDSA Signature Verification	DigSig-SigVer	Signature Verification for TLS/VPN		ECDSA SigVer (FIPS186-5): (A4793) Curves: P-224, P-256, P-384, and P-521 SHA2-224: (A4793) SHA2-256: (A4793) SHA2-384: (A4793) SHA2-512: (A4793)
Entropy source	ENT-ESV	Non-physical entropy source with a conditioning function		SHA3-256: (A4426)
Firmware integrity	DigSig-SigVer	Firmware signature verification		ECDSA SigVer (FIPS186-5): (A4793) Curve: P-256 SHA2-512: (A4793)
HMAC for Configuration File Authentication	MAC	HMAC for the authentication of configuration files stored on disk		HMAC-SHA2-256: (A2166) Key Sizes: 112-512 bits MAC Length: 32-160 bits SHA2-256: (A2166)
HMAC for IKE	MAC	HMAC for the authentication of IKEv1/IKEv2 packets		SHA-1: (A4793) SHA2-256: (A4793) SHA2-384: (A4793) SHA2-512: (A4793) HMAC-SHA-1: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-160 bits HMAC-SHA2-256: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-256 bits HMAC-SHA2-384: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-384 bits

Name	Type	Description	Properties	Algorithms
				HMAC-SHA2-512: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-512 bits
HMAC for IPsec/VPN	MAC	Authentication Key used in IPsec/VPN connections.		HMAC-SHA2-256: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-256 bits SHA2-256: (A4793) HMAC-SHA2-384: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-384 bits SHA2-384: (A4793) HMAC-SHA2-512: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-512 bits SHA2-512: (A4793) HMAC-SHA-1: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-160 bits SHA-1: (A4793)
HMAC for Key Pair Management	MAC	HMAC for the authentication of key pairs stored on disk		SHA2-256: (A2166) HMAC-SHA2-256: (A2166) Key Sizes: 256 bits MAC Length: 32-256 bits
HMAC for Peer Heartbeat	MAC	HMAC for the authentication of Peer Heartbeat messages		HMAC-SHA-1: (A2166) Key Sizes: 256 bits MAC Length: 32-160 bits SHA-1: (A2166)
HMAC for Peer State Synchronization	MAC	HMAC for the authentication of Peer State Synchronization messages		HMAC-SHA-1: (A2166) Key Sizes: 160 bits MAC Length: 32-160 bits SHA-1: (A2166) Counter DRBG: (A4793)
HMAC for SNMP	MAC	HMAC for the Authentication of SNMP packets		HMAC-SHA-1: (A2166) Key Sizes: 160 bits MAC Length: 32-160 bits SHA-1: (A2166)
HMAC for TLS	MAC	Authentication Key used in TLS session.		HMAC-SHA2-256: (A4793) Key Sizes: 256 bits MAC Length: 80-256 bits SHA2-256: (A4793) HMAC-SHA2-384: (A4793) Key Sizes: 256 bits MAC Length: 80-384 bits SHA2-384: (A4793)

Name	Type	Description	Properties	Algorithms
IPSec Key Agreement using KAS-ECC-SSC	KAS-Full	Key Agreement for IPSec using KAS-ECC-SSC	Caveat:No parts of the IKEv1 or IKEv2 protocol, other than the KDF, have been tested by CAVP. Strength:SSP establishment methodology provides between 112 and 256 bits of encryption strength	KDF IKEv1: (A4793) KDF IKEv2: (A4793) SHA2-256: (A4793) KAS-ECC-SSC Sp800-56Ar3: (A4793) Scheme: ephemeralUnified ECDSA KeyGen (FIPS186-5): (A4793) ECDSA KeyVer (FIPS186-5): (A4793) Counter DRBG: (A4793) Asymmetric CKG: () Key Type: Asymmetric
IPSec Key Agreement using KAS-FFC-SSC	KAS-Full	Key Agreement for IPSec using KAS-FFC-SSC	Caveat:No parts of the IKEv1 or IKEv2 protocol, other than the KDF, have been tested by CAVP. Strength:Key establishment methodology provides between 112 and 202 bits of encryption strength.	KDF IKEv1: (A4793) KDF IKEv2: (A4793) KAS-FFC-SSC Sp800-56Ar3: (A4793) Scheme: dhEphem Safe Primes Key Generation: (A4793) Safe Primes: modp-2048, modp-3072, modp-4096, modp-6144, and modp-8192 Safe Primes Key Verification: (A4793) Safe Primes: modp-2048, modp-3072, modp-4096, modp-6144, and modp-8192 SHA2-256: (A4793)
KAS-ECC Full Public Key Validation	AsymKeyPair-PubKeyVal	Full Public Key Validation for Ephemeral Key Pairs	Special Publication:SP 800-56Arev3 (Section: 5.6.2.3.3)	ECDSA KeyVer (FIPS186-5): (A4793)
KAS-FFC Full Public Key Validation	AsymKeyPair-PubKeyVal	Full Public Key Validation for Ephemeral Key Pairs	Special Publication :SP 800-56Arev3 (Section: 5.6.2.3.1)	Safe Primes Key Verification: (A4793)

Name	Type	Description	Properties	Algorithms
KBKDF	KBKDF	Key-Based Key Derivation		KDF SP800-108: (A4550) Mode: Counter and Feedback SHA2-256: (A4793) HMAC-SHA2-256: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-256 bits
KTS for TLS/IPsec Connections (AES-CBC + HMAC)	KTS-Wrap	Key transport (Encryption with Message Authentication)	Strength:SSP establishment methodology provides between 128 and 256 bits of encryption strength.	AES-CBC: (A4793) Key Sizes: 128 and 256 bits HMAC-SHA2-256: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-256 bits SHA2-256: (A4793) HMAC-SHA2-384: (A4793) Key Sizes: 112-2048 bits MAC Length: 80-384 bits SHA2-384: (A4793)
KTS for TLS/IPsec Connections (AES-GCM)	KTS-Wrap	Key transport (Encryption with Message Authentication)	Strength:SSP establishment methodology provides between 128 and 256 bits of encryption strength.	AES-GCM: (A4793) Key Sizes: 128 and 256 bits AES-CBC: (A4793)
KTS for VPN	KTS-Wrap	Key Wrapping used in VPN Connections	Special Publication:SP 800-38F	AES-KWP: (A4793) Key Sizes: 256 bits AES-ECB: (A4793) Key Sizes: 256 bits
PBKDF	PBKDF	Password-based key derivation		PBKDF: (A4550) Key Sizes: 128 and 256 bits HMAC-SHA2-256: (A4550) Key Sizes: 256 bits MAC Length: 256 bits SHA2-256: (A4550)
RSA Key Generation	AsymKeyPair-KeyGen	Key Pair Generation for TLS/VPN.	Special Publication :FIPS 186-5	Counter DRBG: (A4793) Asymmetric CKG: () Key Type: Asymmetric RSA KeyGen (FIPS186-5): (A4793) Modulus Size: 2048, 3072, and 4096 bits
RSA Signature Generation	DigSig-SigGen	Signature Generation for TLS/VPN		RSA SigGen (FIPS186-5): (A4793) Modulo Size: 2048, 3072,

Name	Type	Description	Properties	Algorithms
				and 4096 SHA2-384: (A4793) SHA2-512: (A4793) SHA2-256: (A4793)
RSA Signature Verification (FIPS 186-4)	DigSig-SigVer	FIPS 186-4 Signature Verification for TLS/VPN		RSA SigVer (FIPS186-4): (A4793) Modulo Size: 1024 SHA-1: (A4793) SHA2-224: (A4793) SHA2-256: (A4793) SHA2-384: (A4793) SHA2-512: (A4793)
RSA Signature Verification (FIPS 186-5)	DigSig-SigVer	FIPS 186-5 Signature Verification for TLS/VPN		RSA SigVer (FIPS186-5): (A4793) Modulo Size: 2048, 3072, and 4096 SHA-1: (A4793) SHA2-384: (A4793) SHA2-512: (A4793) SHA2-224: (A4793)
Safe Prime Key Generation	KAS-KeyGen	Key generation for all module services that utilize KAS-FFC-SSC.		Safe Primes Key Generation: (A4793) Safe Prime Groups: ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, modp-2048, modp-3072, modp-4096, modp-6144, and modp-8192 Counter DRBG: (A4793)
Safe Prime Key Verification	KAS-SSC	Key Verification for all module services that utilize KAS-FFC-SSC.		Safe Primes Key Verification: (A4793) Safe Prime Groups: Safe Prime Groups : ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, modp-2048, modp-3072, modp-4096, modp-6144, and modp-8192
TLS Key Agreement using KAS-ECC-SSC	KAS-Full	Key Agreement for TLS using KAS-ECC-SSC.	Caveat:No parts of the TLS protocol, other than the KDF, have been tested by	TLS v1.2 KDF RFC7627: (A4793) MAC Algorithms: HMAC-SHA2-256 and HMAC-SHA2-384 SHA2-256: (A4793)

Name	Type	Description	Properties	Algorithms
			CAVP. Key Strength:SSP establishment methodology provides between 112 and 256 bits of encryption strength	HMAC-SHA2-256: (A4793) Key Sizes: 256 bits MAC Lengths: 80-256 bits KAS-ECC-SSC Sp800-56Ar3: (A4793) Scheme: ephemeralUnified SHA2-384: (A4793) HMAC-SHA2-384: (A4793) Key Sizes: 384 bits MAC Length : 80-384 bits ECDSA KeyGen (FIPS186-5): (A4793) ECDSA KeyVer (FIPS186-5): (A4793) Counter DRBG: (A4793) Asymmetric CKG: () Key Type: Asymmetric
TLS Key Agreement using KAS-FFC-SSC	KAS-Full	Key Agreement for TLS using KAS-FFC-SSC	Caveat:No parts of the TLS protocol, other than the KDF, have been tested by CAVP. Strength:Key establishment methodology provides between 112 and 202 bits of encryption strength.	TLS v1.2 KDF RFC7627: (A4793) MAC Algorithm: HMAC-SHA2-256 and HMAC-SHA2-384 SHA2-256: (A4793) HMAC-SHA2-256: (A4793) Key Sizes: 256 bits MAC Length: 80-256 bits KAS-FFC-SSC Sp800-56Ar3: (A4793) Scheme: dhEphem Safe Primes Key Generation: (A4793) Generated Safe Primes: ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, and ffdhe8192 Safe Primes Key Verification: (A4793) Verified Safe Primes: ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, and ffdhe8192 SHA2-384: (A4793) HMAC-SHA2-384: (A4793) Key Sizes: 384 bits MAC Length: 80-384 bits

Table 7: Security Function Implementations

2.7 Algorithm Specific Information

- IG C.D - Truncated HMAC
  - HMAC-SHA-1
    - Security Strength: 112-bits
    - MAC Sizes: 80-160 bits
  - HMAC-SHA2-256
    - Security Strength: 112-bits
    - MAC Sizes: 80-256 bits
  - HMAC-SHA2-384
    - Security Strength: 112-bits
    - MAC Sizes: 80-384 bits
  - HMAC-SHA2-512
    - Security Strength: 112-bits
    - MAC Sizes: 80-512 bits

- IG C.M – Legacy Algorithms
  - Algorithm: RSA SigVer (FIPS186-4)
  - Modulus Sizes: 1024 bits
  - Caveat: Legacy usage only. This legacy algorithms can only be used on data that was generated prior to the Legacy Date specified in FIPS 140-3 IG C.M.

2.8 RBG and Entropy

The table below specifies the module's entropy certificates.

Cert Number	Vendor Name
E126	Forcepoint

Table 8: Entropy Certificates

The table below specifies the module's entropy sources.

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Forcepoint NGFW Entropy Library	Non-Physical	Intel Atom® C3338R (Goldmont)/Intel Atom® C5315 (Tremont)/Intel Atom® C5325 (Tremont) on a NGFW OS 10 on Linux 4.19	256 bits	256 bits	SHA3-256 (A4426)

Table 9: Entropy Sources

The module's entropy source is expected to provide full entropy. A total of 512 bits of this full-entropy data is used to seed the module's CTR_DRBG via its entropy input. This amount of entropy is sufficient to support 256 bits of security strength, in accordance with NIST SP 800-57.

For more details about the entropy source please see the public use document for ESV certificate number E126.

2.9 Key Generation

As per IG D.H and SP 800-133rev2, the module uses an Approved Counter DRBG to generate random values and seeds that are used for asymmetric and symmetric key generation. The generated seed is an unmodified output from the Counter DRBG.

2.10 Key Establishment

The module supports the following Key Establishment methods:

1. Key Agreement Schemes:
    - SP 800-56Arev3 KAS-FFC-SSC
    - SP 800-56Arev3 KAS-ECC-SSC
2. Key Transport Schemes
    - SP 800-38F Compliant SSP Transport via AES-CBC and HMAC
    - SP 800-38F Compliant SSP Transport via AES-GCM
    - SP 800-38F Compliant SSP Transport via AES-KWP

All Key Establishment methods are entered and outputted using the AD/EE method as per IG 9.5.A.

2.11 Industry Protocols

The following industry protocols[1] are used by the module:

- TLS
- IPsec
- IKEv1
- IKEv2
- SNMPv3

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

[1] No parts of the TLS v1.2, IKEv1, IKEv2, and SNMPv3 protocols, other than the KDF, have been tested by the CAVP or CMVP

Physical Port	Logical Interface(s)	Data That Passes
USB ports (x2) [N60/N60L/N120/120L]	None	Can be used to input initial configuration from SMC. Disabled after initial configuration has been loaded.
SIM Card Slot (x1) [N60L]	None	Used to hold a Nano-SIM card for the LTE capabilities
Fixed Gb ethernet port (x4)	Data Input Data Output Control Input Control Output Status Output	Network traffic
Rear fixed ethernet port LEDs (x8) [N60/N60L]	Status Output	Used to indicate link status and network activity
Front Ethernet Indicator LEDs (x8) [N60/N60L/N120/120L]	Status Output	Used to indicate port activity and link status
Console port	Status Output	Used for external connections to console monitors, which can be used for status monitoring. Can be used to input initial configuration from SMC, but disabled after initial configuration has completed.
Power LED (x1) [N60/N60L/N120/120L]	Status Output	Used to indicate whether the module is running, in a standby state, or powered down
Disk status LEDs (x1)	Status Output	Used to determine the presence of disk activity
MGMT LED (x1)	Status Output	Used to determine the presence management connections
HA LED (x1)	Status Output	Used to determine if the module is running in High Availability
LTE LED (x1) [N60L/120L]	Status Output	Used to determine the strength of the LTE connection
Power button (x1)	Control Input	Used to turn on and turn off the module
12V Power Connector (x1) [N60/N60L/N120/120L]	Power	Used to input power to the module
Fixed Gb ethernet port (x2) [N120/120L]	Data Input Data Output Control Input Control Output Status Output	Network traffic/PoE

Physical Port	Logical Interface(s)	Data That Passes
Fixed Gb WAN ethernet port (x2) [N120/120L/N352/N355]	Data Input Data Output Control Input Control Output Status Output	Network traffic
Rear fixed ethernet port LEDs (x16) [N120/120L/N352/N355]	Status Output	Used to indicate link status and network activity
PoE LED (x1) [N120/120L/N352/N355]	Status Output	Used to determine if there is any power that is actively fed through a supported Ethernet port
54V Power Connector (x1) [N120/120L]	Power	Used to input power to the module
USB ports (x3) [N352/N355]	Control Input	Can be used to input initial configuration from SMC
Fixed Ethernet RJ45 ports (x8) [N352/N355]	Data Input Data Output Control Input Control Output Status Output	Network traffic
Fixed Ethernet 2.5 Gbps RJ45 ports (x1) [N352/N355]	Data Input Data Output Control Input Control Output Status Output	Network traffic
Fixed SFTP+ ports (x4) [N352/N355]	Data Input Data Output Control Input Control Output Status Output	Network traffic
Rear fixed ethernet port LEDs (x10)[N352/N355]	Status Output	Used to indicate link status and network activity
Front Ethernet Indicator LEDs (x26) [N352/N355]	Status Output	Used to indicate port activity and link status

Physical Port	Logical Interface(s)	Data That Passes
Power LED (x2) [N352/N355]	Status Output	Used to indicate whether the module is running, in a standby state, or powered down
19V Power Connector (x1) [N352/N355]	Power	Used to input power to the module
Reset Button	None	Disabled

Table 10: Ports and Interfaces

4 Roles, Services, and Authentication

4.1 Authentication Methods

The module supports role-based authentication within the module, where all roles must authenticate to the module by providing their authentication data.

The module does not implement a limit on consecutive authentication attempts, as described in section 5.2.2 of SP 800-63B. However, this is mitigated by the two-second delay for failed user password attempts and by a conservative argument about network session rate for the other authenticators. The success probability for random attempts during a one-minute period, as shown in the third column of the below table, shows that the module is well protected against password guessing attacks for all authenticators.

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
Single factor cryptographic software for SMC, Peer NGFW, and Log Server	Authentication for SMC, Peer NGFW, and Log Server	HMAC-SHA-1 (A4793)	Min: 1/(2^112)	Min: 60,000,000/(2^112)
Single factor cryptographic software for SNMP	Authentication for SNMP	HMAC-SHA-1 (A4793)	Min: 1/(2^128)	Min: 60,000,000/(2^128)
Memorized secret for HTTPS and Mobile VPN	Authentication for HTTPS and Mobile VPN	SHA2-512 (A4793)	Min: 1 / (94^10)	Min: 30 / (94^10)
Single factor cryptographic software for IPsec VPN (RSA)	Authentication for IPsec VPN using Digital Signatures	RSA Signature Generation	Min: 1 / (2^112)	Min: 60,000,000 / (2^112)
Single factor cryptographic software for IPsec VPN (ECDSA)	Authentication for IPsec VPN using Digital Signatures	ECDSA Signature Generation	Min: 1 / (2^112)	Min: 60,000,000 / (2^112)

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
Single factor cryptographic software for IPsec VPN (HMAC)	Authentication for IPsec VPN using MAC's	HMAC-SHA-1 (A4793)	Min: 1 / (94^14)	Min: 60,000,000 / (94^14)

Table 11: Authentication Methods

4.2 Roles

The mapping of the cryptographic module's roles services is in the table below:

Name	Type	Operator Type	Authentication Methods
Crypto-Officer	Role	CO	Single factor cryptographic software for SMC, Peer NGFW, and Log Server
User	Role	User	Single factor cryptographic software for SNMP Memorized secret for HTTPS and Mobile VPN Single factor cryptographic software for IPsec VPN (RSA) Single factor cryptographic software for IPsec VPN (ECDSA) Single factor cryptographic software for IPsec VPN (HMAC)

Table 12: Roles

4.3 Approved Services

All services listed in the table below can be accessed in approved mode and when in this mode exclusively use the security functions listed in Cryptographic Algorithms.

- In the 'Access Rights to Keys and/or SSPs' column:
  - G = Generate: The module generates or derives the SSP.
  - R = Read: The module exports the SSP.
  - W = Write: The SSP is imported or updated
  - E = Execute: The module uses the SSP in performing a cryptographic operation.
  - Z = Zeroize: The module zeroizes the SSP.

- In the 'Roles SSP Access' column:

- For a complete description of SSP referenced from the table, see Section “9 Sensitive Security Parameters Management”.

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Export Logs and Monitoring Data Service	Traffic logs and monitoring data are exported to Log Server securely.	Log field - [sendlogd: FIPS: starting in FIPS compliant mode], [entries are received by the log server.]; SMC Monitoring - [data shown in the SMC monitoring window]	TLS protocol messages	TLS protocol messages, Log data Monitoring data, TLS ECDSA Public Key, TLS ECDH Public Key	AES-GCM Encryption/Decryption for TLS CKG DRBG ECDSA Key Generation ECDSA Key Verification ECDSA Signature Generation ECDSA Signature Verification Entropy source HMAC for TLS KAS-ECC Full Public Key Validation TLS Key Agreement using KAS-ECC-SSC	Crypto-Officer - DRBG 'Key' Value: G,W,E - DRBG 'V' Value: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - HTTPS ECDSA Private Key: G,W,E - TLS Authentication Key: G,W,E - TLS ECDH Private Key: G,W,E - TLS ECDH Public Key: G,W,E - TLS ECDSA Private Key: G,W,E - TLS ECDSA Public Key: G,W,E - TLS Encryption Key: G,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS Pre-Master Secret: G,W,E - TLS Trusted Certificates: G,W,E
HTTPS Proxy Service	Sidewinder proxy used for outbound traffic.	Log field - [SSM Proxy; TLS Decrypted=true]	TLS protocol messages	TLS protocol messages, SSM HTTPS DH Public Key, SSM HTTPS ECDH Public Key, SSM Client Protection RSA Public Key, SSM Client, Protection ECDSA Public Key, Status indicator	AES CBC Encryption/Decryption for TLS AES-GCM Encryption/Decryption for TLS CKG DRBG ECDSA Key Generation ECDSA Key Verification ECDSA Signature Generation ECDSA Signature Verification Entropy source HMAC for TLS KAS-ECC Full Public Key Validation KAS-FFC Full Public Key Validation PBKDF RSA Key Generation RSA Signature Generation RSA Signature Verification (FIPS 186-4) RSA Signature Verification	User - DRBG 'Key' Value: G,W,E - DRBG 'V' Value: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - Key Encryption Key: G,W,E - SSM Client Protection ECDSA Private Key: G,W,E - SSM Client Protection ECDSA Public Key: G,W,E - SSM Client Protection RSA Private Key: G,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					(FIPS 186-5) Safe Prime Key Generation Safe Prime Key Verification TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	- SSM Client Protection RSA Public Key: G,W,E - SSM HTTPS Authentication Key: G,W,E - SSM HTTPS DH Private Key: G,W,E - SSM HTTPS DH Public Key: G,R,W,E - SSM HTTPS ECDH Private Key: G,W,E - SSM HTTPS ECDH Public Key: G,R,W,E - SSM HTTPS Encryption Key: G,W,E - SSM HTTPS Master Secret: G,W,E - SSM HTTPS Pre-Master Secret: G,W,E - Trusted Internet

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Certificates: G,W,E
HTTPS User Authentication Service	End user's authentication to the module via web browser.	Log field - [New user has been authorized, User has been reauthorized]	TLS protocol messages, User authentication data	TLS protocol messages, HTTPS DH Public Key, HTTPS ECDH Public Key, HTTPS RSA Public Key, Status indicator	AES CBC Encryption/Decryption for TLS AES-GCM Encryption/Decryption for TLS Authentication for HTTPS and Mobile VPN CKG DRBG ECDSA Key Generation ECDSA Key Verification Entropy source HMAC for TLS RSA Key Generation RSA Signature Generation Safe Prime Key Generation Safe Prime Key Verification TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	User - DRBG 'Key' Value: G,W,E - DRBG 'V' Value: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - HTTPS Authentication Key: G,W,E - HTTPS DH Private Key: G,W,E - HTTPS DH Public Key: G,R,W,E - HTTPS ECDH Private Key: G,W,E - HTTPS ECDH Public Key: G,R,W,E - HTTPS Encryption Key: G,W,E - HTTPS Master Secret: G,W,E - HTTPS Pre-Master

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Secret: G,W,E - HTTPS RSA Private Key: G,W,E - HTTPS RSA Public Key: G,W,E - TLS Master Secret: G,W,E - User Password: E
Initialize module	Set up the module using NGFW Initial Configuration Wizard. The setup process includes mandatory firmware upgrade, applying initial configuration and enabling the Approved Mode of operation.	Log field: "Started in FIPS 140 operating mode."	Firmware image and signature Initial configuration data - TLS Trusted Certificates	Status Indicator	DRBG Firmware integrity ECDSA Signature Verification	Crypto-Officer - Configuration File Protection Key: G,W,E - Configuration File Protection Passphrase: G,W,E
IPsec VPN Service	VPN tunneling clients establish secure	Log field - [IPsec VPN: IPsec SA initiator done], [IPsec	IKE protocol messages, IPSec	IKE protocol messages, IPSec protocol	AES CBC Encryption/Decryption for IPsec/VPN AES-GCM	User - DRBG 'Key' Value: G,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	IPsec VPN connections to the module.	VPN: IPsec SA responder done]	protocol messages	messages, VPN DH Public Key, VPN ECDH Public Key, VPN RSA Public Key, VPN ECDSA Public Key, Status indicator	Encryption/Decryption for IPsec/VPN Authentication for IPsec VPN DRBG ECDSA Key Generation ECDSA Key Verification ECDSA Signature Generation ECDSA Signature Verification Entropy source HMAC for IKE HMAC for IPsec/VPN HMAC for SNMP IPSec Key Agreement using KAS-ECC-SSC IPSec Key Agreement using KAS-FFC-SSC KAS-ECC Full Public Key Validation KAS-FFC Full Public Key Validation KTS for VPN RSA Signature Generation RSA Signature Verification (FIPS 186-4) RSA Signature Verification	- DRBG 'V' Value: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - IKE Authentication Key: G,W,E - IKE Encryption Key: G,W,E - IPsec Authentication Key: G,W,E - IPsec Encryption Key: G,W,E - SKEYID, SKEYID_d: G,W,E - SKEYSEED, SK_d, SK_pi, SK_pr: G,W,E - User Password: E - VPN DH Private Key: G,W,E - VPN DH Public Key: G,W,E - VPN DH Shared

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					(FIPS 186-5) Safe Prime Key Generation Safe Prime Key Verification	Secret: G,W,E - VPN ECDH Private Key: G,W,E - VPN ECDH Public Key: G,W,E - VPN ECDH Shared Secret: G,W,E - VPN ECDSA Private Key: G,W,E - VPN ECDSA Public Key: G,W,E - VPN Key Wrapping Key: G,W,E - VPN Pre-Shared Key: G,W,E - VPN RSA Private Key: G,W,E - VPN RSA Public Key: G,W,E - VPN Trusted Certificates: G,W,E
Key Pair Management Service	SMC using the management communic	Log field - [Private key <filename> has been created]	Certificate data, Issued certificate	Certificate signing request, Status indicator	CKG TLS Key Agreement using KAS-ECC-SSC	Crypto-Officer - Configuration File

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	ation protocol requests engine to generate key pair and certificate signing request.				RSA Key Generation AES-GCM Encryption/Decryption for TLS HMAC for TLS DRBG Entropy source ECDSA Key Generation ECDSA Key Verification ECDSA Signature Generation ECDSA Signature Verification RSA Signature Generation AES Encryption/Decryption for Key Pair Management HMAC for Key Pair Management	Authentication Key: G,W,E - Configuration File Encryption Key: G,W,E - DRBG 'Key' Value: G,W,E - DRBG 'V' Value: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - HTTPS ECDSA Private Key: G,W,E - HTTPS ECDSA Public Key: G,W,E - HTTPS RSA Private Key: G,W,E - HTTPS RSA Public Key: G,W,E - TLS Authentication Key: G,W,E - TLS ECDH Private Key: G,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS ECDH Public Key: G,W,E - TLS ECDSA Private Key: G,W,E - TLS ECDSA Public Key: G,W,E - TLS Encryption Key: G,W,E - TLS Master Secret: G,W,E - TLS Pre-Master Secret: G,W,E - VPN ECDSA Private Key: G,W,E - VPN ECDSA Public Key: G,W,E - VPN RSA Private Key: G,W,E - VPN RSA Public Key: G,W,E
Management Connection Service	SMC establishes secure management connections to the module	Log field: Management: TLS Connection established	TLS protocol messages, Management protocol requests	TLS protocol messages, Management protocol responses, TLS ECDSA Public Key,	AES-GCM Encryption/Decryption for TLS DRBG ECDSA Key Generation ECDSA Key Verification	Crypto-Officer - DRBG 'Key' Value: G,W,E - DRBG 'V' Value:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	over TLS. After initializing the module and initial contact with SMC, all post-installation configuration and modification of initial configuration is secured using TLS connections from SMC.			TLS ECDH Public Key, VPN RSA Public Key, VPN ECDSA, Public Key, HTTPS RSA Public Key	ECDSA Signature Generation ECDSA Signature Verification HMAC for TLS KAS-ECC Full Public Key Validation TLS Key Agreement using KAS-ECC-SSC	G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - HTTPS RSA Public Key: G,R,W,E - TLS Authentication Key: G,W,E - TLS ECDH Private Key: G,W,E - TLS ECDH Public Key: G,R,W,E - TLS ECDSA Public Key: G,R,W,E - TLS Encryption Key: G,W,E - TLS Master Secret: G,W,E - TLS Pre-Master Secret: G,W,E - TLS Trusted Certificates: G,W,E - VPN ECDSA Public Key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						G,R,W,E - VPN RSA Public Key: G,R,W,E
Mobile VPN Service	VPN tunneling clients establish secure IPsec VPN connections to the module.	Log field - [IPsec VPN: Mobile session created],[IPsec VPN: Mobile session closed]	IKE protocol messages, IPSec protocol messages, User authentication data	IKE protocol messages, IPSec protocol messages, VPN DH Public Key, Status indicator, VPN ECDH Public Key, VPN ECDH Public Key, VPN RSA Public Key, VPN ECDSA Public Key, Status indicator	CKG IPSec Key Agreement using KAS-ECC-SSC IPSec Key Agreement using KAS-FFC-SSC DRBG Entropy source ECDSA Key Generation ECDSA Key Verification ECDSA Signature Generation ECDSA Signature Verification Safe Prime Key Generation Safe Prime Key Verification RSA Signature Generation KAS-ECC Full Public Key Validation KAS-FFC Full Public Key Validation Authentication for HTTPS and Mobile VPN	User - DRBG 'Key' Value: G,W,E - DRBG 'V' Value: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - IKE Authentication Key: G,W,E - IKE Encryption Key: G,W,E - IPsec Authentication Key: G,W,E - IPsec Encryption Key: G,W,E - SKEYID, SKEYID_d: G,W,E - SKEYSEED, SK_d, SK_pi, SK_pr: G,W,E - User Password:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						E - VPN DH Private Key: G,W,E - VPN DH Public Key: G,W,E - VPN DH Shared Secret: G,W,E - VPN ECDH Private Key: G,W,E - VPN ECDH Public Key: G,W,E - VPN ECDH Shared Secret: G,W,E - VPN ECDSA Private Key: G,W,E - VPN ECDSA Public Key: G,W,E - VPN Key Wrapping Key: G,W,E - VPN Pre-Shared Key: G,W,E - VPN RSA Private Key: G,W,E - VPN RSA Public Key: G,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- VPN Trusted Certificates: G,W,E
Modify and Apply Configuration	Verify and apply the configuration changes to the modules securely including configuration of client protection and server protection certificate authority and TLS credentials.	Log field - [Inspection: System Policy-Loaded],[Inspection: System Policy-Applied]	Configuration data, VPN Pre-Shared Key, Client Protection CA RSA Public/Private Key, Server Protection RSA Public/Private Key, Server Protection ECDSA Public/Private Key, SNMP Encryption Key, SNMP Authentication Key, Trusted Internet Certificates, VPN Trusted Certificates, Cluster Protocol Key, Key Encryption Passphrase	Status indicator	AES Decryption for Modify and Apply Configuration AES Encryption/Decryption for Configuration File Protection AES-GCM Encryption/Decryption for TLS ECDSA Signature Verification HMAC for Configuration File Authentication HMAC for TLS KBKDF KTS for TLS/IPsec Connections (AES-GCM) PBKDF RSA Signature Verification (FIPS 186-4) RSA Signature Verification (FIPS 186-5) KTS for TLS/IPsec Connections (AES-CBC + HMAC)	Crypto-Officer - Client Protection CA RSA Private Key: G,W,E - Client Protection ECDSA Private Key: G,W,E - Client Protection IM CA ECDSA Private Key: G,W,E - Client Protection IM CA RSA Private Key: G,W,E - Client Protection RSA Private Key: G,W,E - Cluster Protocol Key: G,W,E - Configuration File Authentication Key: G,W,E - Configuration File

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Encryption Key: G,W,E - Key Encryption Key: G,W,E - Key Encryption Passphrase: G,W,E - SNMP Authentication Key: G,W,E - SNMP Encryption Key: G,W,E - VPN Pre-Shared Key: G,W,E
Peer Connection Service	Peer NGFW modules establish secure network connection within a cluster.	Log field - [dsd: FIPS: starting in FIPS compliant mode], [ssd: FIPS: starting in FIPS compliant mode]	TLS protocol messages, Heartbeat State synchronization data, Data synchronization data, State Synchronization Key, IKE Encryption Key, IKE Authentication Key, SKEYID, SKEYID_D, SKEYSEED, SK_D, SK_PI, SK_PR, IPSec	TLS protocol messages, Heartbeat State synchronization data, Data synchronization data, State Synchronization Key, IKE Encryption Key, IKE Authentication Key, SKEYID, SKEYID_D, SKEYSEED, SK_D, SK_PI, SK_PR, IPSec	AES CBC Encryption/Decryption for IPsec/VPN AES-CBC Encryption/Decryption for State Synchronization AES-GCM Encryption/Decryption for IPsec/VPN AES-GCM Encryption/Decryption for TLS CKG DRBG ECDSA Key Generation ECDSA Key Verification ECDSA Signature Generation	Crypto-Officer - Cluster Protocol Key: G,W,E - DRBG 'Key' Value: G,W,E - DRBG 'V' Value: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - HTTPS ECDSA Private Key: G,W,E - HTTPS RSA

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			Encryption Key, IPsec Authentication Key, VPN RSA Public/Private Key, VPN ECDSA Public/Private Key, HTTPS RSA Public/Private Key	Encryption Key, IPsec Authentication Key, VPN RSA Public/Private Key, VPN ECDSA Public/Private Key, HTTPS RSA Public/Private Key, TLS ECDH Public Key, TLS ECDSA Public Key	ECDSA Signature Verification Entropy source HMAC for Peer Heartbeat HMAC for Peer State Synchronization HMAC for TLS IPSec Key Agreement using KAS-ECC-SSC IPSec Key Agreement using KAS-FFC-SSC KAS-ECC Full Public Key Validation KAS-FFC Full Public Key Validation KTS for TLS/IPsec Connections (AES-GCM) Safe Prime Key Generation Safe Prime Key Verification TLS Key Agreement using KAS-ECC-SSC KTS for TLS/IPsec Connections (AES-CBC + HMAC)	Private Key: G,R,W,E - HTTPS RSA Public Key: G,R,W,E - IKE Authentication Key: G,R,W,E - IKE Encryption Key: G,W,E - IPsec Authentication Key: G,R,W,E - IPsec Encryption Key: G,R,W,E - SKEYID, SKEYID_d: G,R,W,E - SKEYSEED, SK_d, SK_pi, SK_pr: G,R,W,E - State Synchronization Key: G,R,W,E - TLS Authentication Key: G,W,E - TLS ECDH Private Key: G,W,E - TLS ECDH Public Key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						G,R,W,E - TLS ECDSA Private Key: G,W,E - TLS ECDSA Public Key: G,W,E - TLS Encryption Key: G,W,E - TLS Master Secret: G,W,E - TLS Pre-Master Secret: G,W,E - TLS Trusted Certificates: G,W,E - VPN DH Private Key: G,R,W,E - VPN ECDH Public Key: G,W,E - VPN ECDSA Private Key: G,R,W,E - VPN ECDSA Public Key: G,R,W,E - VPN RSA Private Key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						G,R,W,E - VPN RSA Public Key: G,R,W,E
Perform Self-Tests	Perform all power-on self-tests.	Console output: FIPS power-up tests succeeded; Log field: Cryptographic self-tests succeeded	None	Self-test result	Firmware integrity ECDSA Signature Verification	Crypto-Officer - Firmware Integrity Check Public Key : E
Show Status	Report the status of the module.	SMC monitoring: interface SMC: Approved mode displayed in SMC monitoring window	None	Module status information	None	Crypto-Officer
Show Versioning Information	Display the module name and version information.	Console output: Forcepoint NGFW version <version>; SMC monitoring interface: version displayed in SMC monitoring window.	None	Module name and version	None	Crypto-Officer
Shut down the module	Terminate module operations in preparation for powering off.	Power LED: A shut down module is indicated by an unlit power LED.	Power control command	Status indicator	None	Crypto-Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
SNMP Monitoring Service	SNMP manager receives network management information and traps.	Log field - [smonitd: FIPS starting in FIPS compliant mode]	Encrypted and authenticated SNMP requests	Encrypted and authenticated SNMP responses, Status indicator	AES-CBC Encryption/Decryption for SNMP HMAC for SNMP	User - SNMP Authentication Key: E - SNMP Encryption Key: E
TLS Inspection Service	Perform TLS inspection on HTTPS network traffic.	Log field - [Inspection; TLS Decrypted=true]	TLS protocol messages	TLS protocol messages, TLS ECDSA Public Key, Inspection DH Public Key, Inspection ECDH Public Key, Client Protection RSA Public Key, Client Protection ECDSA Public Key, Client Protection IM CA RSA Public Key, Client Protection IM CA ECDSA Public Key, Server Protection RSA Public Key, Server Protection ECDSA Public Key,	AES CBC Encryption/Decryption for TLS AES-GCM Encryption/Decryption for TLS CKG DRBG ECDSA Key Generation ECDSA Key Verification ECDSA Signature Generation ECDSA Signature Verification Entropy source HMAC for TLS KAS-ECC Full Public Key Validation KAS-FFC Full Public Key Validation RSA Key Generation RSA Signature Generation RSA Signature Verification (FIPS 186-4) RSA Signature Verification	User - Client Protection CA RSA Private Key: G,W,E - Client Protection CA RSA Public Key: G,W,E - Client Protection ECDSA Private Key: G,W,E - Client Protection ECDSA Public Key: G,R,W,E - Client Protection IM CA ECDSA Private Key: G,W,E - Client Protection IM CA ECDSA Public Key: G,W,E - Client Protection

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
				Status indicator	(FIPS 186-5) Safe Prime Key Generation Safe Prime Key Verification TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	IM CA RSA Private Key: G,W,E - Client Protection IM CA RSA Public Key: G,W,E - Client Protection RSA Private Key: G,W,E - Client Protection RSA Public Key: G,R,W,E - DRBG 'Key' Value: G,W,E - DRBG 'V' Value: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - Inspection Authentication Key: G,W,E - Inspection DH Private Key: G,W,E - Inspection DH Public

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Key: G,R,W,E - Inspection ECDH Private Key: G,W,E - Inspection ECDH Public Key: G,R,W,E - Inspection Encryption Key: G,W,E - Inspection Master Secret: G,W,E - Inspection Pre-Master Secret: G,W,E - Server Protection ECDSA Private Key: G,W,E - Server Protection ECDSA Public Key: G,W,E - Server Protection RSA Private Key: G,W,E - Server Protection RSA Public

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Key: G,R,W,E - SSM Client Protection ECDSA Public Key: G,R,W,E - SSM Client Protection RSA Public Key: G,R,W,E - Trusted Internet Certificates: G,W,E
User Management Service	SMC enters the user password hashes using LDAPS.	Log field - [slapd: FIPS: running in FIPS compliant mode]	TLS protocol messages, LDAP protocol messages, User Password	TLS protocol messages, LDAP protocol messages, TLS ECDSA Public Key, TLS ECDH Public Key, Status indicator	AES-GCM Encryption/Decryption for TLS CKG DRBG ECDSA Key Generation ECDSA Key Verification ECDSA Signature Generation ECDSA Signature Verification Entropy source HMAC for TLS KAS-ECC Full Public Key Validation TLS Key Agreement using KAS-ECC-SSC	Crypto-Officer - DRBG 'Key' Value: G,W,E - DRBG 'V' Value: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - HTTPS ECDSA Private Key: G,W,E - HTTPS ECDSA Public Key: G,W,E - TLS Authentica

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						tion Key: G,W,E - TLS ECDH Private Key: G,W,E - TLS ECDH Public Key: G,W,E - TLS ECDSA Private Key: G,W,E - TLS ECDSA Public Key: G,W,E - TLS Encryption Key: G,W,E - TLS Master Secret: G,W,E - TLS Pre-Master Secret: G,W,E - User Password: E
Zeroize keys	The module will overwrite all CSPs. Zeroization of keys can be invoked by performing a factory reset exercising commands	Console output: System zeroization complete following reboot. Factory default settings restored.	Power control command Keyboard input	Status indicator	None	Crypto-Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	. The zeroization occurs while the module is still in the Approved mode, and the module is restored to a factory state.					

Table 13: Approved Services

4.4 Non-Approved Services

N/A for this module.

4.5 External Software/Firmware Loaded

Not Applicable, the module does not load any new firmware while in the compliant state.

4.6 Bypass Actions and Status

The module operates in an alternating bypass mode according to the policies set. The enabling and disabling of the bypass capability is performed via 'Modify and apply configuration' service allocated to the CO role. The module implements the following forms of alternating bypass:

VPN network traffic:

For policy-based VPN traffic, the module operates with bypass deactivated if the module action is set to IPsec VPN, where the module is operating to provide VPN service for the specified source/destination addresses. The module will encrypt/decrypt network traffic according to the policy. The module operates with bypass activated if the module action is set to allow in Access rules for network traffic, where the module is accepting/sending plaintext data for the specified source/destination addresses. For route-based VPN traffic, the module operates with bypass deactivated when network traffic is routed to module interfaces that are designated as endpoints for a VPN tunnel and is sent into the VPN tunnel. If Access rules allow the traffic, traffic is automatically sent through the tunnel to the endpoint. The module operates with bypass activated when network

traffic is routed to module interfaces that accept plaintext data. Based on the Access rule (allow/discard), the traffic is either forwarded to the endpoint or dropped. In both cases, to activate the bypass feature, two independent actions must be taken by a CO. The CO must create the firewall policy allowing the bypass feature and apply the policy to the module to enable it.

Firewall network traffic:

The default action for network traffic in firewall Access rules is discard. For firewall traffic, the module operates with bypass deactivated if the traffic from the endpoint is sent/received using HTTPS, and the module action is set to allow. If traffic from the endpoint is passed directly to the module using HTTP, and the module action is set to allow, then the module is operating with bypass activated. For incoming traffic, if the HTTPS option is selected, the module connections with the endpoint are encrypted using TLS (bypass deactivated). If the HTTP option is selected, the module accepts connections in plaintext (bypass activated). For Outgoing traffic, If HTTPS is selected, web traffic will be re-encrypted using TLS (bypass deactivated). If HTTP is configured, web traffic is sent in plaintext (bypass activated). Two independent actions must be taken by a CO. The CO must create the firewall policy allowing bypass and apply to the module to enable it.

The rules in the policy that is currently applied to the module specify whether the module allows the encrypted or plaintext traffic. The status information for the bypass activation and deactivation can be viewed via established management connection from SMC as indicated below:

- **Bypass** – When bypass is activated, the Situation field in the Logs view shows “Connection Allowed” and the TLS decrypted field in the Connections view is blank.
- **IPSEC VPN/HTTPS** – The Situation field in the Logs view indicates the respective operations performed by these services. For example, “IPsec-SA-Responder-Done”.
- **TLS Inspection** – For this service the Situation field in the Logs view shows “Connection_Allowed” and the TLS decrypted field in Connections view is "true".

4.7 Cryptographic Output Actions and Status

The Export Logs and Monitoring Data Service and the Peer Connection Service are self-initiated cryptographic output capabilities supported by the module. In both cases, these services are triggered by the module itself without a specific request to perform the service.

- The **Export Logs and Monitoring Data Service** is enabled on the first policy push from the SMC where the Log Server address is specified. This is configured through two independent steps: adding the Log Server and activating the policy.

- The **Peer Connection Service** is enabled when the module is joined to a cluster. While the module is in a cluster, this communication happens, and it stops when module is removed from the cluster. This is also configured through two independent steps:

    - Installing the new engine (module) and performing the initial configuration after adding the engine to the Firewall Cluster element’s properties and defining the node-specific IP addresses of the new node.

- Refreshing the security policy of the Firewall Cluster.

The Crypto officer can make usage of the module's Show status service to determine if the self-initiated cryptographic output capability is active by observing the 'Status' tab on the SMC Web GUI:

5 Software/Firmware Security

5.1 Integrity Techniques

The Forcepoint Next Generation Firewall for Desktop Appliances firmware integrity is checked on startup as described in Section "10 Self-Tests". The module runs the self-test functions to check the firmware integrity as well as the cryptographic algorithms used. Any failures during these tests will result in a module halt in which an error message is output, the module reboots and data output is inhibited.

The images are stored as signed binary (6.10.13.26655.fips.2.bin) using the "Firmware Integrity Check Public Key" which uses ECDSA SigVer (A4793) with P-521 and SHA2-512.

5.2 Initiate on Demand

The operator can trigger an on-demand check of the module firmware by rebooting the module.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: The module supports a Limited as defined in ISO/IEC 19790:2012, as the module is not designed to accept firmware changes in the Approved mode of operation.

7 Physical Security

Mechanism	Inspection Frequency	Inspection Guidance
N/A	N/A	N/A

Table 14: Mechanisms and Actions Required

The module's embodiment type is Multi-chip Standalone.

8 Non-Invasive Security

N/A: Section “8 Non-Invasive Security” is Not-Applicable as there are currently no requirements in SP 800-140F.

9 Sensitive Security Parameters Management

9.1 Storage Areas

The table below lists sensitive security parameters (SSPs) storage areas for this module. Section “9.4 SSPs” below selects from the storage areas listed and specifies the appropriate parameter in the “Storage” column if applicable to a specific SSP.

Storage Area Name	Description	Persistence Type
Disk - Encrypted	Files on the file system, stored on a solid state disk	Static
Disk - Plaintext	Files on the file system, stored on a solid state disk	Static
SDRAM	Memory area reserved for the process	Dynamic

Table 15: Storage Areas

9.2 SSP Input-Output Methods

The table below lists SSP input and output methods for this module. Section “9.4 SSPs” below selects from the input and output methods listed and specifies the appropriate parameter in the “Inputs/Outputs” column if applicable to a specific SSP.

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Encrypted Input via Ethernet (AES CBC + HMAC)	Outside Cryptographic Boundary	SDRAM	Encrypted	Automated	Electronic	KTS for TLS/IPsec Connections (AES-GCM)
Encrypted Input via Ethernet (AES GCM)	Outside Cryptographic Boundary	SDRAM	Encrypted	Automated	Electronic	KTS for TLS/IPsec Connections (AES-GCM)
Encrypted Input via Ethernet (AES-KWP)	Outside Cryptographic Boundary	SDRAM	Encrypted	Automated	Electronic	KTS for VPN

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Encrypted Output via Ethernet (AES CBC + HMAC)	SDRAM	Outside Cryptographic Boundary	Encrypted	Automated	Electronic	KTS for TLS/IPsec Connections (AES-GCM)
Encrypted Output via Ethernet (AES GCM)	SDRAM	Outside Cryptographic Boundary	Encrypted	Automated	Electronic	KTS for TLS/IPsec Connections (AES-GCM)
Encrypted Output via Ethernet (AES-KWP)	SDRAM	Outside Cryptographic Boundary	Encrypted	Automated	Electronic	KTS for VPN
Input Hashed and Salted	Outside Cryptographic Boundary	Disk	Encrypted	Automated	Electronic	Authentication for HTTPS and Mobile VPN
Plaintext Input via Ethernet	Outside Cryptographic Boundary	Disk	Plaintext	Automated	Electronic	
Plaintext Output via Ethernet	Disk	Outside Cryptographic Boundary	Plaintext	Automated	Electronic	

Table 16: SSP Input-Output Methods

9.3 SSP Zeroization Methods

The table below lists SSP zeroization methods for this module. Section “9.4 SSPs” below selects from the zeroization methods listed and specifies the appropriate parameter in the “Zeroization” column if applicable to a specific SSP.

Zeroization Method	Description	Rationale	Operator Initiation
Factory Reset	All SSPs stored in memory are zeroized by overwriting them with ‘0’ during shutdown. All SSPs stored on disk are zeroized by overwriting them	After overwriting the original SSP data is no longer available	1. Reboot the appliance and select System restore options from the boot menu. NGFW System Restore starts. 2. Enter 2 for Advanced data removal options.

Zeroization Method	Description	Rationale	Operator Initiation
	an operator-specified number of times during factory reset		3. Enter one of the following options: - 1 for 1 pass overwrite - 8 for a Custom number of overwrite passes
Power-Cycling	All SSPs stored in memory are zeroized by overwriting them with '0' during shutdown	After overwriting the original SSP data is no longer available	Operator presses the power button to shut down and restart the appliance

Table 17: SSP Zeroization Methods

9.4 SSPs

The following table lists Sensitive Security Parameters (SSP) used to perform approved security functions supported by the cryptographic module.

The following notes should be observed when reading the table:

- When reading the 'strength' column, the listed security strength is calculated using methods in FIPS 140-3 IG D.B, 'Strength of SSP Establishment Methods'.
- When reading the 'Security Function and Cert Number' column, this is the security function that will consume the SSP.
- When reading the 'Use and Related Keys' column, this will contain the other SSPs that are either established via the SSP, other SSPs that are used to establish the SSP, or if it is a key pair the associated public or private component will be listed as well.

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Client Protection CA RSA Private Key	Private signature key used in TLS inspection CA.	2048-4096 bits - 112-150 bits	Private Key - CSP	CKG RSA Key Generation		RSA Signature Generation
Client Protection CA RSA Public Key	Public signature key used in TLS inspection CA.	2048-4096 bits - 112-150 bits	Public Key - PSP	CKG RSA Key Generation		RSA Signature Verification (FIPS 186-4) RSA Signature Verification (FIPS 186-5)
Client Protection ECDSA Private Key	Private authentication key used in TLS inspection.	224-521 bits - 112-256 bits	Private Key - CSP	CKG ECDSA Key Generation		ECDSA Signature Generation

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Client Protection ECDSA Public Key	Public authentication key used in TLS inspection.	224-521 bits - 112-256 bits	Public Key - PSP	CKG ECDSA Key Generation		ECDSA Signature Verification
Client Protection IM CA ECDSA Private Key	Private authentication key used in TLS inspection.	224-521 bits - 112-256 bits	Private Key - CSP	CKG ECDSA Key Generation		ECDSA Signature Generation
Client Protection IM CA ECDSA Public Key	Public authentication key used in TLS inspection.	224-521 bits - 112-256 bits	Public Key - PSP	CKG ECDSA Key Generation		ECDSA Signature Verification
Client Protection IM CA RSA Private Key	Private authentication key used in TLS inspection.	2048-4096 bits - 112-150 bits	Private Key - CSP	CKG RSA Key Generation		ECDSA Signature Generation
Client Protection IM CA RSA Public Key	Public authentication key used in TLS inspection.	2048-4096 bits - 112-150 bits	Public Key - PSP	CKG RSA Key Generation		RSA Signature Verification (FIPS 186-4) RSA Signature Verification (FIPS 186-5)
Client Protection RSA Private Key	Private authentication key used in TLS inspection.	2048-4096 bits - 112-150 bits	Private Key - CSP	CKG RSA Key Generation		RSA Signature Generation
Client Protection RSA Public Key	Public authentication key used in TLS inspection.	2048-4096 bits - 112-150 bits	Public Key - PSP	CKG RSA Key Generation		RSA Signature Verification (FIPS 186-4) RSA Signature Verification (FIPS 186-5)
Cluster Protocol Key	Used for authentication within the cluster protocol.	256 bits - 256 bits	Authentication key - CSP			HMAC for Peer State Synchronization

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Configuration File Authentication Key	Used to authenticate configuration files on disk.	256 bits - 256 bits	Authentication Key - CSP	KBKDF		HMAC for Configuration File Authentication
Configuration File Encryption Key	Used to encrypt configuration files on disk.	256 bits - 256 bits	Symmetric Key - CSP	KBKDF		AES Encryption/Decryption for Configuration File Protection
Configuration File Protection Key	Used to derive the configuration file encryption and authentication keys.	256 bits - 256 bits	Symmetric Key - CSP	CKG		KBKDF
Configuration File Protection Passphrase	Used to derive the key pair obfuscation and integrity protection keys.	256 bits - 256 bits	Symmetric Key - CSP	CKG		PBKDF
DRBG 'Key' Value	DRBG State Value	256 bits - 256 bits	DRBG State - CSP	DRBG		Entropy source
DRBG 'V' Value	DRBG State Value	128 bits - 128 bits	DRBG State - CSP	DRBG		Entropy source
DRBG Entropy Input	Entropy input for DRBG	512 bits - 512 bits	Entropy Input - CSP	Entropy source		DRBG
DRBG Seed	Seeding material for the DRBG	512 bits - 512 bits	DRBG Seed - CSP	Entropy source		DRBG
Firmware Integrity Check Public Key	Used for firmware integrity check.	P-521 curve - 256	Integrity Public Key - Neither		Firmware integrity	Firmware integrity
HTTPS Authentication Key	Authentication key used in TLS.	256 bits - 256 bits	Authentication Key - CSP		TLS Key Agreement using	HMAC for TLS

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
					KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	
HTTPS DH Private Key	Private ephemeral key agreement key used in TLS.	2048-8192 bits - 112-202 bits	Private Key - CSP	Safe Prime Key Generation		TLS Key Agreement using KAS-FFC-SSC
HTTPS DH Public Key	Public ephemeral key agreement key used in TLS.	2048-8192 bits - 112-202 bits	Public Key - PSP	Safe Prime Key Generation		TLS Key Agreement using KAS-FFC-SSC
HTTPS ECDH Private Key	Private ephemeral key agreement key used in TLS.	224-521 bits - 112-256 bits	Private Key - CSP	ECDSA Key Generation		TLS Key Agreement using KAS-ECC-SSC
HTTPS ECDH Public Key	Public ephemeral key agreement key used in TLS.	224-521 bits - 112-256 bits	Public Key - PSP	ECDSA Key Generation		TLS Key Agreement using KAS-ECC-SSC
HTTPS ECDSA Private Key	Private key used to authenticate the server during HTTPS connections via digital signatures.	224-521 bits - 112-256 bits	Private Key - CSP	CKG ECDSA Key Generation		ECDSA Signature Generation
HTTPS ECDSA Public Key	Public key used to authenticate the server	224-521 bits - 112-256 bits	Public Key - CSP	CKG ECDSA Key		ECDSA Signature Generation

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	during HTTPS connections via digital signatures.			Generation		
HTTPS Encryption Key	Data encryption key used in TLS.	128, 256 bits - 128, 256 bits	Symmetric Key - CSP		TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	AES CBC Encryption/Decryption for TLS AES-GCM Encryption/Decryption for TLS
HTTPS Master Secret	Value calculated during TLS handshake.	384 bits - 384 bits	Master Secret - CSP		TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	HMAC for TLS
HTTPS Pre-Master Secret	Shared secret generated or established for a TLS session.	KAS-ECC (224-521 bits) / KAS-FFC (2048-8192 bits) - 112-256 bits / 112-202 bits	Shared Secret - CSP		TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC
HTTPS RSA Private Key	Private authentication key used in HTTPS user authentication.	2048-4096 bits - 112-150 bits	Private Key - CSP	CKG RSA Key Generation		RSA Signature Generation

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
HTTPS RSA Public Key	Public authentication key used in HTTPS user authentication.	2048-4096 bits - 112-150 bits	Public Key - PSP	CKG RSA Key Generation		RSA Signature Verification (FIPS 186-4) RSA Signature Verification (FIPS 186-5)
IKE Authentication Key	Authentication key used in IKE negotiations.	128-256 bits - 112-256 bits	Symmetric Key - CSP		IPSec Key Agreement using KAS-ECC-SSC IPSec Key Agreement using KAS-FFC-SSC	HMAC for IPsec/VPN KTS for VPN
IKE Encryption Key	Data encryption key used in IKE negotiations.	128, 256 bits - 128, 256 bits	Symmetric Key - CSP		IPSec Key Agreement using KAS-ECC-SSC IPSec Key Agreement using KAS-FFC-SSC	AES CBC Encryption/Decryption for IPsec/VPN AES-GCM Encryption/Decryption for IPsec/VPN KTS for VPN
Inspection Authentication Key	Authentication key used in TLS inspection.	256 bits - 256 bits	Authentication Key - CSP		HMAC for TLS TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	HMAC for TLS
Inspection DH Private Key	Private ephemeral key agreement key used in	2048-8192 bits - 112-202 bits	Private Key - CSP	Safe Prime Key Generation		TLS Key Agreement using KAS-FFC-SSC

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	TLS inspection.					
Inspection DH Public Key	Public ephemeral key agreement key used in TLS inspection.	2048-8192 bits - 112-202 bits	Public Key - PSP	Safe Prime Key Generation		TLS Key Agreement using KAS-FFC-SSC
Inspection ECDH Private Key	Private ephemeral key agreement key used in TLS inspection.	224-521 bits - 112-256 bits	Private Key - CSP	ECDSA Key Generation		TLS Key Agreement using KAS-ECC-SSC
Inspection ECDH Public Key	Public ephemeral key agreement key used in TLS inspection.	224-521 bits - 112-256 bits	Public Key - PSP	ECDSA Key Generation		TLS Key Agreement using KAS-ECC-SSC
Inspection Encryption Key	Data encryption key used in TLS inspection.	128, 256 bits - 128, 256 bits	Symmetric Key - CSP		TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	AES CBC Encryption/Decryption for TLS AES-GCM Encryption/Decryption for TLS
Inspection Master Secret	Value calculated during TLS inspection.	384 bits - 384 bits	Master Secret - CSP		TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	HMAC for TLS

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Inspection Pre-Master Secret	Shared secret generated or established for TLS inspection.	KAS-ECC (224-521 bits) / KAS-FFC (2048-8192 bits) - 112-256 bits / 112-202 bits	Pre-Master Secret - CSP		TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC
IPsec Authentication Key	Authentication key used in IPsec negotiations.	128-256 bits - 128-256 bits	Authentication Key - CSP		IPSec Key Agreement using KAS-ECC-SSC IPSec Key Agreement using KAS-FFC-SSC	HMAC for IPsec/VPN KTS for VPN
IPsec Encryption Key	Data encryption key used in IPsec negotiations.	128, 256 bits - 128, 256 bits	Symmetric Key - CSP		IPSec Key Agreement using KAS-ECC-SSC IPSec Key Agreement using KAS-FFC-SSC	AES CBC Encryption/Decryption for IPsec/VPN AES-GCM Encryption/Decryption for IPsec/VPN KTS for VPN
Key Encryption Key	Used to encrypt/decrypt private keys stored on disk.	128 bits - 128 bits	Symmetric Key - CSP	PBKDF		AES Encryption/Decryption for Key Pair Management
Key Encryption Passphrase	Derive the key encryption key used to protect private keys	256 bits - 256 bits	Symmetric Key - CSP			PBKDF

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	stored on disk.					
Server Protection ECDSA Private Key	Private authentication key used in TLS inspection.	224-521 bits - 112-256 bits	Private Key - CSP	CKG ECDSA Key Generation		ECDSA Signature Generation
Server Protection ECDSA Public Key	Public authentication key used in TLS inspection.	224-521 bits - 112-256 bits	Public Key - PSP	CKG ECDSA Key Generation		ECDSA Signature Verification
Server Protection RSA Private Key	Private authentication key used in TLS inspection.	2048-4096 bits - 112-150 bits	Private Key - CSP	CKG RSA Key Generation		RSA Signature Generation
Server Protection RSA Public Key	Public authentication key used in TLS inspection.	2048-4096 bits - 112-150 bits	Public Key - PSP	CKG RSA Key Generation		RSA Signature Verification (FIPS 186-4) RSA Signature Verification (FIPS 186-5)
SKEYID, SKEYID_d	Values calculated during IKE v1 negotiation.	112-256 bits - 112-256 bits	Symmetric Key Material - CSP		IPSec Key Agreement using KAS-ECC-SSC IPSec Key Agreement using KAS-FFC-SSC	HMAC for Peer Heartbeat HMAC for Peer State Synchronization
SKEYSEED, SK_d, SK_pi, SK_pr	Values calculated during IKEv2 negotiation.	112-256 bits - 112-256 bits	Symmetric Key Material - CSP		IPSec Key Agreement using KAS-ECC-SSC IPSec Key Agreement using KAS-FFC-SSC	HMAC for Peer Heartbeat HMAC for Peer State Synchronization

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
SNMP Authentication Key	Authentication key used in SNMPv3.	160 bits - 160 bits	Authentication Key - CSP			HMAC for SNMP
SNMP Encryption Key	Data encryption key used in SNMPv3.	128, 256 bits - 128, 256 bits	Symmetric Key - CSP			AES-CBC Encryption/Decryption for SNMP
SSM Client Protection ECDSA Private Key	Used to identify the module in the SSM Proxy Service.	224-521 bits - 112-256 bits	Private key - CSP	ECDSA Key Generation		ECDSA Signature Generation
SSM Client Protection ECDSA Public Key	Used to identify the module in the SSM Proxy Service.	224-521 bits - 112-256 bits	Public Key - PSP	ECDSA Key Generation		ECDSA Signature Verification
SSM Client Protection RSA Private Key	Used to identify the module in the SSM Proxy Service.	2048-4096 bits - 112-150 bits	Private key - CSP	RSA Key Generation		RSA Signature Generation
SSM Client Protection RSA Public Key	Used to identify the module in the SSM Proxy Service.	2048-4096 bits - 112-150 bits	Public Key - PSP	RSA Key Generation		RSA Signature Verification (FIPS 186-4) RSA Signature Verification (FIPS 186-5)
SSM HTTPS Authentication Key	Authentication key used in HTTPS inspection.	256 bits - 256 bits	Authentication Key - CSP		HMAC for TLS TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	HMAC for TLS

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
SSM HTTPS DH Private Key	Private ephemeral key agreement key used in HTTPS inspection.	2048-8192 bits - 112-202 bits	Private key - CSP	Safe Prime Key Generation		TLS Key Agreement using KAS-FFC-SSC
SSM HTTPS DH Public Key	Public ephemeral key agreement key used in HTTPS inspection.	2048-8192 bits - 112-202 bits	Public Key - PSP	Safe Prime Key Generation		TLS Key Agreement using KAS-FFC-SSC
SSM HTTPS ECDH Private Key	Private ephemeral key agreement key used in HTTPS inspection.	224-521 bits - 112-256 bits	Private Key - CSP	ECDSA Key Generation		TLS Key Agreement using KAS-ECC-SSC
SSM HTTPS ECDH Public Key	Public ephemeral key agreement key used in HTTPS inspection.	224-521 bits - 112-256 bits	Public Key - PSP	ECDSA Key Generation		TLS Key Agreement using KAS-ECC-SSC
SSM HTTPS Encryption Key	Data encryption key used in HTTPS inspection.	128, 256 bits - 128, 256 bits	Symmetric Key - CSP		TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	AES CBC Encryption/Decryption for TLS AES-GCM Encryption/Decryption for TLS
SSM HTTPS Master Secret	Value calculated during HTTPS inspection.	384 bits - 384 bits	Master Secret - CSP		TLS Key Agreement using KAS-ECC-SSC	HMAC for TLS

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
					TLS Key Agreement using KAS-FFC-SSC	
SSM HTTPS Pre-Master Secret	Shared secret generated or established for HTTPS inspection.	KAS-ECC (224-521 bits) / KAS-FFC (2048-8192 bits) - 112-256 bits / 112-202 bits	Pre-Master Secret - CSP		TLS Key Agreement using KAS-ECC-SSC	TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC
State Synchronization Key	Used for encryption and authentication in the state synchronization protocol.	128 bits - 128 bits	Symmetric Key - CSP	CKG		AES-CBC Encryption/Decryption for State Synchronization HMAC for Peer State Synchronization
TLS Authentication Key	Authentication key used in TLS.	256 bits - 256 bits	Authentication Key - CSP		TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	HMAC for TLS
TLS ECDH Private Key	Private ephemeral key agreement key used in TLS.	224-521 bits - 112-256 bits	Private Key - CSP	CKG ECDSA Key Generation		TLS Key Agreement using KAS-ECC-SSC
TLS ECDH Public Key	Public ephemeral key	224-521 bits -	Public Key - PSP	CKG ECDSA Key		TLS Key Agreement using KAS-ECC-SSC

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	agreement key used in TLS.	112-256 bits		Generation		
TLS ECDSA Private Key	Private key used in TLS signature.	224-521 bits - 112-256 bits	Private Key - CSP	CKG ECDSA Key Generation		ECDSA Signature Generation
TLS ECDSA Public Key	Public key used in TLS signature.	224-521 bits - 112-256 bits	Public Key - PSP	CKG ECDSA Key Generation		ECDSA Signature Verification
TLS Encryption Key	Data encryption key used in TLS	256 bits - 256 bits	Symmetric Key - CSP		TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	AES CBC Encryption/Decryption for TLS AES-GCM Encryption/Decryption for TLS
TLS Master Secret	Value calculated during TLS handshake.	112-256 bits - 112-256 bits	Master Secret - CSP		TLS Key Agreement using KAS-ECC-SSC TLS Key Agreement using KAS-FFC-SSC	TLS v1.2 KDF RFC7627 (A4793)
TLS Pre-Master Secret	Shared secret generated or established for a TLS session.	112-256 bits - 112-256 bits	Pre-Master Secret - CSP		TLS Key Agreement using KAS-ECC-SSC	TLS v1.2 KDF RFC7627 (A4793)
TLS Trusted Certificates	Trusted certificates for use in TLS	112-256 bits - 112-256 bits	Certificates - PSP			ECDSA Signature Verification RSA Signature Verification (FIPS

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
						186-4) RSA Signature Verification (FIPS 186-5)
Trusted Internet Certificates	Trusted certificates for authenticating internet servers	112-256 bits - 112-256 bits	Certificates - PSP			ECDSA Signature Verification RSA Signature Verification (FIPS 186-4) RSA Signature Verification (FIPS 186-5)
User Password	Identify users in HTTPS authentication and Mobile VPN.	10 characters - N/A	Password - CSP			Authentication for HTTPS and Mobile VPN
VPN DH Private Key	Private ephemeral key agreement key used in IKE.	2048-8192 bits - 112-202 bits	Private Key - CSP	Safe Prime Key Generation		IPSec Key Agreement using KAS-FFC-SSC
VPN DH Public Key	Public ephemeral key agreement key used in IKE.	2048-8192 bits - 112-202 bits	Public Key - PSP	Safe Prime Key Generation		IPSec Key Agreement using KAS-FFC-SSC
VPN DH Shared Secret	Diffie-Hellman shared secret in IKE	112-202 bits - 112-202 bits	Shared Secret - CSP		IPSec Key Agreement using KAS-FFC-SSC Safe Prime Key Generation Safe Prime Key	

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
					Verification	
VPN ECDH Private Key	Private ephemeral key agreement key used in IKE.	224-521 bits - 112-256 bits	Private Key - CSP	ECDSA Key Generation		IPSec Key Agreement using KAS-ECC-SSC
VPN ECDH Public Key	Public ephemeral key agreement key used in IKE.	224-521 bits - 112-256 bits	Public Key - PSP	ECDSA Key Generation		IPSec Key Agreement using KAS-ECC-SSC
VPN ECDH Shared Secret	Elliptical curve Diffie-Hellman shared secret in IKE	224-521 bits - 112-256 bits	Shared Secret - CSP		IPSec Key Agreement using KAS-ECC-SSC ECDSA Key Generation	
VPN ECDSA Private Key	Private authentication key used in IKE.	224-521 bits - 112-256 bits	Private Key - CSP	ECDSA Key Generation		ECDSA Signature Generation
VPN ECDSA Public Key	Public authentication key used in IKE.	224-521 bits - 112-256 bits	Authentication Key - PSP	ECDSA Key Generation		ECDSA Signature Verification
VPN Key Wrapping Key	IKE and IPsec key and key wrapping material	256 bits - 256 bits	Wrapping Key - CSP		KBKDF	KTS for VPN
VPN Pre-Shared Key	Shared secret used in IKE.	112-256 bits - 112-256 bits	Shared Secret - CSP			KDF IKEv1 (A4793) KDF IKEv2 (A4793)
VPN RSA Private Key	Private authentication key used in IKE.	2048-4096 bits -	Private Key - CSP	RSA Key Generation		KTS for TLS/IPsec Connections (AES-GCM) RSA Signature

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		112-150 bits				Generation KTS for TLS/IPsec Connections (AES-CBC + HMAC)
VPN RSA Public Key	Public authentication key used in IKE.	2048-4096 bits - 112-150 bits	Public Key - PSP	RSA Key Generation		RSA Signature Verification (FIPS 186-4) RSA Signature Verification (FIPS 186-5)
VPN Trusted Certificates	Trusted certificates for use in VPNs	112-256 bits - 112-256 bits	Certificates - PSP			ECDSA Signature Verification RSA Signature Verification (FIPS 186-4) RSA Signature Verification (FIPS 186-5)

Table 18: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Client Protection CA RSA Private Key	Encrypted Input via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC)	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	Client Protection CA RSA Public Key:Paired With
Client Protection CA RSA Public Key	Encrypted Input via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC)	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	Client Protection CA RSA Private Key:Paired With
Client Protection		SDRAM :Plaintext	Until Zeroized	Factory Reset	Client Protection ECDSA Public Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
ECDSA Private Key				Power-Cycling	
Client Protection ECDSA Public Key	Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Client Protection ECDSA Private Key:Paired With
Client Protection IM CA ECDSA Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Client Protection IM CA ECDSA Public Key:Paired With
Client Protection IM CA ECDSA Public Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Client Protection IM CA ECDSA Private Key:Paired With
Client Protection IM CA RSA Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Client Protection IM CA RSA Public Key:Paired With
Client Protection IM CA RSA Public Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Client Protection IM CA RSA Private Key:Paired With
Client Protection RSA Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Client Protection RSA Public Key:Paired With
Client Protection RSA Public Key	Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Client Protection RSA Private Key:Paired With
Cluster Protocol Key	Encrypted Input via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC)	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	
Configuration File Authentication Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Configuration File Protection Key:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Configuration File Encryption Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Configuration File Protection Key:Derived From
Configuration File Protection Key		Disk - Plaintext:Plaintext	Until Zeroized	Factory Reset Power-Cycling	Configuration File Encryption Key:Derived From Configuration File Authentication Key:Derived From
Configuration File Protection Passphrase		Disk - Plaintext:Plaintext	Until Zeroized	Factory Reset Power-Cycling	
DRBG 'Key' Value		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	DRBG 'V' Value:Paired With DRBG Seed:Paired With DRBG Entropy Input:Paired With
DRBG 'V' Value		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	DRBG Entropy Input:Paired With DRBG Seed:Paired With
DRBG Entropy Input		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	DRBG 'Key' Value:Paired With DRBG 'V' Value:Paired With
DRBG Seed		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	DRBG Entropy Input:Paired With DRBG 'Key' Value:Paired With DRBG 'V' Value:Paired With
Firmware Integrity Check Public Key		Disk - Plaintext:Plaintext	Until Zeroized	Factory Reset Power-Cycling	
HTTPS Authentication Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	HTTPS Master Secret:Derived From
HTTPS DH Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset	HTTPS Pre-Master Secret:Derived

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				Power-Cycling	From HTTPS DH Public Key:Paired With
HTTPS DH Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	HTTPS DH Private Key:Paired With HTTPS Pre-master Secret:Derived From
HTTPS ECDH Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	HTTPS Pre-Master Secret:Derived From HTTPS ECDH Public Key:Paired With
HTTPS ECDH Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	HTTPS ECDH Private Key:Paired With HTTPS Pre-master Secret:Derived From
HTTPS ECDSA Private Key		Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	HTTPS ECDSA Public Key:Paired With
HTTPS ECDSA Public Key	Plaintext Output via Ethernet	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	HTTPS ECDSA Private Key:Paired With
HTTPS Encryption Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	HTTPS Master Secret:Derived From
HTTPS Master Secret		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	HTTPS Pre-Master Secret:Derived From HTTPS Encryption Key:Derived From HTTPS Authentication Key:Derived From
HTTPS Pre-Master Secret		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	HTTPS DH Private Key:Derived From HTTPS ECDH Private

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Key:Derived From HTTPS ECDH Public Key:Used With HTTPS Master Secret:Used With
HTTPS RSA Private Key	Encrypted Input via Ethernet (AES GCM) Encrypted Output via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC) Encrypted Output via Ethernet (AES CBC + HMAC)	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	HTTPS RSA Public Key:Paired With
HTTPS RSA Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	Disk - Plaintext:Plaintext	Until Zeroized	Factory Reset Power-Cycling	HTTPS RSA Private Key:Paired With
IKE Authentication Key	Encrypted Input via Ethernet (AES-KWP) Encrypted Output via Ethernet (AES-KWP)	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SKEYID, SKEYID_d:Derived From SKEYSEED, SK_d, SK_pi, SK_pr:Derived From
IKE Encryption Key	Encrypted Input via Ethernet (AES-KWP) Encrypted Output via	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SKEYID, SKEYID_d:Derived From SKEYSEED, SK_d, SK_pi, SK_pr:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Ethernet (AES-KWP)				
Inspection Authentication Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Inspection Master Secret:Derived From
Inspection DH Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Inspection DH Public Key:Paired With Inspection Pre-Master Secret:Derived From
Inspection DH Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Inspection DH Private Key:Paired With Inspection Pre-Master Secret:Derived From
Inspection ECDH Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Inspection ECDH Public Key:Paired With Inspection Pre-Master Secret:Derived From
Inspection ECDH Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Inspection ECDH Private Key:Paired With Inspection Pre-Master Secret:Derived From
Inspection Encryption Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Inspection Master Secret:Derived From
Inspection Master Secret		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Inspection Pre-Master Secret:Derived From Inspection Encryption

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Key:Derived From Inspection Authentication Key:Derived From
Inspection Pre-Master Secret		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Inspection ECDH Private Key:Derived From Inspection ECDH Public Key:Derived From Inspection DH Private Key:Derived From Inspection DH Public Key:Derived From Inspection Master Secret:Derived From
IPsec Authentication Key	Encrypted Input via Ethernet (AES-KWP) Encrypted Output via Ethernet (AES-KWP)	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SKEYID, SKEYID_d:Derived From SKEYSEED, SK_d, SK_pi, SK_pr:Derived From
IPsec Encryption Key	Encrypted Input via Ethernet (AES-KWP) Encrypted Output via Ethernet (AES-KWP)	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SKEYID, SKEYID_d:Derived From SKEYSEED, SK_d, SK_pi, SK_pr:Derived From
Key Encryption Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Key Encryption Passphrase:Paired With
Key Encryption Passphrase	Plaintext Input via Ethernet Plaintext	Disk - Plaintext:Plaintext	Until Zeroized	Factory Reset Power-Cycling	Key Encryption Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Output via Ethernet				
Server Protection ECDSA Private Key	Encrypted Input via Ethernet (AES GCM) Encrypted Output via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC) Encrypted Output via Ethernet (AES CBC + HMAC)	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	Server Protection ECDSA Public Key:Paired With
Server Protection ECDSA Public Key	Encrypted Input via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC)	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	Server Protection ECDSA Private Key:Paired With
Server Protection RSA Private Key	Encrypted Input via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC)	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	Server Protection RSA Public Key:Paired With
Server Protection RSA Public Key	Encrypted Input via Ethernet (AES GCM) Encrypted Input via	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	Server Protection RSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Ethernet (AES CBC + HMAC)				
SKEYID, SKEYID_d	Encrypted Input via Ethernet (AES-KWP) Encrypted Output via Ethernet (AES-KWP)	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	VPN DH Shared Secret:Derived From VPN ECDH Shared Secret:Derived From VPN Pre-Shared Key:Derived From IKE Encryption Key:Used With IKE Authentication Key:Used With IPsec Encryption Key:Used With TLS Authentication Key:Used With
SKEYSEED, SK_d, SK_pi, SK_pr	Encrypted Input via Ethernet (AES-KWP) Encrypted Output via Ethernet (AES-KWP)	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	VPN DH Shared Secret:Derived From VPN ECDH Shared Secret:Derived From VPN Pre-Shared Key:Derived From IKE Encryption Key:Used With IKE Authentication Key:Used With IPsec Encryption Key:Used With IPsec Authentication Key:Used With
SNMP Authentication Key	Encrypted Input via Ethernet (AES GCM) Encrypted Input via Ethernet	Disk - Plaintext:Plaintext Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	(AES CBC + HMAC)				
SNMP Encryption Key	Encrypted Input via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC)	Disk - Plaintext:Plaintext Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	
SSM Client Protection ECDSA Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM Client Protection ECDSA Public Key:Paired With
SSM Client Protection ECDSA Public Key	Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM Client Protection ECDSA Private Key:Paired With
SSM Client Protection RSA Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM Client Protection RSA Public Key:Paired With
SSM Client Protection RSA Public Key	Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM Client Protection RSA Private Key:Paired With
SSM HTTPS Authentication Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM HTTPS Master Secret:Derived From
SSM HTTPS DH Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM HTTPS DH Public Key:Paired With SSM HTTPS Pre-Master Secret:Derived From
SSM HTTPS DH Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM HTTPS DH Private Key:Paired With SSM HTTPS Pre-Master

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Secret:Derived From
SSM HTTPS ECDH Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM HTTPS ECDH Public Key:Paired With SSM HTTPS Pre-Master Secret:Derived From
SSM HTTPS ECDH Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM HTTPS ECDH Private Key:Paired With SSM HTTPS Pre-Master Secret:Derived From
SSM HTTPS Encryption Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM HTTPS Master Secret:Derived From
SSM HTTPS Master Secret		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM HTTPS Pre-Master Secret:Derived From SSM HTTPS Encryption Key:Derived From SSM HTTPS Authentication Key:Derived From
SSM HTTPS Pre-Master Secret		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	SSM HTTPS DH Private Key:Derived From SSM HTTPS DH Public Key:Derived From SSM HTTPS ECDH Private Key:Derived From SSM HTTPS ECDH Public Key:Derived From SSM HTTPS Master

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Secret:Derived From
State Synchronization Key	Encrypted Input via Ethernet (AES-KWP) Encrypted Output via Ethernet (AES-KWP)	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	
TLS Authentication Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	TLS Master Secret:Derived From
TLS ECDH Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	TLS Pre-Master Secret:Derived From TLS ECDH Public Key:Paired With
TLS ECDH Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	TLS ECDH Private Key:Paired With TLS Pre-Master Secret:Paired With
TLS ECDSA Private Key		Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	TLS ECDSA Public Key:Paired With
TLS ECDSA Public Key	Plaintext Output via Ethernet	Disk - Plaintext:Plaintext	Until Zeroized	Factory Reset Power-Cycling	TLS ECDSA Private Key:Paired With
TLS Encryption Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	TLS Master Secret:Derived From
TLS Master Secret		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	TLS Pre-Master Secret:Derived From TLS Encryption Key:Derived From TLS Authentication Key:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
TLS Pre-Master Secret		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	TLS ECDH Private Key:Derived From TLS Master Secret:Derived From TLS ECDH Public Key:Used With
TLS Trusted Certificates	Plaintext Input via Ethernet	Disk - Plaintext:Plaintext	Until Zeroized	Factory Reset Power-Cycling	
Trusted Internet Certificates	Encrypted Input via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC)	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	
User Password	Input Hashed and Salted	Disk - Plaintext:Plaintext	Until Zeroized	Factory Reset Power-Cycling	
VPN DH Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	VPN DH Public Key:Paired With VPN DH Shared Secret:Paired With
VPN DH Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	Disk - Plaintext:Plaintext	Until Zeroized	Factory Reset Power-Cycling	VPN DH Private Key:Paired With VPN DH Shared Secret:Derived From
VPN DH Shared Secret		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	VPN DH Private Key:Paired With VPN DH Public Key:Paired With
VPN ECDH Private Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	VPN ECDH Public Key:Paired With VPN ECDH Shared Secret:Paired With
VPN ECDH Public Key	Plaintext Input via	SDRAM :Plaintext	Until Zeroized	Factory Reset	VPN ECDH Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Ethernet Plaintext Output via Ethernet			Power-Cycling	VPN ECDH Shared Secret:Derived From
VPN ECDH Shared Secret		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	VPN ECDH Private Key:Paired With TLS ECDH Public Key:Paired With
VPN ECDSA Private Key	Encrypted Input via Ethernet (AES GCM) Encrypted Output via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC) Encrypted Output via Ethernet (AES CBC + HMAC)	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	VPN ECDSA Public Key:Paired With
VPN ECDSA Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	VPN ECDSA Private Key:Paired With
VPN Key Wrapping Key		SDRAM :Plaintext	Until Zeroized	Factory Reset Power-Cycling	Cluster Protocol Key:Used With
VPN Pre-Shared Key	Encrypted Input via Ethernet (AES GCM) Encrypted Input via Ethernet	Disk - Plaintext:Plaintext Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	SKEYID, SKEYID_d:Derived From SKEYSEED, SK_d, SK_pi, SK_pr:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	(AES CBC + HMAC)				
VPN RSA Private Key	Encrypted Input via Ethernet (AES GCM) Encrypted Output via Ethernet (AES GCM) Encrypted Input via Ethernet (AES CBC + HMAC) Encrypted Output via Ethernet (AES CBC + HMAC)	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	VPN RSA Public Key:Paired With
VPN RSA Public Key	Plaintext Input via Ethernet Plaintext Output via Ethernet	Disk - Encrypted:Encrypted	Until Zeroized	Factory Reset Power-Cycling	VPN RSA Private Key:Paired With
VPN Trusted Certificates	Plaintext Input via Ethernet	Disk - Plaintext:Plaintext	Until Zeroized	Factory Reset Power-Cycling	

Table 19: SSP Table 2

10 Self-Tests

10.1 Pre-Operational Self-Tests

The module performs pre-operational self-tests upon power-up to confirm firmware integrity and verify the correct operation of each implemented cryptographic algorithm used in support of the integrity checks. Specifically, the module performs the ECDSA Verify KAT and SHA2-512 KAT during initialization.

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
Pre-operational Bypass Test	Bypass Test	Bypass Test	Bypass	Success: "FIPS bypass test passed" output to console Failure: "FIPS bypass test FAILED, rebooting..." output to console, module reboots	Alternating Bypass mode with preset policy and routing configuration
Root Filesystem Integrity Test	ECDSA with P-521 and SHA2-512	Firmware Integrity	SW/FW Integrity	Success: "FIPS: rootfs integrity check OK" output to console Failure: "FIPS: rootfs integrity check FAILED, rebooting..." output to console, module reboots	ECDSA KAT (Sign, Verify) SHA2-512 KAT (Digest) Firmware Integrity (Verify, Digest)

Table 20: Pre-Operational Self-Tests

10.2 Conditional Self-Tests

The module automatically performs conditional self-tests based on the module operation. These self-tests do not require operator input to initiate. Implemented conditional tests are in one of the following forms:

- Known Answer Test (KAT)
- Pair-Wise Consistency Test (PCT)
- Health tests
- Bypass tests

All KATs alongside health testing of the noise source are performed immediately following the pre-operational self-tests at module power-on.

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Adaptive Proportion Test	APT	Fault Detection Test	CAST	FIPS: OpenSSL self-tests FAILED, rebooting...	Comparison of Samples within Window	At Startup and Upon Entropy Generation
AES-CBC Decrypt (A2166)	128, 192, 256 bit	KAT	CAST	NGFW Cryptographic Kernel Module failed to load, rebooting...	Decrypt	Upon Library Load

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CBC Decrypt (A4793)	128-bit	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Decrypt	Upon Library Load
AES-CBC Encrypt (A2166)	128, 192, 256 bit	KAT	CAST	NGFW Cryptographic Kernel Module failed to load, rebooting...	Encrypt	Upon Library Load
AES-CBC Encrypt (A4793)	128-bit	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Encrypt	Upon Library Load
AES-GCM Decrypt (A2166)	128 bit	KAT	CAST	NGFW Cryptographic Kernel Module failed to load, rebooting...	Decrypt	Upon Library Load
AES-GCM Decrypt (A4793)	128-bit	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Decrypt	Upon Library Load
AES-GCM Encrypt (A2166)	128 bit	KAT	CAST	NGFW Cryptographic Kernel Module failed to load, rebooting...	Encrypt	Upon Library Load
AES-GCM Encrypt (A4793)	128-bit	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Encrypt	Upon Library Load
Configuration Bypass Test	HMAC Bypass Test	Bypass Test	Bypass	FIPS: Cryptographic module self-tests FAILED, rebooting..	MAC Verification	New Policy Files Received

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Counter DRBG Reseed Health Test	AES-256	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Reseed	Upon Library Load
Counter DRBG (A4793)	AES-256	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	[Prediction Resistance: No],[Derivation Function Enabled: Yes]	Upon Library Load
Counter DRBG Instantiate and Generate Health Tests (A4793)	AES-256	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Instantiate and Generate	Upon Library Load
ECDSA SigGen (FIPS186-5) (A4793)	P-224 w/ SHA2-224	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Sign	Upon Library Load
ECDSA SigVer (FIPS186-5) (A4793)	P-224 w/ SHA2-224	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Verify	Upon Library Load
HMAC-SHA-1 (A2166)	SHA-1; Key length 640 bits	KAT	CAST	NGFW Cryptographic Kernel Module failed to load, rebooting...	Message Authentication	Upon Library Load
HMAC-SHA2-256 (A2166)	SHA2-256; Key length 1048 bits HMAC-SHA2-256-96	KAT	CAST	NGFW Cryptographic Kernel Module failed to load, rebooting...	Message Authentication	Upon Library Load
HMAC-SHA2-256 (A4793)	SHA2-256	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Message Authentication	Upon Library Load

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-512 (A2166)	SHA2-512; Key length 160 bits	KAT	CAST	NGFW Cryptographic Kernel Module failed to load, rebooting...	Message Authentication	Upon Library Load
KAS-ECC-SSC Sp800-56Ar3 (A4793)	P-224	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Shared Secret "Z" Computation	Upon Library Load
KAS-FFC-SSC Sp800-56Ar3 (A4793)	2048	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Shared Secret "Z" Computation	Upon Library Load
KDF IKEv1 (A4793)	SHA2-256	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Secure Hash	Upon Library Load
KDF IKEv2 (A4793)	SHA2-256	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Secure Hash	Upon Library Load
KDF SP800-108 (A4550)	HMAC-SHA2-256 (Mode: Counter)	KAT	CAST	FIPS: OpenSSL self-tests FAILED, rebooting...	Key Derivation	Upon Library Load
Lag Predictor Test	Lag Predictor Test	Fault Detection Test	CAST	FIPS: OpenSSL self-tests FAILED, rebooting...	Designed to detect a known failure mode where the result becomes mostly deterministic.	At Startup and Upon Entropy Generation
PBKDF (A4550)	HMAC-SHA2-256	KAT	CAST	FIPS: OpenSSL self-tests FAILED, rebooting...	Key Derivation	Upon Library Load
PCT for ECDSA key pairs created	ECDSA Sign/Verify	PCT	PCT	FIPS: OpenSSL self-tests	Key Generation	Upon ECDSA Key Generation

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
for digital signature purposes	and SHA2-256			FAILED, rebooting...		
PCT for KAS-ECC key pairs created for key agreement purposes	KAS-ECC-SSC	PCT	PCT	FIPS: OpenSSL self-tests FAILED, rebooting...	Key Generation	Upon ECDH Key Generation
PCT for KAS-FFC key pairs created for key agreement purposes	KAS-FFC-SSC	PCT	PCT	FIPS: OpenSSL self-tests FAILED, rebooting...	Key Generation	Upon DH Key Generation
PCT for RSA key pairs created for digital signature purposes	RSA Sign/Verify w/ PKCS#1v1.5 and SHA2-256	PCT	PCT	FIPS: OpenSSL self-tests FAILED, rebooting...	Key Generation	Upon RSA Key Generation
Repetition Count Test	RCT	Fault Detection Test	CAST	FIPS: OpenSSL self-tests FAILED, rebooting...	Comparison of Subsequent Entropy Samples	At Startup and Upon Entropy Generation
RSA SigGen (FIPS186-5) (A4793)	RSA 2048 w/ SHA2-256 PKCS#1v1.5	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Sign	Upon Library Load
RSA SigVer (FIPS186-5) (A4793)	RSA 2048 w/ SHA2-256 PKCS#1v1.5	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Verify	Upon Library Load
SHA-1 (A2166)	SHA-1	KAT	CAST	NGFW Cryptographic Kernel Module failed to load, rebooting...	Secure Hash	Upon Library Load
SHA-1 (A4793)	SHA-1	KAT	CAST	FIPS: Cryptographic module self-	Secure Hash	Upon Library Load

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
				tests FAILED, rebooting..		
SHA2-256 (A2166)	SHA2-256	KAT	CAST	NGFW Cryptographic Kernel Module failed to load, rebooting...	Secure Hash	Upon Library Load
SHA2-512 (A2166)	SHA2-512	KAT	CAST	NGFW Cryptographic Kernel Module failed to load, rebooting...	Secure Hash	Upon Library Load
SHA2-512 (A4793)	SHA2-512	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Secure Hash	Upon Library Load
SHA3-256 (A4426)	SHA3-256	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting...	Entropy Conditioning Component	Upon Library Load
TLS v1.2 KDF RFC7627 (A4793)	HMAC-SHA2-256	KAT	CAST	FIPS: Cryptographic module self-tests FAILED, rebooting..	Secure Hash	Upon Library Load

Table 21: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Pre-operational Bypass Test	Bypass Test	Bypass	On-Demand	Programmatically
Root Filesystem Integrity Test	Firmware Integrity	SW/FW Integrity	On-Demand	Programmatically

Table 22: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Adaptive Proportion Test	Fault Detection Test	CAST	On Demand	Programmatically
AES-CBC Decrypt (A2166)	KAT	CAST	On Demand	Programmatically
AES-CBC Decrypt (A4793)	KAT	CAST	On Demand	Programmatically
AES-CBC Encrypt (A2166)	KAT	CAST	On Demand	Programmatically
AES-CBC Encrypt (A4793)	KAT	CAST	On Demand	Programmatically
AES-GCM Decrypt (A2166)	KAT	CAST	Upon Library Load	Programmatically
AES-GCM Decrypt (A4793)	KAT	CAST	On Demand	Programmatically
AES-GCM Encrypt (A2166)	KAT	CAST	Upon Library Load	Programmatically
AES-GCM Encrypt (A4793)	KAT	CAST	On Demand	Programmatically
Configuration Bypass Test	Bypass Test	Bypass	On Demand	Programmatically
Counter DRBG Reseed Health Test	KAT	CAST	On Demand	On Programmatically
Counter DRBG (A4793)	KAT	CAST	On Demand	Programmatically
Counter DRBG Instantiate and Generate Health Tests (A4793)	KAT	CAST	On Demand	Programmatically
ECDSA SigGen (FIPS186-5) (A4793)	KAT	CAST	On Demand	Programmatically
ECDSA SigVer (FIPS186-5) (A4793)	KAT	CAST	On Demand	Programmatically
HMAC-SHA-1 (A2166)	KAT	CAST	On Demand	Programmatically
HMAC-SHA2-256 (A2166)	KAT	CAST	On Demand	Programmatically
HMAC-SHA2-256 (A4793)	KAT	CAST	On Demand	Programmatically
HMAC-SHA2-512 (A2166)	KAT	CAST	On Demand	Programmatically

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KAS-ECC-SSC Sp800-56Ar3 (A4793)	KAT	CAST	On Demand	Programmatically
KAS-FFC-SSC Sp800-56Ar3 (A4793)	KAT	CAST	On Demand	Programmatically
KDF IKEv1 (A4793)	KAT	CAST	On Demand	Programmatically
KDF IKEv2 (A4793)	KAT	CAST	On Demand	Programmatically
KDF SP800-108 (A4550)	KAT	CAST	On Demand	Programmatically
Lag Predictor Test	Fault Detection Test	CAST	On Demand	On Programmatically
PBKDF (A4550)	KAT	CAST	On Demand	Programmatically
PCT for ECDSA key pairs created for digital signature purposes	PCT	PCT	On Demand	Programmatically
PCT for KAS-ECC key pairs created for key agreement purposes	PCT	PCT	On Demand	On Programmatically
PCT for KAS-FFC key pairs created for key agreement purposes	PCT	PCT	On Demand	Programmatically
PCT for RSA key pairs created for digital signature purposes	PCT	PCT	On Demand	Programmatically
Repetition Count Test	Fault Detection Test	CAST	On Demand	Programmatically
RSA SigGen (FIPS186-5) (A4793)	KAT	CAST	On Demand	Programmatically
RSA SigVer (FIPS186-5) (A4793)	KAT	CAST	On Demand	Programmatically
SHA-1 (A2166)	KAT	CAST	On Demand	Programmatically
SHA-1 (A4793)	KAT	CAST	On Demand	Programmatically

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
SHA2-256 (A2166)	KAT	CAST	On Demand	Programmatically
SHA2-512 (A2166)	KAT	CAST	On Demand	Programmatically
SHA2-512 (A4793)	KAT	CAST	On Demand	Programmatically
SHA3-256 (A4426)	KAT	CAST	On Demand	Programmatically
TLS v1.2 KDF RFC7627 (A4793)	KAT	CAST	On Demand	Programmatically

Table 23: Conditional Periodic Information

10.4 Error States

If one of the pre-operational self-tests fails or a conditional self-test fails, the module enters an error state. An error message is output on the status output interface specifying the library within the module that failed the self-test. In this state, all data output via the module's data output interfaces is inhibited. The module proceeds to reboot and re-runs all self-tests. Successful completion of the self-tests will clear the error state, and the module will return to the Approved mode of operation. For any consecutive failure of the self-tests during restart, the appliance continues to restart. If the problem persists, CO intervention is required to either perform a restore to factory defaults settings and reinstall, or power-off and contact Forcepoint Customer Support.

Name	Description	Conditions	Recovery Method	Indicator
Error State	Performs automatic power-cycle upon self-test failure.	Module fails any of the other self-tests	No operator action is required, the appliance will restart itself. If the restart also fails then the appliance needs to be reset and reconfigured, or if due to hardware malfunction the appliance must be replaced	Pre-operational bypass test: "FIPS bypass test FAILED, rebooting..." is printed to the kernel log Rootfs integrity test: "FIPS: OpenSSL self-tests" is logged CASTs (depending on which algorithm and service was used for the failing test: "FIPS: OpenSSL self-tests FAILED, rebooting..." is logged or "FIPS: Cryptographic module self-tests FAILED, rebooting..." is logged

Table 24: Error States

10.6 Additional Information

While the module is running these self-tests, all data output interfaces are disabled until the successful completion of the self-tests.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

11.1.1 Hardware Setup

Upon receiving the NGFW hardware, the CO shall check that the appliance is not damaged and that all required parts and instructions are included. If the Network Components are not installed in the appliance, the CO must insert them by performing the following:

Note: Read all safety instructions before installing the Network Components. Do not install any Network Components while the appliance is on. Fasten a grounding strip from the wrist to the appliance.

1. Locate the Network Component slots on the front of the appliance.
2. If the appliance was shipped with the Network Component slot(s) covered by a plate, remove the thumbscrew and plate from the appliance. Store the thumbscrew and plate in case the Network Component is eventually removed.
3. Push the Network Component into the slot. The Network Component is properly installed when the front of the Network Component is flush with the front of the appliance.

11.1.2 Creating a Configuration for the Approved Mode of Operation

The administration of the NGFW modules is done through the SMC, which provides centralized administrative functionalities for all the managed NGFW modules. The SMC can be shipped preinstalled on its own Forcepoint hardware appliance, installed as a virtual machine on a virtualization platform, or installed on a third-party Windows or Linux platform. The SMC can be accessed by an administrator via a Java-based Management Client running on the administrator's workstation.

Using the Management Client, create a configuration for the NGFW Engine in the Approved Mode of Operation.

1. To use HTTPS User Authentication and TLS Inspection for Client Protection or Server Protection, create a TLS Cryptography Suite Set element. Select only the Approved and Allowed algorithms and TLS cipher suites. The Management Connection Service, Peer Connection Service, Key Pair Management Service, and User Management Service utilize the TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 cipher suite. Refer to Section "2.5 Algorithms" above for a list of algorithms implemented. For more information, see the "Create TLS Cryptographic Suite Set elements" topic of the Forcepoint NGFW Product Guide.
    - In accordance with the NGFW Product user guide, the following TLS Cipher Suites are utilized within the module:
        - TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
        - TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
        - TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
        - TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
        - TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256

- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384

2. To use certificates signed by a Certificate Authority (CA) that is not one of the default Trusted Certificate Authority elements, create a Trusted Certificate Authority element. Import only a certificate signed using an Approved signature algorithm. For more information, see the "Create Trusted Certificate Authority elements" topic of the Forcepoint NGFW Product Guide.

3. To use HTTPS User Authentication, create a TLS Profile element. Select the TLS Cryptography Suite Set element, the Trusted Certificate Authority, and the minimum TLS version. For more information, see the "Create TLS Profile elements" topic of the Forcepoint NGFW Product Guide.

4. Create the NGFW Engine Element by defining the properties in the Engine Editor.
   - Browse to Advanced Settings, then select FIPS-Compatible Operating Mode.
   - Select "FIPS-Disable Remote Engine Upgrades" for the NGFW to prevent firmware load attempts from the SMC.
   - To use HTTPS User Authentication, browse to Add-Ons | User Authentication, then enable HTTPS and select the TLS Profile element. Use 2048 or greater as the Key Length when creating a certificate signing request in HTTPS Settings. For more information, see the "Enable browser-based user authentication" topic of the Forcepoint NGFW Product Guide.
   - To use TLS Inspection for Client Protection, create a Client Protection Certificate Authority element and import the private key and the certificate used to issue certificates in TLS Inspection. Use only the Approved algorithms and key size for the key pair and certificate. In the Engine Editor, browse to Add-Ons | TLS Inspection, then select the Cryptography Suite Set. For more information, see the "Configure TLS inspection for client protection" and "Activating TLS inspection" topics in the "Setting up TLS Inspection" chapter of the Forcepoint NGFW Product Guide.
   - To use TLS Inspection for Server Protection, browse to Add-Ons | TLS Inspection, then select the Cryptography Suite Set. For more information, see the "Activating TLS inspection" topic in the "Setting up TLS Inspection" chapter of the Forcepoint NGFW Product Guide.
   - When using TLS Inspection or Sidewinder HTTPS proxy, create a Firewall Policy that has an Access rule that allows the TLS connection and create an Inspection Policy that has an Inspection rule that terminates connections that match the TLS_Certificate-Verify-Failed Situation. On the Inspection tab of the Firewall Policy, you must select the Inspection Policy that you created.
   - To use Sidewinder HTTP and HTTPS proxies, browse to Add-Ons | Sidewinder Proxy, click Advanced, then set the value of the tls_cipher_override property to TLSv1.2+ECDHE+AES!AESCCM:TLSv1.2+DHE+AES!AESCCM!DSS on the HTTP tab. For

more information, see the “Advanced settings for Sidewinder Proxies” topic in the “Sidewinder Proxies” chapter of the Forcepoint NGFW Product Guide.

- When using IPsec, disable Automated RSA Certificate Management. Browse to VPN | Certificates, then deselect Automated RSA Certificate Management.
- To use IPsec, right-click the Gateway element, then select Tools | Generate Certificate to create a certificate signing request. Select RSA with 2048 or greater key size, or ECDSA as the Public Key Algorithm. For more information, see the “Create a VPN certificate or certificate request for a VPN Gateway element” topic in the “Managing VPN certificates” chapter of the Forcepoint NGFW Product Guide.
- To use an IPsec VPN, create a VPN Profile element. Use only the Approved and Allowed algorithms and key sizes in the profile. Refer to Table 3: Approved Algorithms above for a list of algorithms implemented. Additionally, in the profile element, the IPsec Tunnel Lifetime should be set to less than 232 bytes. Select the VPN Profile element. For more information, see the “Create VPN Profile elements” topic in the “VPNs in Forcepoint NGFW” chapter of the Forcepoint NGFW Product Guide.
- Create Access Rules to configure the Alternating Bypass Feature.
- Save the initial configuration for the NGFW Engine. Make a note of the one-time password, which is required for initial contact with the SMC.

See Section “11.1.4 Setting up the Approved Configuration” for setting up the device configurations.

11.1.3 Downloading and Upgrading to an Approved Firmware Version

The NGFW appliances are delivered in an operational state with the most recent firmware preinstalled. The NGFW firmware must be upgraded to the FIPS 140-3 validated NGFW firmware version to be placed in the Approved mode of operation.

Note: The upgrade to the FIPS 140-3 validated NGFW firmware version is necessary even if the same version was installed previously. This is required because the file system checksum is stored during the upgrade process. A method to update the firmware image with a SHA2-512 checksum signed with ECDSA P-521 is provided. Prior to installing the new image, its associated checksum is checked. If the signature check fails, the new firmware is ignored, and the current firmware remains loaded. If the signature check passes, the new image will be installed and executed after the appliance is restarted. Any firmware loaded into the module other than version 6.10.13.26655.fips.2 is out of the scope of this validation and will mean that the module is not operating in the approved mode of operation.

A FIPS 140-3 Validated NGFW firmware version is downloaded as follows:

1. Login to the Forcepoint Support https://support.forcepoint.com/Login
2. Proceed to the Forcepoint NGFW downloads section.
3. Download the firmware version 6.10.13.26655.fips.2 installation file (sg_engine_6.10.13.26655.fips.2_x86-64-small.zip).
4. Verify the SHA checksum.

Note: The correct checksums are shown on the download page and can also be found in the release notes

After downloading the firmware, the operator can upgrade to a FIPS 140-3 validated firmware version:

1. Save the FIPS 140-3 validated NGFW firmware version upgrade .zip file to the root directory of a USB drive. **Note** – The firmware upgrade zip file must be in the root directory of the media.
2. Connect to the appliance using a monitor and keyboard.
3. Power on the appliance and start the NGFW Configuration Wizard.
4. Select the Firewall/VPN option.
5. Select Upgrade. The Select Source Media dialog opens.
6. Select the appropriate media type and select OK. The firmware update signature is verified.
7. Select OK. The upgrade starts.
8. Select “Set kernel in FIPS mode” after restart. Select OK.
9. The NGFW appliance restarts and displays the upgraded version.
10. Verify the NGFW firmware version to ensure that the FIPS validated NGFW firmware version is loaded.

11.1.4 Setting up the Approved Configuration

To configure the NGFW Engine:

1. Start the NGFW Configuration Wizard as instructed in the Configuring the Engine in the Engine Configuration Wizard section of the NGFW Installation Guide.
2. Configure the network interfaces according to your environment as instructed in the Configuring the Network Interfaces section of the NGFW Installation Guide.
    a. Configure the operating system settings according to the "Configuring the Operating System Settings" section of the NGFW Installation Guide.
    b. Select both:
        - "Restricted FIPS-compatible operating mode" (This automatically disables the SSH daemon and root password options in the Engine Configuration Wizard).
        - "FIPS 140-3 compatible mode" (This setting ensures the module uses only FIPS 140-3 approved algorithms and security functions).
3. Contact the Management Server as instructed in the Contacting the Management Server section of the NGFW Installation Guide. Enter node IP address manually is selected by default and other IP address options are disabled when the “Restricted FIPS-compatible operating mode” setting is enabled. The engine restarts.
4. To verify the “FIPS 140-3 compatible operating mode” setting is activated:
    a. Verify that the following messages are displayed on the console when the engine restarts:
        - FIPS: rootfs integrity check OK (Displayed after the root file system integrity test has been executed successfully)
        - FIPS power-up tests succeeded (Displayed after the FIPS 140 power-up tests have been executed successfully)
    b. Continue as instructed in the “After Successful Management Server Contact” section of the NGFW Installation Guide.

Note: If the engine does not enter the “Restricted FIPS-compatible operating mode” even though it is configured to do so, or if the power-up tests fail (a power-up test error message is displayed or the success message is not displayed), the appliance must be reset to factory settings and reinstalled.

Note: The “FIPS 140-3 compatible operating mode” and “Restricted FIPS-compatible operating mode” settings must be enabled during the initial configuration of the appliance.

11.1.5 Resetting the Module to Factory Settings (Sanitization)

Resetting the appliance to factory settings is not part of the normal installation procedure. There is no need to reset the appliance to factory settings before starting to use it for the first time. These instructions can be used to reset the appliance to factory settings when necessary, such as when initial configuration has been completed without enabling the “Restricted FIPS-compatible operating mode”, during use, or when the appliance is being removed from use.

To reset the appliance to factory settings:

1. Reboot the appliance and select System restore options from the boot menu. NGFW System Restore starts.
2. Enter 2 for Advanced data removal options.
3. Enter one of the following options:
    - 1 for 1 pass overwrite.
    - 8 for a Custom number of overwrite passes.

If you selected Custom, enter the number of overwrite passes. A larger number of overwrites is more secure, but it may take a considerable amount of time depending on the appliance storage capacity.

11.2 Administrator Guidance

The Crypto-Officer is responsible for ensuring the module is running in Approved mode of operation using the steps in Section “11.1 Installation, Initialization, and Startup Procedures”.

In order to identify the modules’ version, the operator must do the following:

1. At the Home screen of the SMC that is being used to manage the module, click on the firewall.
2. On the right-hand column, under “Info”, the firewall version (and update package) will be in the “General” tab

Figure 9: Depiction of the Module Version Displayed in the SMC GUI

If problems occur, Crypto-Officer intervention is required to either perform a restore to factory defaults settings and reinstall, or power-off and contact Forcepoint Customer Support.

11.3 Non-Administrator Guidance

The notes below provide additional guidance and policies that must be followed by module operators:

- **Use of AES GCM:** The module generates AES GCM IV in accordance with SP 800-38D in compliance with IG C.H scenario 1. The GCM IV generation in the TLS context follows RFC 5288 and SP 800-52rev2 section 3.3.1 and shall only be used for the TLS protocol version 1.2. The GCM IV generation in the IPsec context follows RFC 4106 and RFC 7296 and shall only be used with IPsec and IKEv2 to be compliant with IG C.H. The implementation of the 64-bit nonce_explicit part of the IV is deterministic and management logic is inside the module. By the design of the module and by virtue of the data size limit (see above Section "11.1.2 Creating a Configuration for the Approved Mode of Operation") set, the maximum number possible value of 2^64 for nonce_explicit part of the IV is never reached. In case the module's power is lost and then restored, the key used for the AES GCM encryption or decryption shall be re-distributed.
- **Use of PBKDF:** The module implements key derivation through the SP 800-132 PBKDF2. The module supports option 1a from Section 5.4 of SP 800-132, whereby the MK is used directly as the DPK. Keys derived from passwords or passphrases are only used for data at rest. The length of the salt should be at least 128 bits and the length of the password or passphrase should be at least 10 characters, which provides the probability of guessing this password or passphrase to be (1/94)10. The caller shall observe all requirements and should consider all recommendations specified in SP 800-132 with respect to the strength of the generated key, including the quality of the password and the quality of the salt. Keys derived from passwords, as shown in SP 800-132, may only be used in storage applications. For encrypted private key entry as part of the configuration, the PBKDF2 iteration count must be between 1000 and 10000 to allow the recommended minimum in SP 800-132 while keeping performance the impact small, and the passphrase must be at least 14 characters.
- **Use of insecure protocols –** The following insecure protocols are disabled by default: SSH, Console Access, and WIFI Interfaces. The root password option is automatically disabled. To maintain compliance with FIPS requirements, these protocols and services shall not be enabled.
- **Network Component replacement –** As noted earlier, the NGFW appliances are modular by design. The Network Components are field-replaceable. Operators in the field can order the desired Network Components directly from Forcepoint Customer Support using the appropriate part numbers. The CO must install the Network Components as described in Section "11.1.1 Hardware Setup" above.
- **Use of Legacy 1024-bit RSA –** Operators of the module shall only import 1024-bit RSA keys created prior to the 2011 transition date and the operator shall not verify any signatures signed after 2011. By default, the module does not include any trusted certificates with 1024-bit RSA keys that were signed after the 2011 transition date.

12 Mitigation of Other Attacks

This Section is Not Applicable. The modules do not claim to mitigate any attacks beyond the FIPS 140-3 Level 1 requirements for this validation.