Palo Alto Networks Inc.

WildFire 11.1/11.2 WF-500 and WF-500-B

FIPS 140-3 Non-Proprietary Security Policy

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This document may freely be reproduced and distributed in its entirety.

The WildFire 11.1.3/11.2.5 WF-500 and WF-500-B from Palo Alto Networks Inc., hereafter referred to as “Wildfire” or the “cryptographic module” is a multi-chip standalone hardware cryptographic module designed to fulfill FIPS 140-3 level 2 requirements. The WildFire 11.1.3/11.2.5 WF-500 and WF-500-B module identifies unknown malware, zero-day exploits, and Advanced Persistent Threats (APTs) through dynamic analysis, and automatically disseminates protection in near real-time to help security teams meet the challenge of advanced cyber-attacks.

Unknown files are analyzed by WildFire (WF) in a scalable sandbox environment where new threats are identified, and protections are automatically developed and delivered in the form of an update. The result is a unique, closed loop approach to controlling cyber threats that begins with positive security controls to reduce the attack surface, inspection of all traffic, ports, and protocols to block all known threats, and rapid detection of unknown threats by observing their actual behavior.

The cryptographic module meets the overall requirements applicable to Level 2 security of FIPS 140-3.

1.2 Security Levels

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

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The Palo Alto Networks, Inc. WildFire 11.2 WF-500 and WF-500-B is a multi-chip standalone hardware module. The cryptographic boundary includes all firmware components contained within the physical enclosure of the module.

Figures below provide images of the module with the physical kit's opacity shields in place. See the Physical Security section for details regarding the module's physical security mechanisms.

Module Type: Hardware

Figure 1 - WF-500 Front

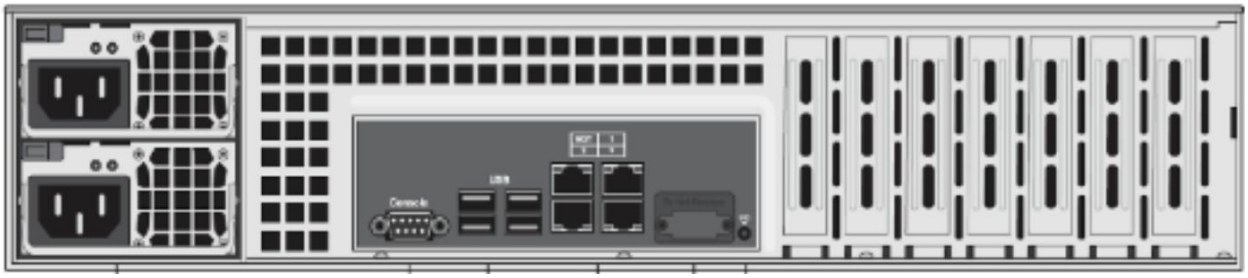

Figure 2 - WF -500 Rear

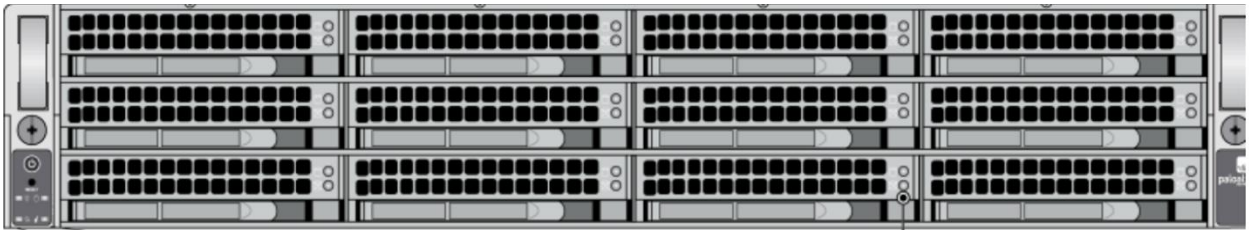

Figure 3 - WF-500-B Front

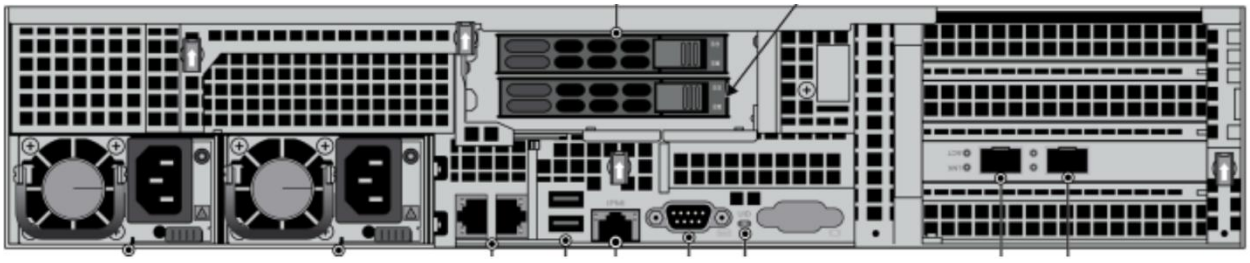

Figure 4 - WF-500-B Rear

Module Embodiment: Multi-Chip Standalone

Cryptographic Boundary:

The cryptographic boundary includes the physical perimeter of the enclosure of the appliance and all logical components within. Please refer to the 'Physical Security' section for depictions of the module with the physical kit installed.

Tested Operational Environment's Physical Perimeter (TOEPP):

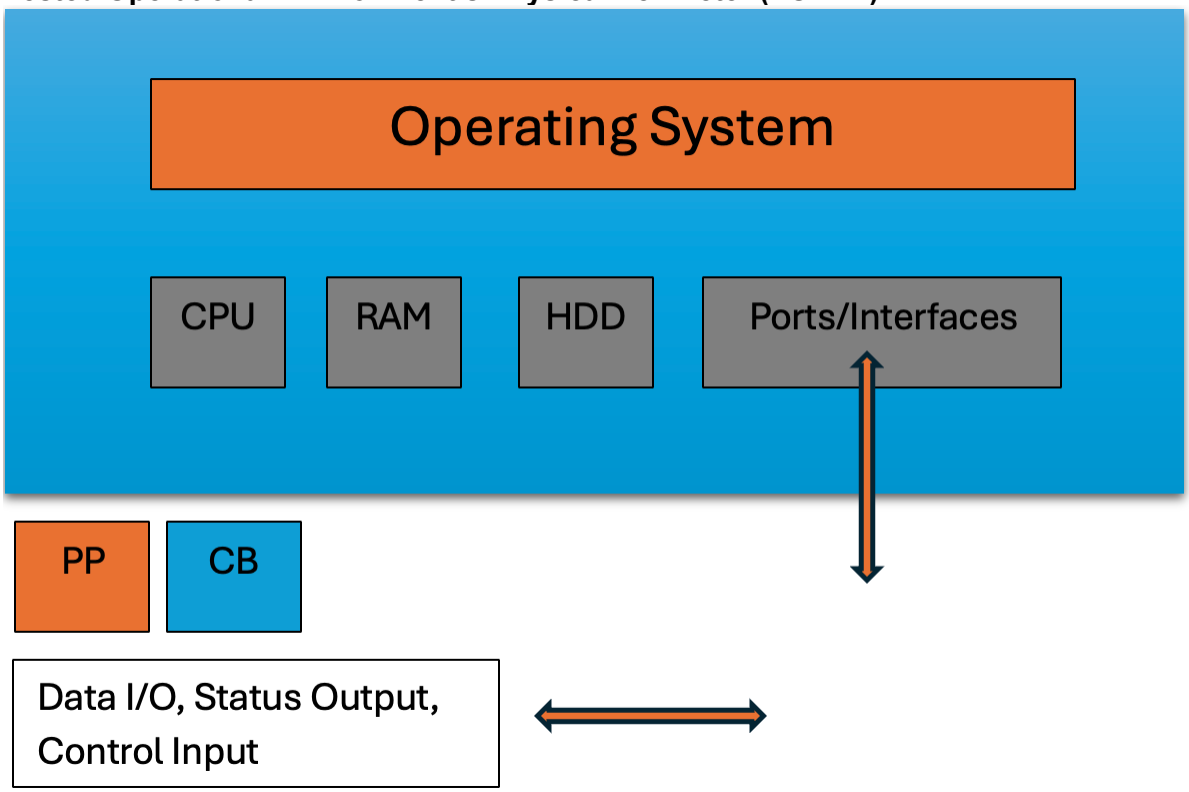

Figure 5 - Block Diagram

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
WF-500	910-000097 Physical Kit: 920-000145	11.1.3, 11.2.5	Intel Xeon E5-2620 V4	RJ45 interfaces, USB ports, LEDs
WF-500-B	910-000270 Physical Kit: 920-000318	11.1.3, 11.2.5	Intel Xeon (Silver) 4316 Broadwell	RJ45 interfaces, USB ports, LEDs, SFP+ ports

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

N/A

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved Mode	The module has one approved mode of operation and is always in approved mode after initialization	Approved	Global indicator ("FIPS-CC")

Table 3: Modes List and Description

The following procedure will initialize the modules into the Approved mode of operation:

● Install module and interface connections in addition to the physical kit.
● The tamper-evident seals and opacity shields must be installed as per the ‘Physical Security’ section for the
module to operate in the Approved mode of operation.
● Apply power to the device.
● Establish a serial connection to the console port and command the module to enter into maintenance mode. During initial boot up, break the boot sequence via the console port connection (by pressing the main button when instructed to do so) to access the main menu.
● Select “Continue.”
● Select the “Set FIPS-CC Mode” option to enter the Approved mode.
● Select “Enable FIPS-CC Mode,” and press enter.
● When prompted, select “Reboot” and the module will re-initialize and continue into the Approved mode.
● The module will reboot.
● In the Approved mode, the console port is available only as a status output port.
● Once the module has finished booting, the Crypto Officer can authenticate using the default credentials that come with the module. Once authenticated, the module will automatically require the operator to change their password; and the default credential is overwritten
The module will automatically indicate the Approved mode of operation in the following manner:

● Status output interface will indicate “**** FIPS-CC MODE ENABLED ****” via the CLI session.
● Status output interface will indicate “FIPS-CC mode enabled successfully” via the console port. Should one or more power-up self-tests fail, the module will not enter the Approved mode of operation. Feedback will consist of:

● The module will output “FIPS-CC failure.

● The module will reboot and enter a state in which the reason for the reboot can be determined by following the
on-screen instructions.

Note: Disabling Approved mode causes a complete factory reset, which is described in the Zeroization section below.

Failure to follow the directions in the Approved Mode of Operation above and Section 11 will result in the module operating in a non-compliant state.

Zeroization:

To initiate the zeroization service, perform the following steps:
● Access the module’s CLI via SSH, and command the module to enter maintenance mode; the module will reboot
  - Note: Establish a serial connection to the console port
● After reboot, select “Continue.
● Select “Factory Reset.
● The module will perform a zeroization, and provide the following message once complete:
  “Factory Reset Status: Success”
If the module does not successfully transition into the Approved mode of operation, or zeroization is performed, the module will be in an uninitialized state. It is required to initialize the module in order to perform cryptographic functions.

Mode Change Instructions and Status:

“Note: The module supports a “maintenance mode” which corresponds to an uninitialized state. The maintenance mode does not support any approved security functions. Please refer to section 11 of this document for further detail.”

Degraded Mode Description:

See Life-Cycle Assurance section.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A3453	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A3453	Key Length - 128, 192, 256	SP 800-38C
AES-CFB1	A3453	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB128	A3453	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB8	A3453	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CTR	A3453	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A3453	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 192, 256	SP 800-38D
Counter DRBG	A3453	Prediction Resistance - No, Yes Mode - AES-256 Derivation Function Enabled - No, Yes	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-4)	A3453	Curve - P-256, P-384, P-521 Secret Generation Mode - Testing Candidates	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A3453	Curve - P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A3453	Curve - P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A3453	Curve - P-256, P-384, P-521 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-4
HMAC-SHA-1	A3453	Key Length - Key Length: 256-2048 Increment 8	FIPS 198-1
HMAC-SHA2-224	A3453	Key Length - Key Length: 256-2048 Increment 8	FIPS 198-1
HMAC-SHA2-256	A3453	Key Length - Key Length: 256-2048 Increment 8	FIPS 198-1
HMAC-SHA2-384	A3453	Key Length - Key Length: 256-2048 Increment 8	FIPS 198-1
HMAC-SHA2-512	A3453	Key Length - Key Length: 256-2048 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A3453	Domain Parameter Generation Methods - P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KAS-FFC-SSC Sp800-56Ar3	A3453	Domain Parameter Generation Methods - MODP-2048, MODP-3072, MODP-4096 Scheme - dhEphem - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDF IKEv2 (CVL)	A3453	Diffie-Hellman Shared Secret Length - Diffie-Hellman Shared Secret Length: 256, 384, 2048 Derived Keying Material Length - Derived Keying Material Length: 800-3072 Increment 8 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
KDF SNMP (CVL)	A3453	Password Length - Password Length: 64, 2048	SP 800-135 Rev. 1
KDF SSH (CVL)	A3453	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA-1, SHA2-256, SHA2-512	SP 800-135 Rev. 1

Algorithm	CAVP Cert	Properties	Reference
RSA KeyGen (FIPS186-4)	A3453	Key Generation Mode - B.3.6 Modulo - 2048, 3072, 4096 Primality Tests - Table C.2 Private Key Format - Standard	FIPS 186-4
RSA SigGen (FIPS186-4)	A3453	Signature Type - ANSI X9.31, PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
RSA SigVer (FIPS186-4)	A3453	Signature Type - ANSI X9.31, PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
Safe Primes Key Generation	A3453	Safe Prime Groups - MODP-2048, MODP-3072, MODP-4096	SP 800-56A Rev. 3
Safe Primes Key Verification	A3453	Safe Prime Groups - MODP-2048, MODP-3072, MODP-4096	SP 800-56A Rev. 3
SHA-1	A3453	Message Length - Message Length: 8-65536 Increment 8	FIPS 180-4
SHA2-224	A3453	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A3453	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A3453	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A3453	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
TLS v1.2 KDF RFC7627 (CVL)	A3453	Hash Algorithm - SHA2-256, SHA2-384	SP 800-135 Rev. 1

Table 4: Approved Algorithms

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Key Type:Symmetric and Asymmetric	N/A	SP 800-133rev2 Section 4 example 1

Table 5: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

From Web Cryptik

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

From Web Cryptik

Non-Approved, Not Allowed Algorithms:

N/A for this module.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
CKG - Symmetric	CKG	Used for symmetric key generation		Counter DRBG: (A3453)
Firmware Load Test	DigSig-SigVer	Signature verification for firmware load test		RSA SigVer (FIPS186-4): (A3453) SHA2-256: (A3453)
IPSec/IKE ECDSA KeyGen	AsymKeyPair-KeyGen CKG	ECDSA KeyGen for IPSec/IKEv2		ECDSA KeyGen (FIPS186-4): (A3453) Counter DRBG: (A3453) CKG: () Key Type: Symmetric and Asymmetric
IPSec/IKE ECDSA SigGen	DigSig-SigGen	ECDSA SigGen for IPSec/IKEv2		ECDSA SigGen (FIPS186-4): (A3453) Counter DRBG: (A3453)
IPSec/IKE ECDSA SigVer	DigSig-SigVer	ECDSA SigVer for IPSec/IKEv2		ECDSA SigVer (FIPS186-4): (A3453) ECDSA KeyVer (FIPS186-4): (A3453)
IPSec/IKE Keying Materials Development	KAS-135KDF	IPSec/IKE session keying materials, used to derive IPSec/IKE session keys		KDF IKEv2: (A3453)
IPSec/IKE RSA KeyGen	AsymKeyPair-KeyGen CKG	RSA KeyGen for IPSec/IKEv2		RSA KeyGen (FIPS186-4): (A3453) Counter DRBG: (A3453) CKG: () Key Type: Symmetric and Asymmetric
IPSec/IKE RSA SigGen	DigSig-SigGen	RSA SigGen for IPSec/IKEv2		RSA SigGen (FIPS186-4): (A3453)

Name	Type	Description	Properties	Algorithms
IPSec/IKE RSA SigVer	DigSig-SigVer	RSA SigVer for IPSec/IKEv2		RSA SigVer (FIPS186-4): (A3453)
KAS-ECC (IPSec/IKE)	KAS-Full	Full KAS-ECC Key Agreement used for IPSec/IKEv2 service	IG:IG D.F Scenario 2 Path 2, split Key Confirmation:No Key Derivation:IG 2.4.B SP 800-135rev1 CVL Caveat:Key establishment methodology providing 128, 192, or 256 bits of security strength	KAS-ECC-SSC Sp800-56Ar3: (A3453) KDF IKEv2: (A3453) SHA2-256: (A3453) SHA2-384: (A3453) SHA2-512: (A3453)
KAS-ECC (SSH)	KAS-135KDF KAS-SSC	Full KAS-ECC Key Agreement used for SSHv2 service	IG:IG D.F Scenario 2 Path 2, split Key Confirmation:No Key Derivation:No Caveat:Key establishment methodology providing between 128 and 256 bits of security strength	KAS-ECC-SSC Sp800-56Ar3: (A3453) KDF SSH: (A3453)
KAS-ECC (TLSv1.2)	KAS-135KDF KAS-SSC	Full KAS-ECC Key Agreement used for TLSv1.2 service	IG:IG D.F Scenario 2 Path 2, split Key Confirmation:No Key Derivation:No Caveat:Key establishment methodology providing between 128 and 256 bits of security strength	KAS-ECC-SSC Sp800-56Ar3: (A3453) TLS v1.2 KDF RFC7627: (A3453)
KAS-ECC-KeyGen (IPSec/IKE)	CKG KAS-KeyGen	KAS ECC keygen used in IPSec/IKEv2 service	Strength:P-256, P-384, and P-521 curves providing 128, 192, or 256 bits of encryption strength	Counter DRBG: (A3453) CKG: () Key Type: Symmetric and Asymmetric
KAS-ECC-KeyGen (SSH)	CKG KAS-KeyGen	KAS ECC keygen used in SSHv2 service	Strength:P-256, P-384, and P-521 curves providing	Counter DRBG: (A3453) ECDSA KeyGen

Name	Type	Description	Properties	Algorithms
			128, 192, or 256 bits of encryption strength	(FIPS186-4): (A3453) CKG: () Key Type: Symmetric and Asymmetric
KAS-ECC-KeyGen (TLSv1.2)	CKG KAS-KeyGen	KAS ECC keygen used in TLSv1.2 service	Strength:P-256, P-384, and P-521 curves providing 128, 192, or 256 bits of encryption strength	Counter DRBG: (A3453) ECDSA KeyGen (FIPS186-4): (A3453) CKG: () Key Type: Symmetric and Asymmetric
KAS-FFC (IPSec/IKE)	KAS-Full	Full KAS-FFC Key Agreement used for IPSec/IKEv2 service	IG:IG D.F Scenario 2 Path 2, split Key Confirmation:No Key Derivation:IG 2.4.B SP 800-135rev1 CVL Caveat:Key establishment methodology providing between 112 and 150 bits of security strength.	KAS-FFC-SSC Sp800-56Ar3: (A3453) KDF IKEv2: (A3453) SHA2-256: (A3453) SHA2-384: (A3453) SHA2-512: (A3453)
KAS-FFC (SSH)	KAS-135KDF KAS-SSC	Full KAS-FFC Key Agreement used for SSHv2 service	IG:IG D.F Scenario 2 Path 2, split Key Confirmation:No Key Derivation:IG 2.4.B SP 800-135rev1 CVL Caveat:Key establishment methodology providing 112 bits of security strength	KAS-FFC-SSC Sp800-56Ar3: (A3453) KDF SSH: (A3453)
KAS-FFC (TLSv1.2)	KAS-135KDF KAS-SSC	Full KAS-FFC Key Agreement used for TLSv1.2 service	IG:IG D.F Scenario 2 Path 2, split Key Confirmation:No Key Derivation:IG 2.4.B SP 800-135rev1 CVL Caveat:Key establishment methodology	KAS-FFC-SSC Sp800-56Ar3: (A3453) TLS v1.2 KDF RFC7627: (A3453) Safe Primes Key Generation: (A3453) Safe Primes Key

Name	Type	Description	Properties	Algorithms
			providing 112 bits of security strength	Verification: (A3453)
KAS-FFC-KeyGen (IPSec/IKE)	CKG KAS-KeyGen	KAS FFC keygen used in IPSec/IKEv2 service	Strength:2048, 3072, and 4096-bit keys providing 112, 128, or 150 bits of encryption strength	Counter DRBG: (A3453) CKG: () Key Type: Symmetric and Asymmetric Safe Primes Key Generation: (A3453) Safe Primes Key Verification: (A3453)
KAS-FFC-KeyGen (SSH)	CKG KAS-KeyGen	KAS FFC keygen used in SSHv2 service	Strength:2048-bit key providing 112 bits of encryption strength	Counter DRBG: (A3453) CKG: () Key Type: Symmetric and Asymmetric Safe Primes Key Generation: (A3453) Safe Primes Key Verification: (A3453)
KAS-FFC-KeyGen (TLSv1.2)	CKG KAS-KeyGen	KAS FFC keygen used in TLSv1.2 service	Strength:2048-bit key providing 112 bits of encryption strength	Counter DRBG: (A3453) CKG: () Key Type: Symmetric and Asymmetric Safe Primes Key Generation: (A3453) Safe Primes Key Verification: (A3453)
KTS (SSHv2 with AES and HMAC)	KTS-Wrap	KTS via SSHv2 service by using AES and HMAC	Standard:SP 800-38F IG D.G:Approved Key Wrapping Caveat:Key establishment methodology providing between 128 and 256 bits of security strength	AES-CBC: (A3453) HMAC-SHA2-256: (A3453) HMAC-SHA2-384: (A3453) SHA2-256: (A3453) SHA2-384: (A3453)
KTS (SSHv2 with AES-GCM)	KTS-Wrap	KTS via SSHv2 service by using AES-GCM	Standard:SP 800-38F IG D.G:Approved Key Wrapping Caveat:Key	AES-GCM: (A3453)

Name	Type	Description	Properties	Algorithms
			establishment methodology providing between 128 and 256 bits of security strength	
KTS (TLSv1.2 with AES and HMAC)	KTS-Wrap	KTS via TLSv1.2 service by using AES and HMAC	Standard:SP 800-38F IG D.G:Approved Key Wrapping Caveat:Key establishment methodology providing between 128 and 256 bits of security strength	AES-CBC: (A3453) HMAC-SHA2-256: (A3453) HMAC-SHA2-384: (A3453) SHA2-256: (A3453) SHA2-384: (A3453)
KTS (TLSv1.2 with AES-GCM)	KTS-Wrap	KTS via TLSv1.2 service by using AES-GCM	Standard:SP 800-38F IG D.G:Approved Key Wrapping Caveat:Key establishment methodology providing between 128 and 256 bits of security strength	AES-GCM: (A3453)
Session Authentication (IPSec/IKE)	MAC	IPSec/IKE session authentication		HMAC-SHA-1: (A3453) HMAC-SHA2-256: (A3453) HMAC-SHA2-384: (A3453) HMAC-SHA2-512: (A3453) SHA-1: (A3453) SHA2-256: (A3453) SHA2-384: (A3453) SHA2-512: (A3453)
Session Authentication (SMPv3)	MAC	SNMPv3 session authentication		HMAC-SHA-1: (A3453) HMAC-SHA2-224: (A3453) SHA-1: (A3453) SHA2-224: (A3453)
Session Authentication (SSHv2)	MAC	SSHv2 session authentication		HMAC-SHA-1: (A3453) HMAC-SHA2-256: (A3453) HMAC-SHA2-

Name	Type	Description	Properties	Algorithms
				512: (A3453) SHA-1: (A3453) SHA2-256: (A3453) SHA2-512: (A3453)
Session Authentication (TLSv1.2)	MAC	TLSv1.2 session authentication		HMAC-SHA2-256: (A3453) HMAC-SHA2-384: (A3453) SHA2-256: (A3453) SHA2-384: (A3453)
Session Encryption/Decryption (IPSec/IKE)	BC-Auth BC-UnAuth	IPSec/IKE session protection		AES-CBC: (A3453) AES-CCM: (A3453) AES-GCM: (A3453)
Session Encryption/Decryption (SNMPv3)	BC-UnAuth	SNMPv3 session protection		AES-CFB1: (A3453) AES-CFB8: (A3453) AES-CFB128: (A3453)
Session Encryption/Decryption (SSH)	BC-Auth BC-UnAuth	SSHv2 session protection		AES-CBC: (A3453) AES-CTR: (A3453) AES-GCM: (A3453)
Session Encryption/Decryption (TLSv1.2)	BC-Auth BC-UnAuth	TLSv1.2 session protection		AES-CBC: (A3453) AES-GCM: (A3453)
SNMPv3 Keying Materials Development	KAS-135KDF	SNMPv3 session keying materials, used to derive SNMPv3 session keys		KDF SNMP: (A3453)
SSH ECDSA KeyGen	AsymKeyPair-KeyGen CKG	ECDSA KeyGen for SSHv2		ECDSA KeyGen (FIPS186-4): (A3453) Counter DRBG: (A3453) CKG: () Key Type: Symmetric and Asymmetric
SSH ECDSA SigGen	DigSig-SigGen	ECDSA SigGen for SSHv2		ECDSA SigGen (FIPS186-4): (A3453)

Name	Type	Description	Properties	Algorithms
SSH ECDSA SigVer	DigSig-SigVer	ECDSA SigVer for SSHv2		ECDSA KeyVer (FIPS186-4): (A3453) ECDSA SigVer (FIPS186-4): (A3453)
SSH RSA KeyGen	AsymKeyPair-KeyGen CKG	RSA KeyGen for SSHv2		RSA KeyGen (FIPS186-4): (A3453) Counter DRBG: (A3453) CKG: () Key Type: Symmetric and Asymmetric
SSH RSA SigGen	DigSig-SigGen	ECDSA SigGen for SSHv2		RSA SigGen (FIPS186-4): (A3453)
SSH RSA SigVer	DigSig-SigVer	RSA SigVer for SSHv2		RSA SigVer (FIPS186-4): (A3453)
TLS ECDSA KeyGen	AsymKeyPair-KeyGen CKG	ECDSA KeyGen for TLSv1.2		ECDSA KeyGen (FIPS186-4): (A3453) Counter DRBG: (A3453) CKG: () Key Type: Symmetric and Asymmetric
TLS ECDSA SigGen	DigSig-SigGen	ECDSA SigGen for TLSv1.2		ECDSA SigGen (FIPS186-4): (A3453)
TLS ECDSA SigVer	DigSig-SigVer	ECDSA SigVer for TLSv1.2		ECDSA KeyVer (FIPS186-4): (A3453) ECDSA SigVer (FIPS186-4): (A3453)
TLS RSA KeyGen	AsymKeyPair-KeyGen CKG	RSA KeyGen for TLSv1.2		RSA KeyGen (FIPS186-4): (A3453) Counter DRBG: (A3453) CKG: () Key Type: Symmetric and Asymmetric
TLS RSA SigGen	DigSig-SigGen	RSA SigGen for TLSv1.2		RSA SigGen (FIPS186-4): (A3453)
TLS RSA SigVer	DigSig-SigVer	RSA SigVer for TLSv1.2		RSA SigVer (FIPS186-4): (A3453)

Table 6: Security Function Implementations

2.7 Algorithm Specific Information

The module is compliant to IG C.H: GCM is used in the context of TLS, and SSH:

● For TLS, The GCM implementation meets Scenario 1 of IG C.H: it is used in a manner compliant with SP800-52 and in accordance with Section 4 of RFC 5288 for TLS key establishment, and ensures when the nonce_explicit part of the IV exhausts all possible values for a given session key, that a new TLS handshake is initiated per sections 7.4.1.1 and 7.4.1.2 of RFC 5246. During operational testing, the module was tested against an independent version of TLS and found to behave correctly.

-From this RFC 5288, the GCM cipher suites in use are
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, and
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384.

● For IPsec/IKEv2, The GCM implementation meets Scenario 1 of IG C.H: it is used in a manner compliant with RFCs 4106 and 7296 (RFC 5282 is not applicable, as the module does not use GCM within IKEv2 itself) and ensures when the module exhausts all possible values for a given session key that this triggers a rekey condition. During operational testing, the module was tested against an independent version of IPsec with IKEv2 and found to behave correctly.

● For SSH, the module meets Scenario 1 of IG C.H. The module conforms to RFCs 4252, 4253, and 5647.
The fixed field is 4-byte in length and is derived using the SSH KDF; this ensures the fixed field is unique for any given GCM session. The invocation field is 8-byte in length and is incremented for each invocation of GCM; this prevents the IV from repeating until the entire invocation field space of 264 is exhausted, which can take hundreds of years. (In “Approved Mode” SSH rekey is automatically configured at 1 GB of data or 1 hour, whichever comes first.)

In all the above cases, the nonce_explicit is always generated deterministically. AES GCM keys are zeroized when the
module is power-cycled. For each new TLS or SSH session, a new AES GCM key is established.

The module does not have any algorithms that fall under:
- Non-Approved Algorithms Allowed in the Approved Mode of Operation
- Non-Approved Algorithms Allowed in the Approved Mode of Operation with No Security Claimed
- Non-Approved Algorithms Not Allowed in the Approved Mode of Operation.

The module is compliant to IG C.F:

The module utilizes Approved modulus sizes 2048, 3072, and 4096 bits for RSA signatures. This functionality has been CAVP tested as noted above. The minimum number of Miller Rabin

tests for each modulus size is implemented according to Table C.2 of FIPS 186-4. For modulus size 4096, the module implements the largest number of Miller-Rabin tests shown in Table C.2. RSA SigVer is CAVP tested for all three supported modulus sizes as noted above. The module does not perform FIPS 186-2 SigVer. All supported modulus sizes are CAVP testable and tested as noted above. The module does not implement

The module is compliant to IG C.K:
The CAVP testing for Cert. #A3453 was performed prior to the transition date for this IG. Additionally, The FIPS 186-4 CAVP implemented in this module tests are mathematically identical to FIPS 186-5 tests.

2.8 RBG and Entropy

Cert Number	Vendor Name
E130	Palo Alto networks
E64	Palo Alto networks

Table 7: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Palo Alto Networks Intel DRNG SP800-90B Compliance Report for Ice Lake 28-Core Die with FCLGA4189 Package	Physical	Intel Xeon (Silver) 4316 - Ice Lake	128 bits	128 bits	A2518 (AES-CBC-MAC)
Palo Alto Networks RTC Entropy Source	Physical	Intel Xeon E5-2620 - Sandy Bridge	80 bits	40.5555 bits	A2153 (AES-CBC-MAC)

Table 8: Entropy Sources

For the WF-500-B, the ESV Cert. #E64 entropy source provides full entropy, which is provided in the 384 bit seed.

For the WF-500, the ESV Cert. #E130 entropy source provides 0.50694395678 bits of entropy per bit of output. The DRBG requests 384 bits of data from the entropy source, so is seeded with at least 194 bits of entropy. The module generates SSPs (e.g., keys) whose strengths are modified by available entropy

2.9 Key Generation

The module implements CKG where symmetric keys and seeds used for asymmetric key pair generation are produced using the unmodified/direct output of the DRBG

2.10 Key Establishment

The module provides the following key/SSP establishment services in the Approved mode of operation:

- KAS-ECC Shared Secret Computation
    - The module provides SP800-56Arev3 compliant key establishment according to FIPS 140-3 IG D.F scenario 2 path (1) with KAS-ECC shared secret computation. The shared secret computation provides between 128 and 256 bits of encryption strength.
- KAS-FFC Shared Secret Computation
    - The module provides SP800-56Arev3 compliant key establishment according to FIPS 140-3 IG D.F scenario 2 path (1) with KAS-FFC shared secret computation. The shared secret computation provides between 112 and 150 bits of encryption strength.

2.11 Industry Protocols

- TLS 1.2
- SSHv2
- SNMPv3
- IPSec and IKEv2

*Note: No parts of these protocols, other than the approved cryptographic algorithms and the KDFs, have been tested by the CAVP and CMVP.

3 Cryptographic Module Interfaces

The modules are multi-chip standalone modules with ports and interfaces as shown below. The modules do not implement a control output interface.

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
LED	Status Output	Module status via LED indicators
Power	Power	N/A
RJ45 Console	Status Output	Self-test output
RJ45 Ethernet	Data Input Data Output Control Input Status Output	TLS, IPSec or SSH
SFP+ (WF-500-B)	Data Input Data Output Control Input Status Output	TLS

Table 9: Ports and Interfaces

From Web Cryptik

4 Roles, Services, and Authentication

4.1 Authentication Methods

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
RSA-Based Certificate	The modules support RSA public-key based authentication mechanism using a minimum of RSA 2048 bits	Single-Factor Cryptographic Software	With a minimum modulus size of 2048, the probability that a random attempt will succeed is 1/(2^112).	The probability of successfully authenticating to the module within a one minute period is 288,000,000/(2112). The module supports at most 4,800,000 new sessions per second.
ECDSA-Based Certificate	The modules support ECDSA public-key based authentication mechanism using a minimum ECDSA curve of P-256	Single-Factor Cryptographic Software	With a minimum curve of P-256, the probability that a random attempt will succeed is 1/(2^128).	The probability of successfully authenticating to the module within a one minute period is 288,000,000/(2112). The module supports at most 4,800,000 new sessions per second.
Password	Password based authentication	Memorized Secret (Unique Username/password)	The minimum length is eight (8) characters (95 possible characters). The probability that a random attempt will succeed or a false acceptance will occur is 1/(958).	The probability of successfully authenticating to the module within one minute is 10/(958). The firewall's configuration supports at most ten failed attempts to authenticate in a one-minute period.

Table 10: Authentication Methods

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Identity	CO	RSA-Based Certificate ECDSA-Based Certificate Password
Peer-to-peer VPN	Identity	CO	RSA-Based Certificate ECDSA-Based Certificate Password
User	Identity	User	RSA-Based Certificate ECDSA-Based Certificate Password

Table 11: Roles

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Data Analysis Management	Networking parameter configuration, logging configuration, and other non-security relevant configuration	Configuration,System Logs	Input configurations for other setup functions	Module uses configuration	KAS-ECC (SSH) KAS-ECC (TLSv1.2) KAS-ECC-KeyGen (SSH) KAS-ECC-KeyGen (TLSv1.2) KAS-FFC (IPSec/IKE) KAS-FFC (SSH) KAS-FFC (TLSv1.2) KAS-FFC-KeyGen (SSH) KAS-FFC-KeyGen (TLSv1.2) KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) KTS (TLSv1.2 with AES and HMAC) KTS (TLSv1.2 with AES-GCM) Session Authentication (SMPv3) Session Authentication (SSHv2) Session Authentication (TLSv1.2) Session Encryption/Decryption (SNMPv3) Session Encryption/Decryption (SSH) Session Encryption/Decryption (TLSv1.2)	Crypto Officer - AES-GCM IV: G,E,Z - CO, User Password: G,E,W - DRBG Key: G,W,E - DRBG Seed: G,E - DRBG V: G,W,E - ECDSA Private Keys: G,W,E - Entropy Input String: G,E - RSA Private Keys: G,W,E - SSH Client Public Key: W,E - SSH DHE/ECDHE Private Components: G,E,Z - SSH DHE/ECDHE Public Components: G,E,R,W,Z - SSH Host Public Key: G,R,E,W - SSH Session Authentication Keys: G,E,Z - SSH Session

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					SNMPv3 Keying Materials Development SSH ECDSA KeyGen SSH ECDSA SigGen SSH ECDSA SigVer SSH RSA KeyGen SSH RSA SigGen SSH RSA SigVer TLS ECDSA KeyGen TLS ECDSA SigGen TLS ECDSA SigVer TLS RSA KeyGen TLS RSA SigGen TLS RSA SigVer	Encryption Keys: G,E,Z - SSH Shared Secret: G,E,Z - TLS DHE/ECDHE Private Components: G,E,Z - TLS DHE/ECDHE Public Components: G,E,R,W,Z - TLS Encryption Keys: G,E,Z - TLS HMAC Keys: G,E,Z - TLS Master Secret: G,E,Z - TLS Pre-Master Secret: G,E,Z
Firmware Update	Provides a method to update the firmware of the module	Configuration,System Logs	Uploading new firmware	Status of the updated firmware installation	Firmware Load Test	Crypto Officer - Public key for firmware content load test: E
IKE/IPsec configuration	Configures IKE/IPsec setup for peer to peer VPN.	Configuration,System	Initialize VPN connection	Confirmation of service via System Logs	CKG - Symmetric IPSec/IKE ECDSA KeyGen IPSec/IKE ECDSA SigGen IPSec/IKE ECDSA SigVer IPSec/IKE Keying Materials Development IPSec/IKE RSA KeyGen IPSec/IKE RSA SigGen	Crypto Officer - AES-GCM IV: G,E - CO, User Password: E - DRBG Key: G,W,E - DRBG Seed: G,E - DRBG V: G,W,E - Entropy Input String: G,E - IKEv2

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					IPSec/IKE RSA SigVer KAS-ECC (IPSec/IKE) KAS-ECC-KeyGen (IPSec/IKE) KAS-FFC (IPSec/IKE) KAS-FFC-KeyGen (IPSec/IKE) Session Authentication (IPSec/IKE) Session Encryption/Decryption (IPSec/IKE)	SKEYSEED: G,E - IPSec/IKE Authentication Keys: G,E,Z - IPSec/IKE DHE/ECDHE Private Components: G,E,Z - IPSec/IKE DHE/ECDHE Public Components: G,R,W,E,Z - IPSec/IKE Session Keys: G,E,Z - IPSec/IKE Shared Secret: G,E,Z Peer-to-peer VPN - AES-GCM IV: G,E - CO, User Password: E - DRBG Key: G,W,E - DRBG Seed: G,E - DRBG V: G,W,E - Entropy Input String: G,E - IKEv2 SKEYSEED: G,E - IPSec/IKE Authentication Keys: G,E,Z - IPSec/IKE DHE/ECDHE Private Components: G,E,Z - IPSec/IKE DHE/ECDH

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						E Public Components: G,R,W,E,Z - IPSec/IKE Session Keys: G,E,Z - IPSec/IKE Shared Secret: G,E,Z
Self-Tests	Initiates self-tests and integrity test	System Logs	Self-test command or rebooting the module	Status of the self-tests	None	Crypto Officer
Show Status	Provides status information of the module	Configuration,System Logs	Initiate show status command	Module provides status output of module	None	Crypto Officer - AES-GCM IV: G,E,Z - CO, User Password: E - DRBG Key: G,W,E - DRBG Seed: G,E - DRBG V: G,W,E - ECDSA Private Keys: E - Entropy Input String: G,E - RSA Private Keys: E - SSH DHE/ECDHE Private Components: G,E,Z - SSH DHE/ECDHE Public Components: G,E,R,W,Z - SSH Session Authentication Keys:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						G,E,Z - SSH Session Encryption Keys: G,E,Z - SSH Shared Secret: G,E,Z - TLS DHE/ECDHE Private Components: G,E,Z - TLS DHE/ECDHE Public Components: G,E,R,W,Z - TLS Encryption Keys: G,E,Z - TLS HMAC Keys: G,E,Z - TLS Master Secret: G,E,Z - TLS Pre-Master Secret: G,E,Z Unauthenticated User - AES-GCM IV: G,E,Z - CO, User Password: E - DRBG Key: G,W,E - DRBG Seed: G,E - DRBG V: G,W,E - ECDSA Private Keys: E - Entropy Input String: G,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- RSA Private Keys: E - SSH DHE/ECDH E Private Component s: G,E,Z - SSH DHE/ECDH E Public Component s: G,E,R,W,Z - SSH Session Authenticati on Keys: G,E,Z - SSH Session Encryption Keys: G,E,Z - SSH Shared Secret: G,E,Z - TLS DHE/ECDH E Private Component s: G,E,Z - TLS DHE/ECDH E Public Component s: G,E,R,W,Z - TLS Encryption Keys: G,E,Z - TLS HMAC Keys: G,E,Z - TLS Master Secret: G,E,Z - TLS Pre-Master Secret: G,E,Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Show Status (LEDs)	Provides status of the module	LEDs	N/A	Status of the module via LEDs	None	Crypto Officer Unauthenticated User
Show Version	Shows the version of the module	Version displayed via System Logs / CLI / UI	Input command for version	Module displays version information	None	Crypto Officer Unauthenticated
System Audit	Allows review of limited configuration and system status via logs, dashboard and configuration screens. Provides no configuration commit capability.	System logs are displayed via CLI	View the System Logs via CLI	System Logs	CKG - Symmetric KAS-ECC (SSH) KAS-ECC-KeyGen (SSH) KAS-FFC (SSH) KAS-FFC-KeyGen (SSH) KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) Session Authentication (SSHv2) Session Encryption/Decryption (SSH) SSH ECDSA KeyGen SSH ECDSA SigGen SSH ECDSA SigVer SSH RSA KeyGen SSH RSA SigGen SSH RSA SigVer	Crypto Officer - CO, User Password: G,W,E - DRBG Key: G,W,E - DRBG Seed: G,E - DRBG V: G,W,E - Entropy Input String: G,E - SSH Client Public Key: W,E - SSH DHE/ECDHE Private Components: G,E,Z - SSH DHE/ECDHE Public Components: G,R,W,E,Z - SSH Host Public Key: R,E - SSH Session Authentication Keys: G,E,Z - SSH Session Encryption Keys: G,E,Z - SSH Shared

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Secret: G,E,Z
System Configuration Management	Configuring and managing cryptographic parameters and setting,modifying security policy, including creating User accounts and additional CO accounts	Configuration, System Logs	Input configuration for various cryptographic functions	Module uses the configuration for cryptographic purposes	CKG - Symmetric IPSec/IKE ECDSA KeyGen IPSec/IKE ECDSA SigGen IPSec/IKE ECDSA SigVer IPSec/IKE Keying Materials Development IPSec/IKE RSA KeyGen IPSec/IKE RSA SigGen IPSec/IKE RSA SigVer KAS-ECC (IPSec/IKE) KAS-ECC (SSH) KAS-ECC (TLSv1.2) KAS-ECC-KeyGen (IPSec/IKE) KAS-ECC-KeyGen (SSH) KAS-ECC-KeyGen (TLSv1.2) KAS-FFC (IPSec/IKE) KAS-FFC (SSH) KAS-FFC (TLSv1.2) KAS-FFC-KeyGen (IPSec/IKE) KAS-FFC-KeyGen (SSH) KAS-FFC-KeyGen (TLSv1.2) KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) KTS (TLSv1.2	Crypto Officer - AES-GCM IV: G,E - CA Certificates: G,R,E,W - CO, User Password: G,E,W - DRBG Key: G,W,E - DRBG Seed: G,E - DRBG V: G,W,E - ECDSA Private Keys: G,W,E - ECDSA Public Keys: G,R,E,W - Entropy Input String: G,E - IKEv2 SKEYSEED: G,E - Protocol Secrets: W,E - Public key for firmware content load test: W,E - RSA Private Keys: G,W,E - RSA Public Keys: G,R,E,W - SNMPv3 Authentication Key: G,E,Z - SNMPv3 Authentication Secret:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					with AES and HMAC) KTS (TLSv1.2 with AES-GCM) Session Authentication (SMPv3) Session Authentication (SSHv2) Session Authentication (TLSv1.2) SNMPv3 Keying Materials Development SSH ECDSA KeyGen SSH ECDSA SigGen SSH ECDSA SigVer SSH RSA KeyGen SSH RSA SigGen SSH RSA SigVer TLS ECDSA KeyGen TLS ECDSA SigGen TLS ECDSA SigVer TLS RSA KeyGen TLS RSA SigGen TLS RSA SigVer	W,E - SNMPv3 Privacy Secret: W,E - SNMPv3 Session Key: G,E,Z - SSH Client Public Key: W,E - SSH DHE/ECDHE Private Components: G,E,Z - SSH DHE/ECDHE Public Components: G,E,R,W,Z - SSH Host Public Key: G,R,E,W - SSH Session Authentication Keys: G,E,Z - SSH Session Encryption Keys: G,E,Z - SSH Shared Secret: G,E,Z - TLS DHE/ECDHE Private Components: G,E,Z - TLS DHE/ECDHE Public Components: G,E,R,W,Z - TLS Encryption Keys: G,E,Z - TLS HMAC

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Keys: G,E,Z - TLS Master Secret: G,E,Z - TLS Pre-Master Secret: G,E,Z
System Operational Management	Configuring and managing networking parameter configuration, logging configuration, and other non-security relevant configuration via CLI	Configuration,System Logs	Input configuration for various cryptographic functions	Module uses the configuration for cryptographic purposes	IPSec/IKE ECDSA KeyGen IPSec/IKE ECDSA SigGen IPSec/IKE ECDSA SigVer IPSec/IKE Keying Materials Development IPSec/IKE RSA KeyGen IPSec/IKE RSA SigGen IPSec/IKE RSA SigVer KAS-ECC (IPSec/IKE) KAS-ECC (SSH) KAS-ECC (TLSv1.2) KAS-ECC-KeyGen (IPSec/IKE) KAS-ECC-KeyGen (SSH) KAS-ECC-KeyGen (TLSv1.2) KAS-FFC (IPSec/IKE) KAS-FFC (SSH) KAS-FFC (TLSv1.2) KAS-FFC-KeyGen (IPSec/IKE) KAS-FFC-KeyGen (SSH) KAS-FFC-KeyGen (TLSv1.2) KTS (SSHv2	Crypto Officer - AES-GCM IV: G,E - CA Certificates: G,R,E,W - CO, User Password: G,E,W - DRBG Key: G,W,E - DRBG Seed: G,E - DRBG V: G,W,E - ECDSA Private Keys: G,W,E - ECDSA Public Keys: G,R,E,W - Entropy Input String: G,E - IKEv2 SKEYSEED: G,E - Protocol Secrets: W,E - Public key for firmware content load test: W,E - RSA Private Keys: G,W,E - RSA Public Keys: G,R,E,W

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					with AES and HMAC) KTS (SSHv2 with AES-GCM) KTS (TLSv1.2 with AES and HMAC) KTS (TLSv1.2 with AES-GCM) Session Authentication (IPSec/IKE) Session Authentication (SMPv3) Session Authentication (SSHv2) Session Authentication (TLSv1.2) Session Encryption/Decryption (IPSec/IKE) Session Encryption/Decryption (SNMPv3) Session Encryption/Decryption (SSH) Session Encryption/Decryption (TLSv1.2) SNMPv3 Keying Materials Development SSH ECDSA KeyGen SSH ECDSA SigGen SSH ECDSA SigVer SSH RSA KeyGen SSH RSA SigGen SSH RSA SigVer TLS ECDSA KeyGen TLS ECDSA	- SNMPv3 Authentication Secret: G,E,W,Z - SNMPv3 Privacy Secret: W,E - SNMPv3 Session Key: G,E,Z - SSH Client Public Key: W,E - SSH DHE/ECDHE Private Components: G,E,Z - SSH DHE/ECDHE Public Components: G,E,R,W,Z - SSH Host Public Key: G,R,E,W - SSH Session Authentication Keys: G,E,Z - SSH Session Encryption Keys: G,E,Z - SSH Shared Secret: G,E,Z - TLS DHE/ECDHE Private Components: G,E,Z - TLS DHE/ECDHE Public Components: G,E,R,W,Z - TLS Encryption

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					SigGen TLS ECDSA SigVer TLS RSA KeyGen TLS RSA SigGen TLS RSA SigVer	Keys: G,E,Z - TLS HMAC Keys: G,E,Z - TLS Master Secret: G,E,Z - TLS Pre-Master Secret: G,E,Z
Zeroize	Zeroizes all keys in the module	Zeroization indicator	Initiating zeroization command	Status of the zeroization process	None	Unauthenticated - AES-GCM IV: Z - CA Certificates: Z - CO, User Password: Z - DRBG Key: Z - DRBG Seed: Z - DRBG V: Z - ECDSA Private Keys: Z - ECDSA Public Keys: Z - Entropy Input String: Z - Firmware integrity verification key: Z - IKEv2 SKEYSEED: Z - Protocol Secrets: Z - Public key for firmware content load test: Z - RSA Private Keys: Z - RSA

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Public Keys: Z - SNMPv3 Authentication Key: Z - SNMPv3 Authentication Secret: Z - SNMPv3 Privacy Secret: Z - SNMPv3 Session Key: Z - SSH Client Public Key: Z - SSH DHE/ECDHE Private Components: Z - SSH DHE/ECDHE Public Components: Z - SSH Host Public Key: Z - SSH Session Authentication Keys: Z - SSH Session Encryption Keys: Z - SSH Shared Secret: Z - TLS DHE/ECDHE Private Components: Z - TLS DHE/ECDHE Public Components: Z - TLS Encryption

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Keys: Z - TLS HMAC Keys: Z - TLS Master Secret: Z - TLS Pre-Master Secret: Z

Table 12: Approved Services

4.4 Non-Approved Services

N/A for this module.

4.5 External Software/Firmware Loaded

The module supports the firmware load test by using RSA 2048 bits with SHA2-256 (RSA Cert. #A3453) for the new validated firmware to be uploaded into the module. A Firmware Load Test Key was preloaded to the module's binary at the factory and used for firmware load test. In order to load new firmware, the Crypto Officer must authenticate into the module before loading any firmware. This ensures that unauthorized access and use of the module is not performed. The module will load the new update upon reboot. The update attempt will be rejected if the verification fails.

5 Software/Firmware Security

5.1 Integrity Techniques

The module performs the Firmware Integrity test by using HMAC-SHA-256 and ECDSA signature verification (HMAC and ECDSA Cert. #A3453) during the Pre-Operational Self-Test. In addition, the module also conducts the firmware load test by using the Public Key for Firmware Load Test (RSA 2048 with SHA-256, Cert. #A3453) for the new validated firmware to be uploaded into the module via the System Operational Management service. The Firmware Integrity Verification Key and Public Key for Firmware Load Test used for the Firmware Integrity and Firmware Load test, respectively, are generated externally and delivered as part of the module firmware image.

The pre-operational self-tests can be initiated by power cycling the module. When this is performed, the module automatically runs the cryptographic algorithm self-tests in addition to the pre-operational firmware integrity test.

The module's executable code is in the form of the compiled firmware image loaded onto the module.

5.2 Initiate on Demand

The pre-operational self-tests can be initiated by power cycling the module. When this is performed, the module automatically runs the cryptographic algorithm self-tests in addition to the pre-operational firmware integrity test.

6 Operational Environment

6.1 Operational Environment Type and Requirements

The FIPS 140-3 Area 5 Operational Environment requirements are not applicable because the module contains a non-modifiable operational environment. The operational environment is limited since the module includes a firmware load service to support necessary updates. New firmware versions within the scope of this validation must be validated through the FIPS 140-3 CMVP. Any other firmware loaded into this module is out of the scope of this validation and requires a separate FIPS 140-3 validation.

Type of Operational Environment: Limited

7 Physical Security

7.1 Mechanisms and Actions Required

Mechanism	Inspection Frequency	Inspection Guidance
Tamper-Evident Seals	30 days	Verify integrity of tamper-evident seals in the locations specified in this section.
Front and Rear Opacity Shields	30 days	Verify that the front and rear opacity shields have not been deformed from their original shape, thereby reducing their effectiveness.
Vent Overlays	30 days	Verify that the vent overlays have not been removed or deformed. All edges should maintain strong adhesion characteristics.

Table 13: Mechanisms and Actions Required

The multi-chip standalone module is production quality and contains standard passivation. Chip components are protected by an opaque enclosure. There are tamper-evident seals that are applied on the module by the Crypto-Officer, and any unused seals are to be controlled by the Crypto-Officer. The Crypto-Officer must ensure that the module surface is clean and dry before applying the seals. The seals prevent removal of the opaque enclosure without evidence, which should be inspected by the Crypto-Officer every 30 days for evidence of tampering. If the seals or opacity shields show evidence of tamper, the Crypto-Officer should assume that the module has been compromised and contact Customer Support.

Note: For ordering information, physical kit part numbers and version. Opacity shields are included in the

physical kits.

7.2 User Placed Tamper Seals

WF-500 Tamper Seal Installation

Number: WF-500 Tamper Seal Installation (12 Seals)

Placement: WF-500 Tamper Seal Installation (12 Seals)

1. Remove the two pull handles and front modules on the left and right side of the appliance by removing the three (3) screws located behind each handle/module. There is no need to disconnect the LED circuit board attached to the end of the ribbon cable. Retain these screws for Step 2.

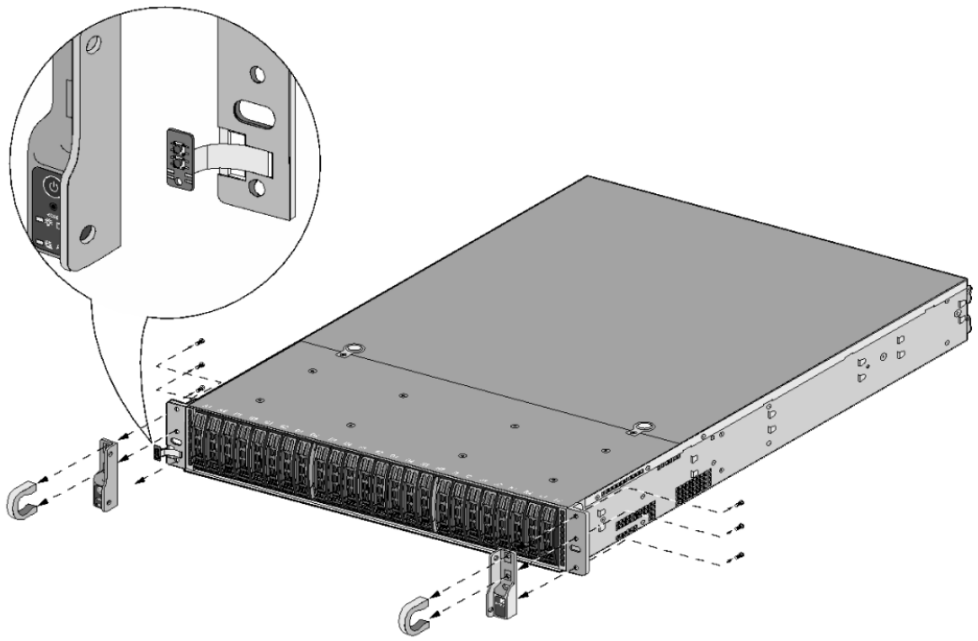

Figure 6 - Remove Front Handles and Modules

2. Attach the left and right front cover brackets to the appliance using the six (6) screws that were removed in Step 1. First attach the brackets using the bottom screws (one (1) on each side) as shown in Figure 6, ensuring that you feed the ribbon cable and LED circuit board through the left bracket. Replace the front modules and secure them using the middle and top screws on each side as shown in Figure 1.

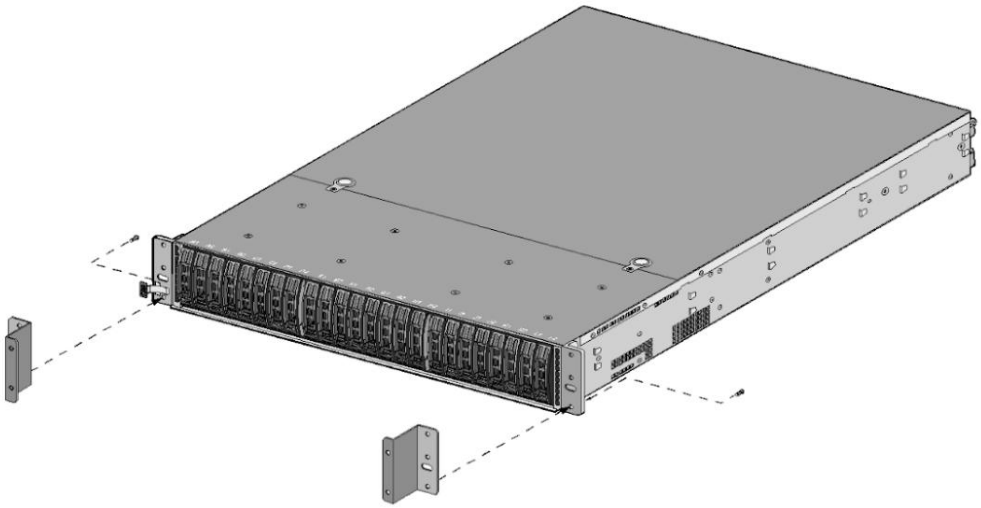

Figure 7 - Secure the Front Brackets

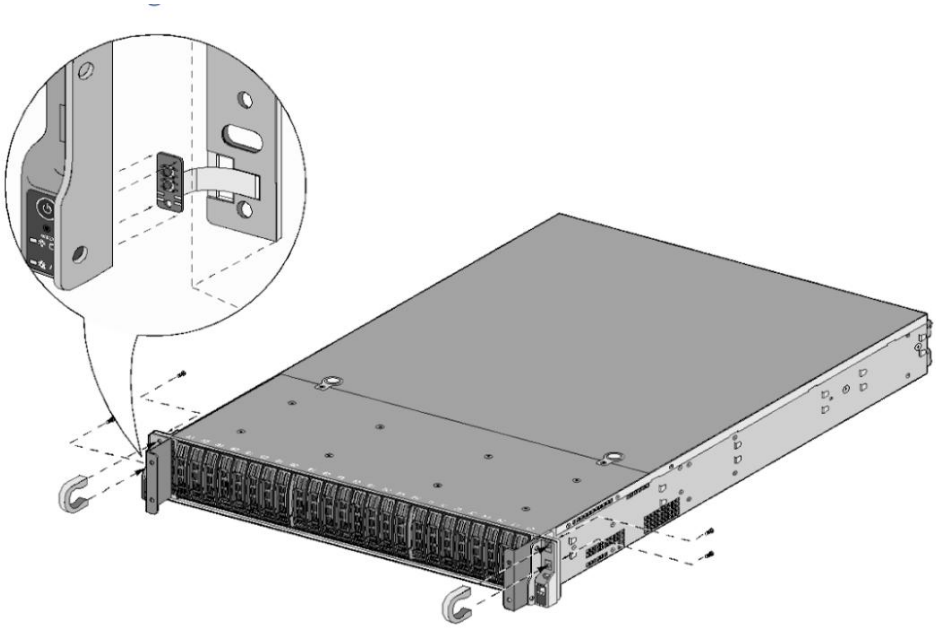

Figure 8 - Attach Pull Handles and Front Modules

3. Secure the front opacity shield to the right and left front brackets that you installed in Step 2. Use two (2) screws (provided) on each side.

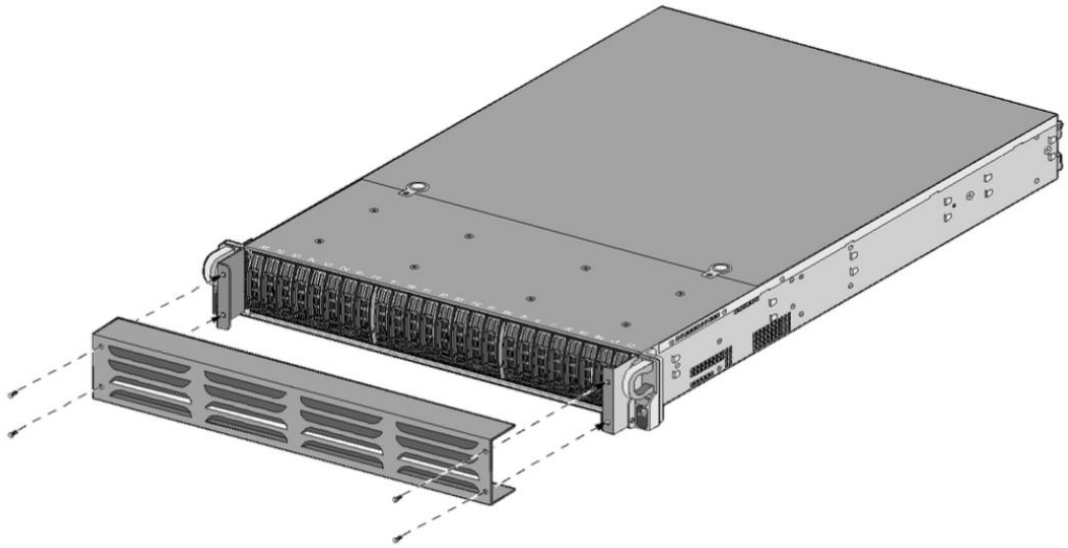

Figure 9 - Install Front Opacity Shield

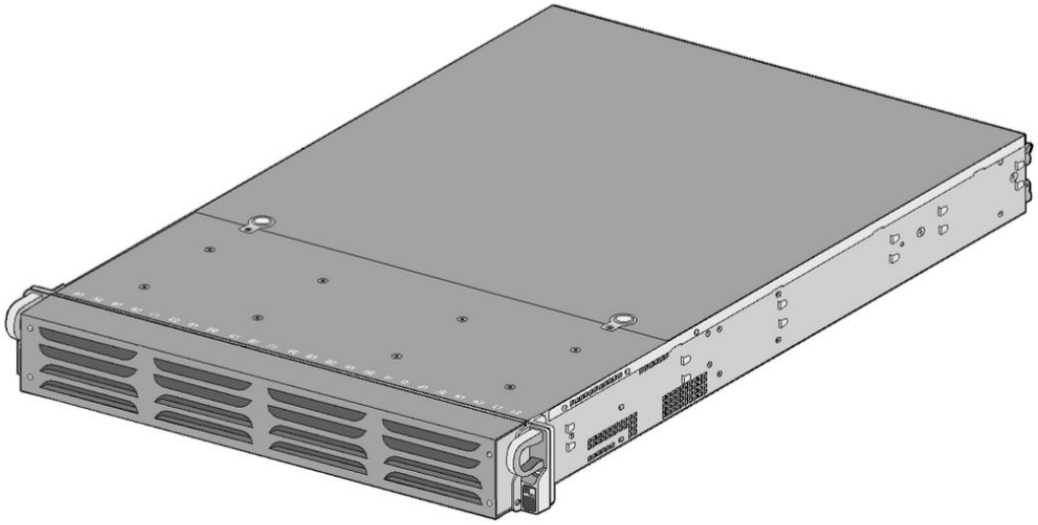

Figure 10 - Front Opacity Shield Installed

4. Attach the rear opacity shield tray to the appliance. appliance and use these screws to secure the rear opacity shield tray. First, remove the two (2) screws (shown in Figure 10) from the
Note: Install the back cables (power cords and network/management cables) because you will not be able to access these
ports after the next step.

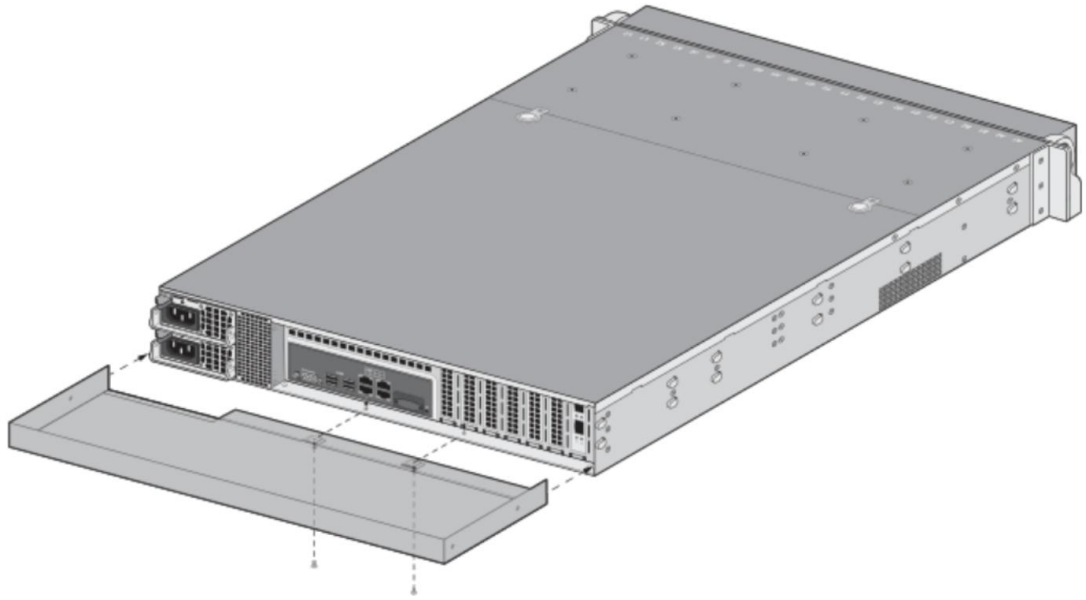

Figure 11 - Install Rear Opacity Shield Tray

5. Place the rear opacity shield on top of the rear opacity shield tray ensuring that you run the cables through the opening at the bottom. Secure the opacity shields with two (2) screws (provided) on each side.

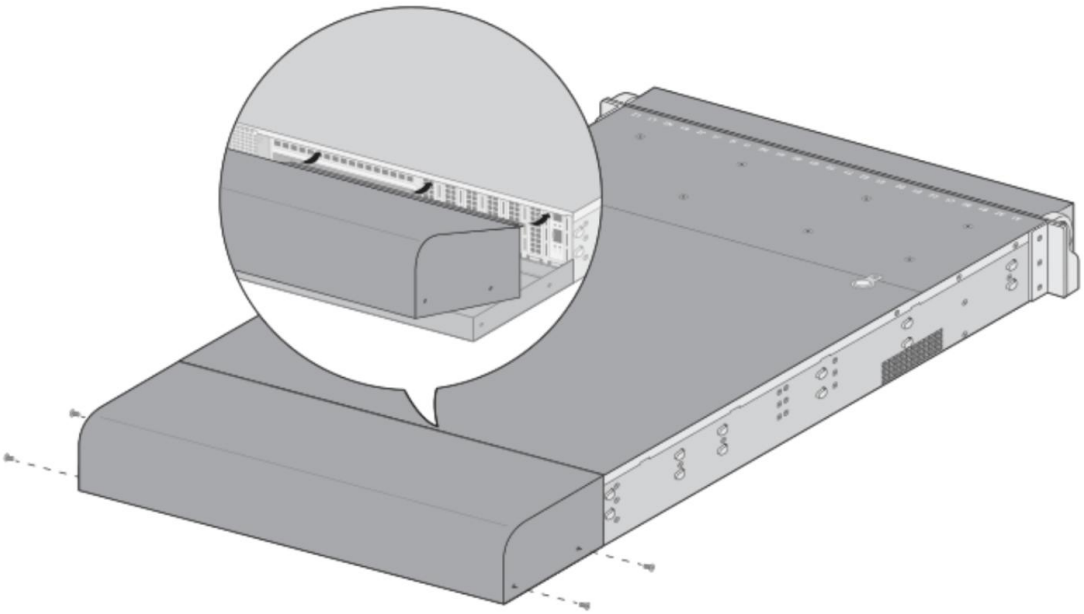

Figure 12 - Installed Rear Opacity Shield

6. Cover the vent openings as shown in Figure 12 by applying one (1) overlay tamper-evident seal over the left side vent and one overlay tamper-evident seal over the right side vent. Each overlay requires two (2) tamper-evident seals as shown in Figure 13. Also apply one (1) additional tamper-evident seal as shown in Figure 13.

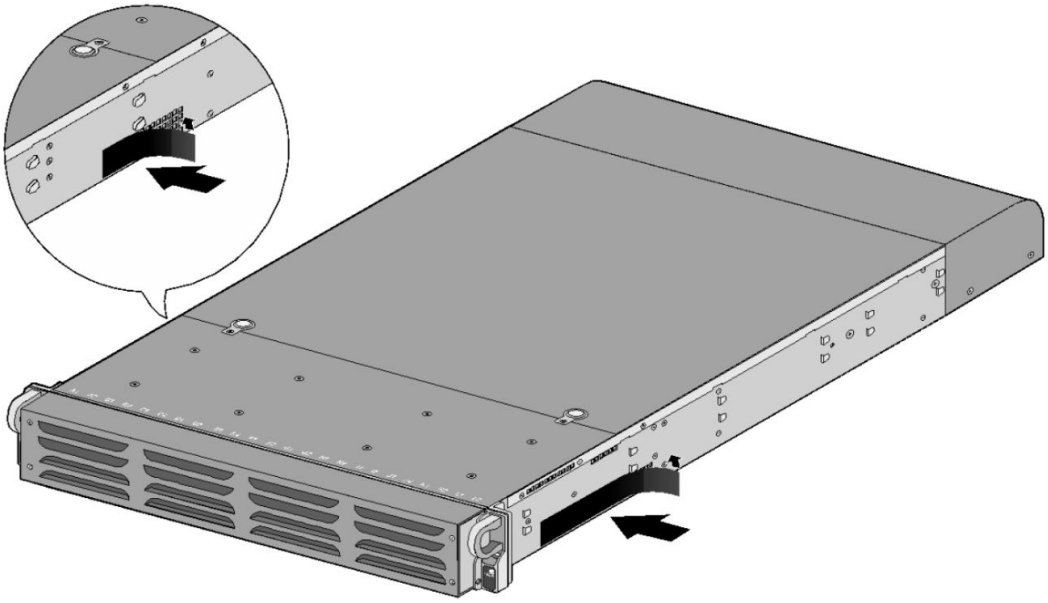

Figure 13 - Apply Tamper-Evident Seals on Vent Overlays

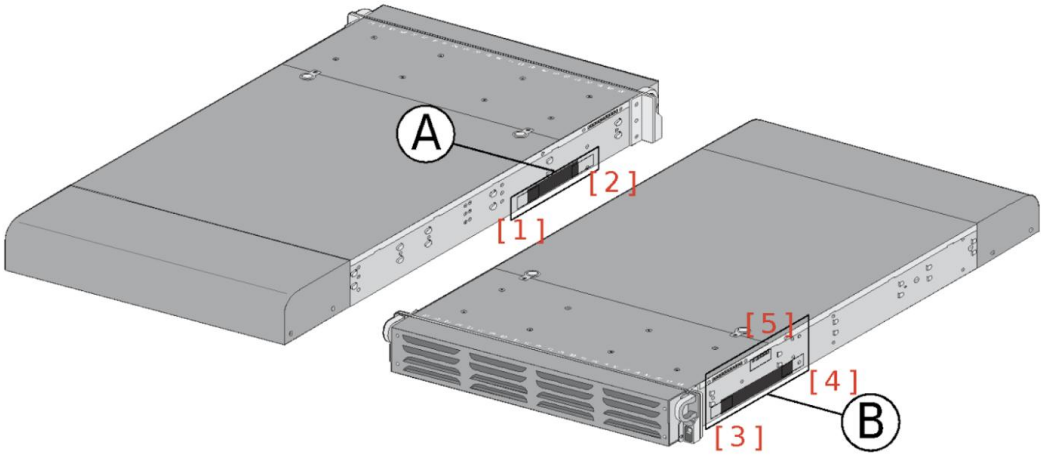

Figure 14 - Apply Tamper-Evident Seals on Vent Overlays and Side Opening

7. Attach the rail kit to the appliance as shown in Figure 14 and then add three (3) tamper-evident seals to the bottom of the appliance as shown in Figure 15. One (1) tamper-evident seal #6 prevents tampering of the front opacity shield connected to the bottom of the appliance and two (2) tamper-evident seals #7 and #8 wrap around the upper and lower rear opacity shields to prevent tampering of the rear opacity shields.

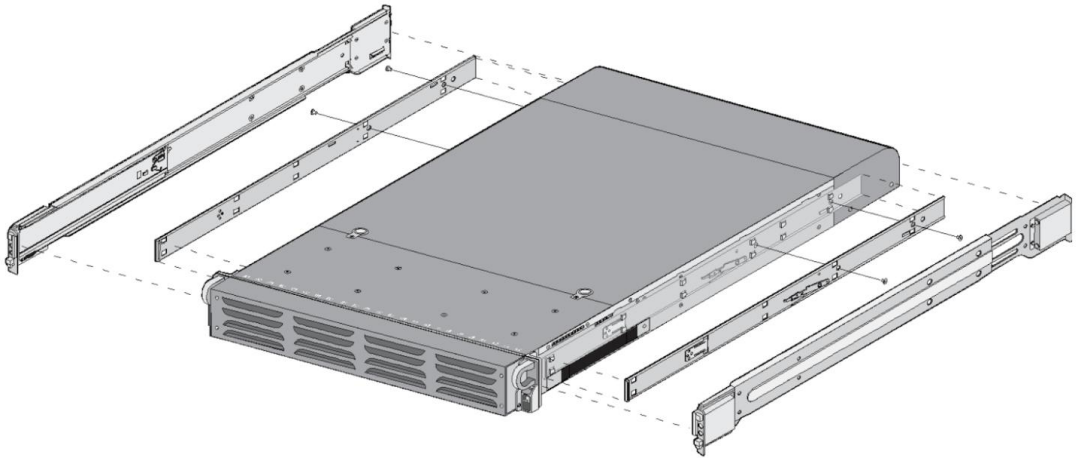

Figure 15 - Install Rail Kits

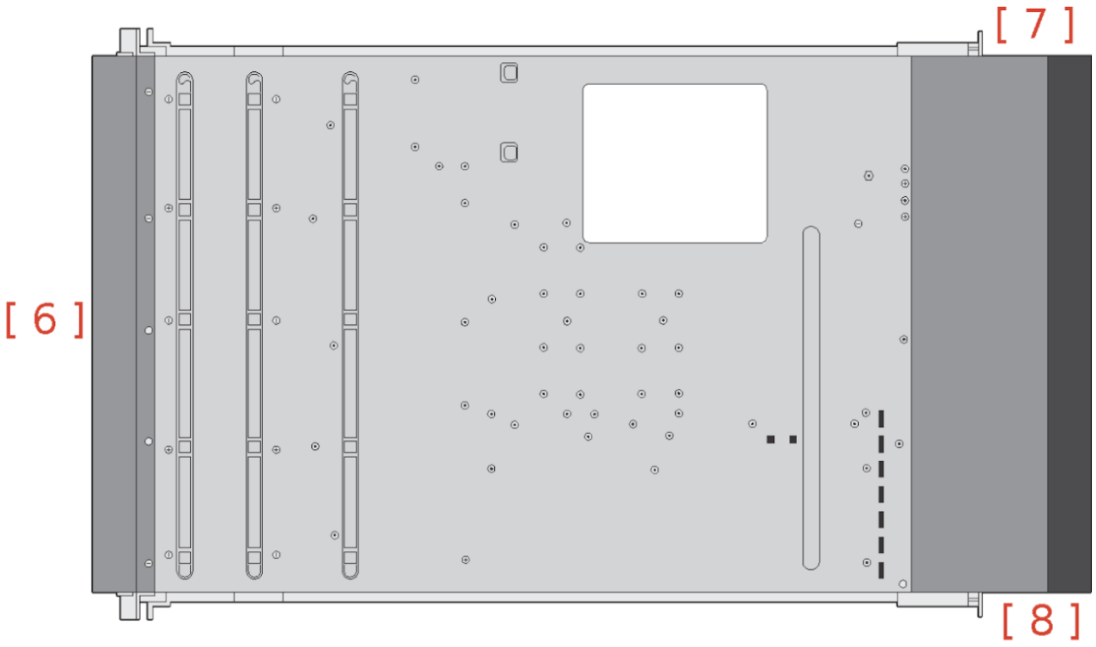

Figure 16 - Apply Tamper-Evident Seals on the Bottom of the Appliance

8. Place four (4) tamper seals on the top of the appliance. Two (2) tamper seals (#9 and #11) prevent tampering of the top front and rear opacity shields and two (2) tamper seals (#10 and #12) prevents someone from attempting to access the vent overlays by sliding the rail kit. This completes the physical kit installation.

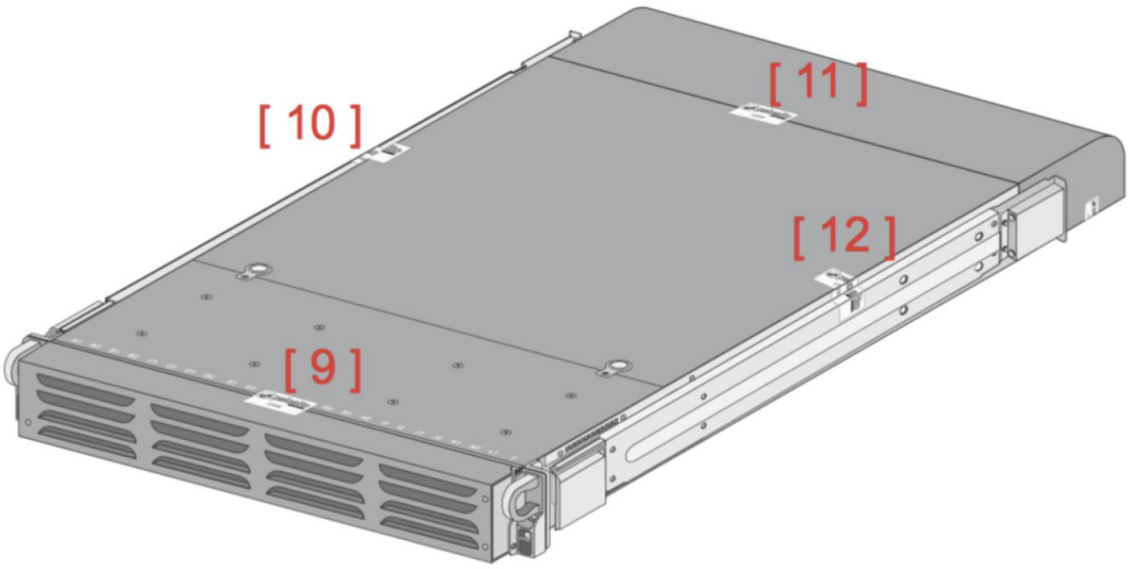

Figure 17 - Apply Tamper-Evident Seals on the Top and Sides of the Appliance

WF-500-B Tamper Seal Installation (21 Seals)

Number: WF-500-B Tamper Seal Installation (21 Seals)

Placement: WF-500 Tamper Seal Installation (21 Seals)

1. Replace the top cover with the physical kit top cover.
Remove the VOID WARRANTY label and cover screws (replacement label included in the kit).

Remove the Void Warranty label that covers the left side cover screw then use a Phillips-head screwdriver to remove both screws as indicated in the illustration.

Simultaneously depress the two (2) release buttons on top of the cover and slide the cover toward the back of the appliance to remove it.

Slide the physical kit top cover (does not have vents) on the appliance until the release buttons click. Replace the two screws that you removed from the old cover

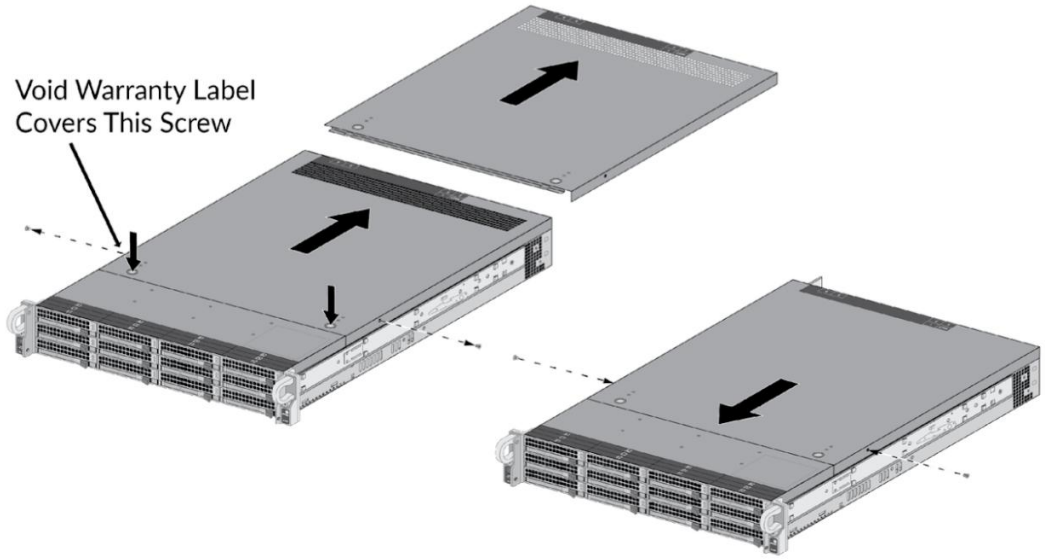

Figure 18 - WF-500-B: Top Cover Replacement

2. Attach the physical kit front cover brackets.
Remove the front pull handles by removing two (2) screws from each handle (one (1) handle on each side), insert the WF-500-B physical kit front-cover brackets under each handle, and then replace the handles and secure them using the screws that you removed. The physical kit handles have standoffs that are used to secure the front cover.

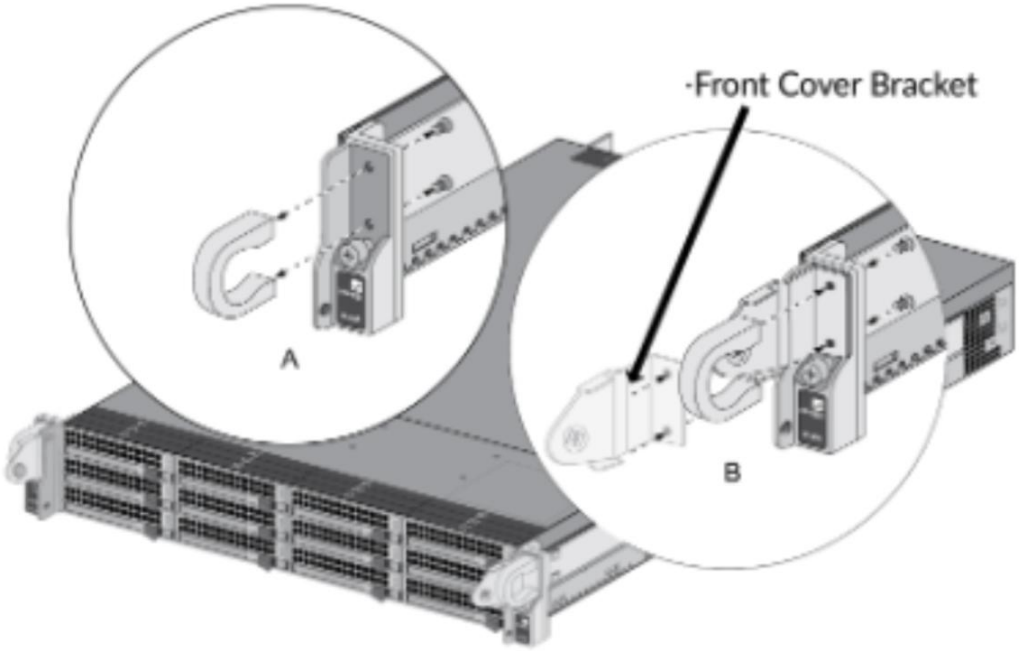

Figure 19 - WF-500-b: Front Cover Bracket

3. Attach the physical kit front cover to the front of the appliance
Slide the WF-500-B physical kit front cover over the physical kit pull handle brackets and secure the cover by turning the thumb screws clockwise (one thumb screw on each side).

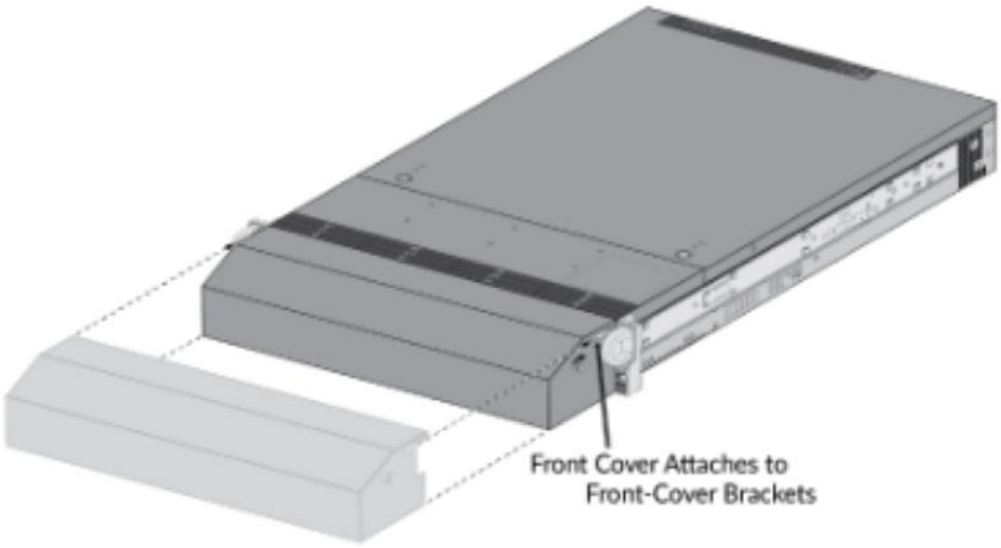

Figure 20 - WF-500-B: FIPS Front Cover

Install a tamper-evident seal on the back of the appliance. This is seal #13 in the WF-500-B Figure 19. You need to install this seal before you install the WF-500-B physical kit back cover.

Attach the physical kit back cover to the back of the appliance.

Slide the back cover onto the back of the appliance and turn the two (2) thumb screws clockwise until tight (one (1) screw on each side) to secure the cover.

Apply a tamper-evident seal to each location shown in the following WF-500-B illustrations below.

Also install the overlay stickers to cover vent openings (two (2) stickers on each side). You then install tamper-evident seals over the overlay stickers. Apply two (2) tamper-evident seals on the back side of the right rack handle (see seals #18 and #19 on the left side in Figure 19). Apply two (2) tamper-evident seals on the power supplies (see seals #11 and #12 with rear insert of Figure 21

Before you apply the tamper-evident seals, ensure that the appliance and physical kit surfaces are clean and dry. Firmly press one (1) seal on each of the locations shown in the illustrations. Avoid touching the seals for at least 24 hours to allow time for the seals to properly adhere to the appliance and physical kit surfaces.

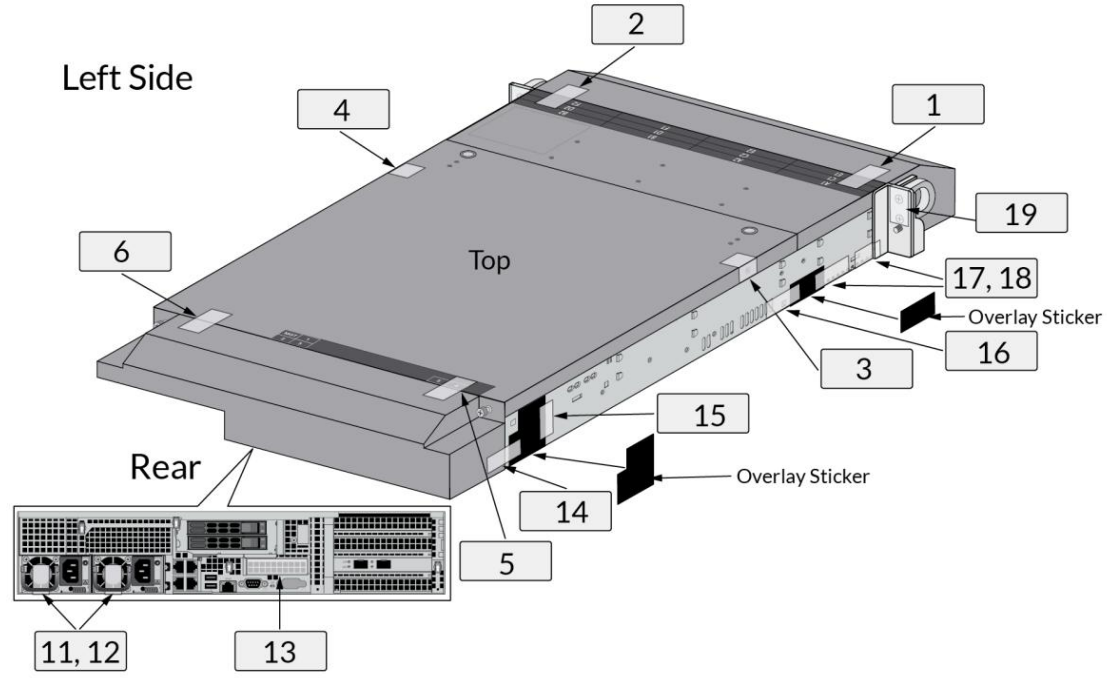

Figure 21 - WF-500-b: Tamper Seal Locations Top and Rear

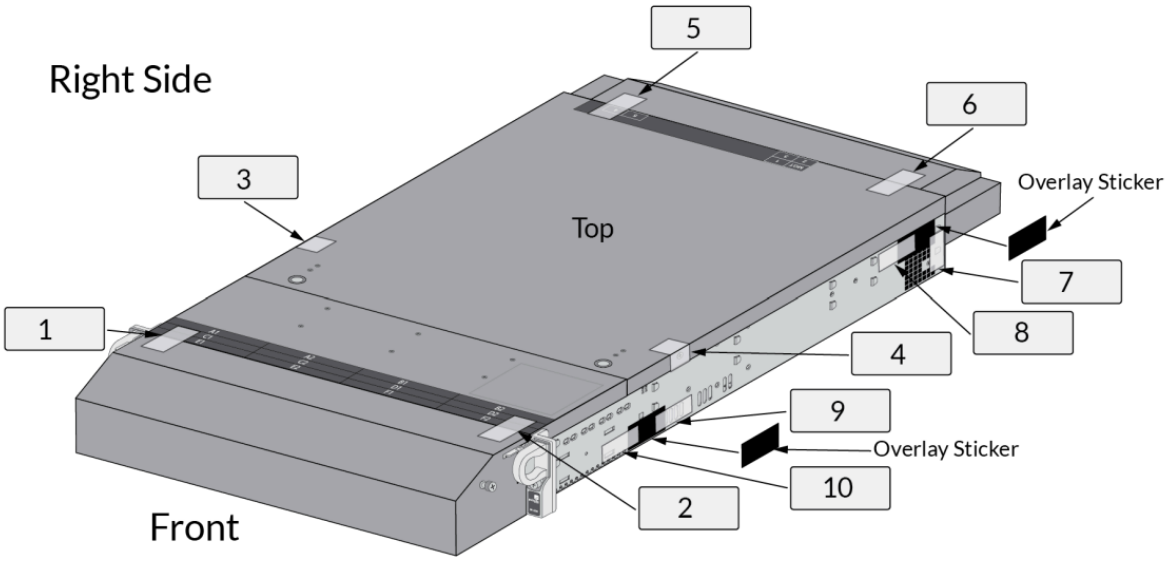

Figure 22 - WF-500-b: Tamper Seal Locations Top and Front

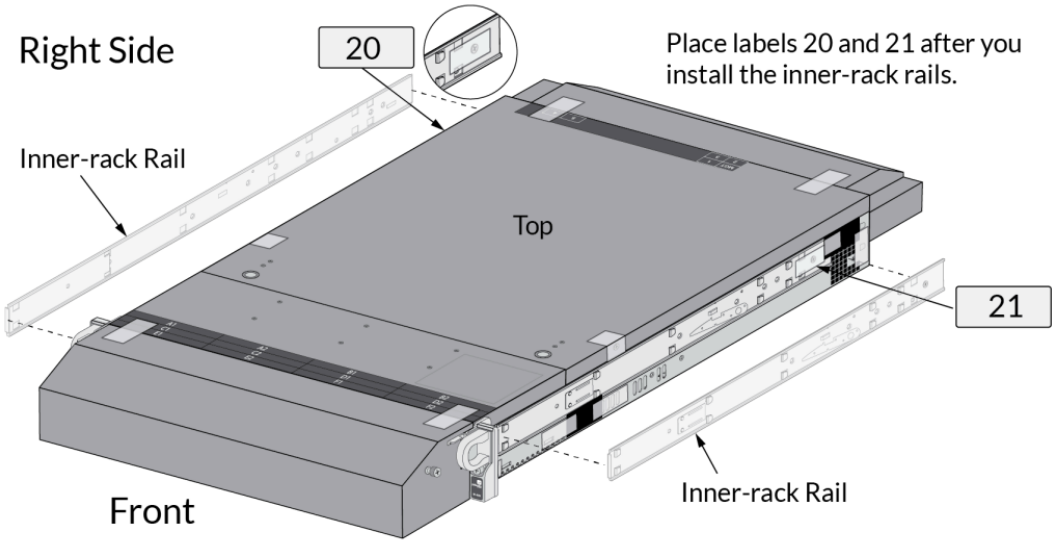

Figure 23 - WF-500-B: Tamper Seal Locations Top for Side Rails

8 Non-Invasive Security

N/A

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
HDD	Non-Volatile Memory	Static
RAM	Volatile Memory	Dynamic

Table 14: Storage Areas

From Web Cryptik

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Module Public Key Output	HDD	External (Outside of the Module's Boundary)	Plaintext	Automated	Electronic	
Password/Secret Input via SSHv2 encrypted by AES and HMAC	External (Outside of the Module's Boundary)	HDD	Encrypted	Automated	Electronic	KTS (SSHv2 with AES and HMAC)
Password/Secret Input via SSHv2 encrypted by AES-GCM	External (Outside of the Module's Boundary)	HDD	Encrypted	Automated	Electronic	KTS (SSHv2 with AES-GCM)
Peer Public Key Input	External (outside of module's boundary)	HDD	Plaintext	Automated	Electronic	

Table 15: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Power Cycle/Session Termination	Operator powers the module off or session terminates, and the module's memory is overwritten with either a random pattern or a constant pattern.	Powering off the module or terminating the session will erase all SSPs stored in the RAM of the module and make them non-retrievable.	Command via CLI by unplugging module
Zeroization Command	CO issues zeroization serviceSSPs are zeroized by overwriting the memory with a random pattern or a constant pattern.	The zeroization command will erase all SSPs stored in the RAM or in the Flash of the module and make them non-retrievable.	Entering into maintenance mode and selecting Factory Reset "debug system maintenance-mode"

Table 16: SSP Zeroization Methods

Once the module is rebooted and zeroization is initiated, the module cannot be accessed in any way and the zeroization process cannot be stopped, thus the SSPs would not be compromised during the time of zeroization. The Crypto Officer shall be in control of the module until the zeroization process is complete.

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
AES-GCM IV	Initialization vector used for AES-GCM	96 bits - 96 bits	Initialization Vector - PSP	CKG - Symmetric		KAS-ECC-KeyGen (IPSec/IKE) KAS-ECC-KeyGen (SSH) KAS-ECC-KeyGen (TLSv1.2) KAS-FFC-KeyGen (IPSec/IKE) KAS-FFC-KeyGen (SSH) KAS-FFC-KeyGen (TLSv1.2)
CA Certificates	ECDSA/RSA Public key - Used to trust a root CA intermediate CA and leaf /end entity certificates	2048 bits, 3072 bits, 4096 bits; 256 bits, 384 bits, 521 bits - 112 bits,128 bits, 150 bits; 128 bits,192 bits, 256 bits	Public Key - PSP	SSH ECDSA KeyGen SSH RSA KeyGen TLS ECDSA KeyGen TLS RSA KeyGen		SSH ECDSA SigGen SSH ECDSA SigVer SSH RSA SigGen SSH RSA SigVer TLS ECDSA SigGen TLS ECDSA SigVer TLS RSA SigGen TLS RSA SigVer
CO, User Password	Authentication string with a minimum length of eight (8) characters.	8 characters minimum - N/A	Authentication Data - CSP			
DRBG Key	AES 256 CTR DRBG state Key used in the generation of a random values	256 bits - 256 bits	DRBG Key - CSP	Counter DRBG (A3453)		Counter DRBG (A3453)
DRBG Seed	DRBG seed coming from the entropy source Seed length = 384 bits	384 bits - 256 bits	DRBG Seed - CSP	Counter DRBG (A3453)		Counter DRBG (A3453)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
DRBG V	AES 256 CTR DRBG state V used in the generation of a random values	128 bits - 128 bits	DRBG Internal State V value - CSP	Counter DRBG (A3453)		Counter DRBG (A3453)
ECDSA Private Keys	ECDSA Private key for generation of signatures and authentication (P-256, P-384, or P-521)	128 - 256 bits - 128 bits minimum	Private Key - CSP	IPSec/IKE ECDSA KeyGen SSH ECDSA KeyGen TLS ECDSA KeyGen		TLS ECDSA SigGen
ECDSA Public Keys	ECDSA public keys managed as certificates for the verification of signatures, establishment of TLS, operator authentication and peer authentication. (ECDSA P-256, P-384, or P-521)	128 - 256 bits - 128 bits minimum	Public Key - PSP	IPSec/IKE ECDSA KeyGen SSH ECDSA KeyGen TLS ECDSA KeyGen		TLS ECDSA SigVer
Entropy Input String	Entropy input string coming from the entropy source Input length = 384 bits	384 bits - 256 bits	DRBG - CSP	Entropy as per SP 800-90B		Counter DRBG (A3453)
Firmware integrity verification key	Used to check the integrity of all software code (HMAC-SHA-256 and ECDSA P-256) (Note: This is not considered an SSP)	128 bits - 128 bits	Public Key - Neither	External - Preloaded		
IKEv2 SKEYSEED	Used to derive encryption keys	160 bits, 256 bits, 384 bits, or 512 bits - 160 bits, 256 bits,	Key Agreement seed - CSP	IPSec/IKE Keying Materials Development		KAS-ECC (IPSec/IKE) KAS-FFC (IPSec/IKE)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		384 bits, or 512 bits				
IPSec/IKE Authentication Keys	HMAC keys for authentication (HMAC-SHA-256/384/512) (key size 256, 384, 512 bits)	256, 384, 512 bits - 256, 384, 512 bits	HMAC Keys - CSP	IPSec/IKE Keying Materials Development	KAS-ECC (IPSec/IKE) KAS-FFC (IPSec/IKE)	Session Authentication (IPSec/IKE)
IPSec/IKE DHE/ECDHE Private Components	Diffie-Hellman or EC Diffie-Hellman private component used in key establishment (DHE MODP-2048, ECDHE P-256, P-384)	2048 bits; 128 - 256 bits - 128 - 256 bits	Private Key - CSP	KAS-ECC-KeyGen (IPSec/IKE) KAS-FFC-KeyGen (IPSec/IKE)		IPSec/IKE Keying Materials Development
IPSec/IKE DHE/ECDHE Public Components	Diffie-Hellman or EC Diffie-Hellman public component used in key agreement (DHE MODP-2048, ECDHE P-256, P-384)	2048 bits; 128 - 256 bits - 128 - 256 bits	Public Key - PSP	KAS-ECC-KeyGen (IPSec/IKE) KAS-FFC-KeyGen (IPSec/IKE)		IPSec/IKE Keying Materials Development
IPSec/IKE Session Keys	Used to encrypt IKE/IPSec data. These are AES CBC or GCM (128 or 256 bits)	128 - 256 bits - 128, 192, 256 bits minimum	Session Key - CSP	KDF IKEv2 (A3453)	KAS-ECC (IPSec/IKE) KAS-FFC (IPSec/IKE)	Session Encryption/Decryption (IPSec/IKE)
IPSec/IKE Shared Secret	Diffie Hellman or EC Diffie-Hellman shared secret (DH MODP-2048, ECDH P-256, ECDH P-384, ECDH P-521)	2048 bits; 128 - 256 bits - 128 - 256 bits	Shared Secret - CSP	KDF IKEv2 (A3453) KDF SSH (A3453)		KAS-ECC (IPSec/IKE) KAS-FFC (IPSec/IKE)
Protocol Secrets	Secrets used by RADIUS or TACACS+ (8 characters minimum)	8 characters minimum - N/A	Authentication Data - CSP			
Public key for firmware content load test	Used to authenticate software/firmware and content	112 bits - 112 bits	Public Key - PSP			Firmware Load Test

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	to be installed on the firewall (RSA 2048 with SHA-256)					
RSA Private Keys	RSA Private keys for generation of signatures, authentication or key establishment. (RSA 2048, 3072, or 4096-bit)	2048 bits, 3072 bits, 4096 bits - 112 bits,128 bits, 150 bits	Private Key - CSP	IPSec/IKE RSA KeyGen SSH RSA KeyGen TLS RSA KeyGen		TLS RSA SigGen
RSA Public Keys	RSA public keys managed as certificates for the verification of signatures, establishment of TLS, operator authentication and peer authentication. (RSA 2048, 3072, or 4096-bit)	2048 - 4096 bits - 112 bits minimum	Public Key - PSP	IPSec/IKE RSA KeyGen SSH RSA KeyGen TLS RSA KeyGen		TLS RSA SigVer
SNMPv3 Authentication Key	HMAC-SHA-1/224/256/384/512 Authentication protocol key (160 bits)	160 - 512 bits - 160, 224, 256, 384, 512 bits minimum	Session Key - CSP	SNMPv3 Keying Materials Development		Session Authentication (SMPv3)
SNMPv3 Authentication Secret	Used to support SNMPv3 services (Minimum 8 characters)	8 characters minimum - N/A	Authentication Secret - CSP			SNMPv3 Keying Materials Development
SNMPv3 Privacy Secret	Used to support SNMPv3 services (Minimum 8 characters)	8 characters minimum - N/A	Authentication Key - CSP			SNMPv3 Keying Materials Development

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
SNMPv3 Session Key	Privacy protocol encryption key (AES 128/192/256 CFB)	128 - 256 bits - 128, 192, 256 bits minimum	Session Key - CSP	CKG - Symmetric KDF SNMP (A3453)		Session Encryption/Decryption (SNMPv3)
SSH Client Public Key	Public RSA key used to authenticate client. (RSA 2048, 3072, and 4096 bits)	2048 - 4096 bits - 112 bits minimum	Public Key - PSP			SSH RSA SigVer
SSH DHE/ECDHE Private Components	Diffie Hellman or EC Diffie-Hellman private (DH Group 14, ECDH P-256, ECDH P-384, ECDH P-521)	2048 bits; 128 - 256 bits - 112 bits minimum	Private Key - CSP	KAS-ECC-KeyGen (SSH) KAS-FFC-KeyGen (SSH)		KAS-ECC (SSH) KAS-FFC (SSH)
SSH DHE/ECDHE Public Components	Diffie Hellman or EC Diffie-Hellman public component (DH Group 14, ECDH P-256, ECDH P-384, ECDH P-521)	2048 bits; 128 - 256 bits - 112 bits minimum	Public Key - PSP	KAS-ECC-KeyGen (SSH) KAS-FFC-KeyGen (SSH)	KAS-ECC (SSH) KAS-FFC (SSH)	KAS-ECC (SSH) KAS-FFC (SSH)
SSH Host Public Key	SSH Host Public Key (RSA 2048, RSA 3072, RSA 4096, ECDSA P-256, P-384, or P-521)	2048 - 4096 bits; 128 - 256 bits - 112 and 256 bits (RSA) 128, 192, and 256 bits (ECDSA) minimum	Public Key - PSP	SSH ECDSA KeyGen SSH RSA KeyGen		SSH RSA SigVer SSH ECDSA SigVer
SSH Session Authentication Keys	Authentication keys used in all SSH connections to the security module's command line interface	160, 256, or 512 bits - 160, 224, 256, 384, 512 bits	Session Key - CSP	KDF SSH (A3453)	KAS-ECC (SSH) KAS-FFC (SSH)	Session Authentication (SSHv2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	(HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-512) (160, 256, 512 bits)	160 bits minimum				
SSH Session Encryption Keys	Used in all SSH connections to the security module's command line interface. (128, 192, or 256 bits: AES CBC or CTR) (128 or 256 bits: AES GCM)	128 - 256 bits - 128 and 256 bits minimum	Session Key - CSP	KDF SSH (A3453)		Session Encryption/Decryption (SSH)
SSH Shared Secret	Diffie Hellman or EC Diffie-Hellman shared secret (DH MODP-2048, ECDH P-256, ECDH P-384, ECDH P-521)	128 - 512 bits - 128 - 256 bits	Shared Secret - CSP	KDF SSH (A3453)		KAS-ECC (SSH) KAS-FFC (SSH)
TLS DHE/ECDHE Private Components	Ephemeral Diffie-Hellman private FFC or EC component used in TLS (DHE 2048, ECDHE P-256, P-384, P-521)	2048 bits - 4096 bits 128 - 256 bits - 112 bits minimum	Private Key - CSP	KAS-ECC-KeyGen (TLSv1.2) KAS-FFC-KeyGen (TLSv1.2)		KAS-ECC (TLSv1.2) KAS-FFC (TLSv1.2)
TLS DHE/ECDHE Public Components	Diffie_Hellman or EC Diffie-Hellman Ephemeral values used in key agreement (DHE 2048, ECDHE P-256, P-384, P-521)	2048 bits - 4096 bits 128 - 256 bits - 112 bits minimum	Public Key - PSP	KAS-ECC-KeyGen (TLSv1.2) KAS-FFC-KeyGen (TLSv1.2)		KAS-ECC (TLSv1.2) KAS-FFC (TLSv1.2)
TLS Encryption Keys	AES (128 or 256 bit) keys used in TLS	128 - 256 bits - 128	Session Key - CSP		KAS-ECC (TLSv1.2)	Session Encryption/Decryption (TLSv1.2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	connections (GCM; CBC)	bits minimum			KAS-FFC (TLSv1.2)	
TLS HMAC Keys	HMAC keys used in TLS connections (HMAC-SHA2-256/384) (256, 384 bits)	256 - 384 bits - 256 bits minimum	Session Key - CSP		KAS-ECC (TLSv1.2) KAS-FFC (TLSv1.2)	Session Authentication (TLSv1.2)
TLS Master Secret	Secret value used to derive the TLS session keys	48 bytes - N/A	Master Secret - CSP		KAS-ECC (TLSv1.2) KAS-FFC (TLSv1.2)	KAS-ECC (TLSv1.2) KAS-FFC (TLSv1.2)
TLS Pre-Master Secret	Secret value used to derive the TLS Master Secret along with client and server random nonces	48 bytes - N/A	Shared Secret - CSP		KAS-ECC (TLSv1.2) KAS-FFC (TLSv1.2)	KAS-ECC (TLSv1.2) KAS-FFC (TLSv1.2)

Table 17: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES-GCM IV		RAM:Plaintext	Until Session Termination	Power Cycle/Session Termination	
CA Certificates	Password/Secret Input via SSHv2 encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by AES-GCM Peer Public Key Input	HDD:Plaintext RAM:Plaintext	Until zeroization command is issued or session termination	Zeroization Command Power Cycle/Session Termination	RSA Private Keys:Decrypts RSA Public Keys:Encrypts ECDSA Private Keys:Decrypts ECDSA Public Keys:Encrypts
CO, User Password	Password/Secret Input via SSHv2 encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by AES-GCM	HDD:Obfuscated	N/A	Zeroization Command	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
DRBG Key		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	Entropy Input String:Paired With DRBG V:Paired With
DRBG Seed		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	Entropy Input String:Paired With DRBG Key:Paired With DRBG V:Paired With
DRBG V		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	Entropy Input String:Paired With DRBG Key:Paired With
ECDSA Private Keys	Password/Secret Input via SSHv2 encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by AES-GCM	HDD:Plaintext RAM:Plaintext	Until zeroization command is issued or session termination	Zeroization Command Power Cycle/Session Termination	ECDSA Public Key:Paired With
ECDSA Public Keys	Module Public Key Output Password/Secret Input via SSHv2 encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by AES-GCM Peer Public Key Input	HDD:Plaintext RAM:Plaintext	Until zeroization command is issued	Zeroization Command	ECDSA Private Keys:Paired With
Entropy Input String		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	DRBG Seed:Paired With DRBG Key:Paired With DRBG V:Paired With
Firmware integrity verification key		HDD:Plaintext			
IKEv2 SKEYSEED		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	IPSec/IKE DHE/ECDHE Private Components:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					IPSec/IKE DHE/ECDHE Public Components:Paired With
IPSec/IKE Authentication Keys		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	
IPSec/IKE DHE/ECDHE Private Components		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	IPSec/IKE DHE/ECDHE Public Components:Paired With
IPSec/IKE DHE/ECDHE Public Components		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	IPSec/IKE DHE/ECDHE Private Components:Paired With
IPSec/IKE Session Keys		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	
IPSec/IKE Shared Secret		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	
Protocol Secrets	Password/Secret Input via SSHv2 encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by AES-GCM	HDD:Plaintext RAM:Plaintext	Until zeroization command is issued	Zeroization Command	
Public key for firmware content load test		HDD:Plaintext	N/A		
RSA Private Keys	Password/Secret Input via SSHv2 encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by AES-GCM	HDD:Plaintext RAM:Plaintext	Until zeroization command is issued or session termination	Zeroization Command Power Cycle/Session Termination	RSA Public Keys:Paired With
RSA Public Keys	Password/Secret Input via SSHv2 encrypted by	HDD:Plaintext RAM:Plaintext	Until zeroization command is issued	Zeroization Command	RSA Private Keys:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	AES and HMAC Password/Secret Input via SSHv2 encrypted by AES-GCM Peer Public Key Input				
SNMPv3 Authentication Key		HDD:Plaintext RAM:Plaintext	Until zeroization command is issued	Zeroization Command	SNMPv3 Authentication Secret:Derived From SNMPv3 Privacy Secret:Derived From
SNMPv3 Authentication Secret	Password/Secret Input via SSHv2 encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by AES-GCM	HDD:Plaintext RAM:Plaintext	Until zeroization command is issued	Zeroization Command	
SNMPv3 Privacy Secret	Password/Secret Input via SSHv2 encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by AES-GCM	HDD:Plaintext RAM:Plaintext	Until zeroization command is issued	Zeroization Command	
SNMPv3 Session Key		HDD:Plaintext RAM:Plaintext	Until zeroization command is issued	Zeroization Command	Derived From SNMPv3 Authentication Secret:Derived From SNMPv3 Privacy Secret:Derived From
SSH Client Public Key	Password/Secret Input via SSHv2 encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by AES-GCM	HDD:Plaintext RAM:Plaintext	Until zeroization command is issued	Zeroization Command	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Peer Public Key Input				
SSH DHE/ECDHE Private Components		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	SSH DHE/ECDHE Public Components:Paired With
SSH DHE/ECDHE Public Components	Module Public Key Output Peer Public Key Input	RAM:Plaintext	Until session termination	Power Cycle/Session Termination	SSH DHE/ECDHE Private Components:Paired With
SSH Host Public Key		HDD:Plaintext RAM:Plaintext	Until zeroization command is issued	Zeroization Command	
SSH Session Authentication Keys		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	SSH DHE/ECDHE Public Components:Derived From SSH DHE/ECDHE Private Components:Derived From
SSH Session Encryption Keys		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	Derived From SSH DHE/ECDHE Public Components:Derived From SSH DHE/ECDHE Private Components:Derived From
SSH Shared Secret		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	
TLS DHE/ECDHE Private Components		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	TLS DHE/ECDHE Public Components:Paired With
TLS DHE/ECDHE Public Components	Module Public Key Output Peer Public Key Input	RAM:Plaintext	Until session termination	Power Cycle/Session Termination	TLS DHE/ECDHE Private Components:Paired With
TLS Encryption Keys		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	TLS Master Secret:Derived From
TLS HMAC Keys		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	TLS Master Secret:Derived From
TLS Master Secret		RAM:Plaintext	Until session termination	Power Cycle/Sessio	TLS Pre-Master Secret:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				n Termination	
TLS Pre-Master Secret		RAM:Plaintext	Until session termination	Power Cycle/Session Termination	

Table 18: SSP Table 2

9.5 Transitions

Key Sizes

- Key sizes with a security strength less than 128-bits will be non-Approved for all uses starting January 1, 2031.

SHA-1

- The module implements SHA-1 for use in non-digital signature applications. This implementation will be non-Approved for all uses starting January 1, 2031.

10 Self-Tests

The cryptographic module performs the following tests below. The operator can command the module to perform the pre-operational and cryptographic algorithm self-tests by cycling power of the module. The pre-operational and conditional self-tests are performed automatically and do not require any additional operator action.

10.1 Pre-Operational Self-Tests

Note: the ECDSA and HMAC-SHA-256 KATs are performed prior to the Software integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
ECDSA SigVer (FIPS186-4) (A3453)	P-256	KAT	SW/FW Integrity	Self-Test successful	Signature Verification
HMAC-SHA2-256 (A3453)	SHA2-256	KAT	SW/FW Integrity	Self-Test successful	Keyed Checksum

Table 19: Pre-Operational Self-Tests

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES CCM Decrypt	192 Bits	KAT	CAST	Self-test output message	Decrypt	After each power-on or via self-test command

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES CCM Encrypt	192 Bits	KAT	CAST	Self-test output message	Encrypt	After each power-on or via self-test command
AES GCM Decrypt	256 Bits	KAT	CAST	Self-test output message	Decrypt	After each power-on or via self-test command
AES GCM Encrypt	256 Bits	KAT	CAST	Self-test output message	Encrypt	After each power-on or via self-test command
AES-ECB Decrypt	128 bits	KAT	CAST	Self-test output message	Decrypt	After each power-on or via self-test command
DRBG	N/A	KAT	CAST	Self-test output message	SP 800-90Arev1 Instantiate/Generate/Reseed Known Answer Tests	After each power-on or via self-test command
ECDSA / KAS-ECC	256 Bit Minimum	PCT	PCT	System log messages	ECDSA / KAS-ECC pairwise consistency test	On session
ECDSA Sign	256 Bits	KAT	CAST	Self-test output message	Sign	After each power-on or via self-test command
ECDSA Verify	256 Bits	KAT	CAST	Self-test output message	Verify	After each power-on or via self-test command
Firmware Load Test	2048 Bit	FW Load Test	SW/FW Load	System log messages	Firmware load test on content load	On session
HMAC-SHA-1	160 Bits	KAT	CAST	Self-test output message	Keyed Hash	After each power-on or via self-test command
HMAC-SHA2-224	224 Bits	KAT	CAST	Self-test output message	Keyed Hash	After each power-on or via self-

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						test command
HMAC-SHA2-256	256 Bits	KAT	CAST	Self-test output message	Keyed Hash	After each power-on or via self-test command
HMAC-SHA2-384	384 Bits	KAT	CAST	Self-test output message	Keyed Hash	After each power-on or via self-test command
HMAC-SHA2-512	512 Bits	KAT	CAST	Self-test output message	Keyed Hash	After each power-on or via self-test command
KAS-FFC	2048 Bit Minimum	PCT	PCT	System log messages	KAS-FCC pairwise consistency test	On session
RSA Keygen	2048 Bit Minimum	PCT	PCT	System log messages	RSA pairwise consistency test	On session
RSA Sign	2048 Bits	KAT	CAST	Self-test output message	Sign	After each power-on or via self-test command
RSA Verify	2048 Bits	KAT	CAST	Self-test output message	Verify	After each power-on or via self-test command
SHA-1	160 Bits	KAT	CAST	Self-test output message	Hash	After each power-on or via self-test command
SHA2-256	256 Bits	KAT	CAST	Self-test output message	Hash	After each power-on or via self-test command
SHA2-384	384 Bits	KAT	CAST	Self-test output message	Hash	After each power-on or via self-test command

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA2-512	512 Bits	KAT	CAST	Self-test output message	Hash	After each power-on or via self-test command
SP 800-135rev1 IKEv2 KDF with SHA-256	N/A	KAT	CAST	Self-test output message	IKEv2 with SHA-256	After each power-on or via self-test command
SP 800-135rev1 SSH KDF with SHA-256	N/A	KAT	CAST	Self-test output message	SSHv2 with SHA-256	After each power-on or via self-test command
SP 800-135rev1 TLS 1.2 with SHA-256 KDF	N/A	KAT	CAST	Self-test output message	TLSv1.2 with SHA-256	After each power-on or via self-test command
SP 800-56A Rev 3 Assurance Tests	N/A	Critical Functions	Critical Function	System log messages	Assurance tests for SP 800-56A Rev3	On session
SP 800-56Ar3 KAS-ECC-SSC	256 Bits	KAT	CAST	Self-test output message	KAS Computation	After each power-on or via self-test command
SP 800-56Ar3 KAS-FFC-SSC	2048 Bits	KAT	CAST	Self-test output message	KAS Computation	After each power-on or via self-test command
SP 800-90B APT Health Tests on Entropy Source	N/A	Fault detection	CAST	Self-test output message	Health tests done on entropy source	After each power-on or via self-test command
SP 800-90B RCT Health Tests on Entropy Source	N/A	Fault detection	CAST	Self-test output message	Health tests done on entropy source	After each power-on or via self-test command

Table 20: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
ECDSA SigVer (FIPS186-4) (A3453)	KAT	SW/FW Integrity	On Demand	Manually or Scheduled
HMAC-SHA2-256 (A3453)	KAT	SW/FW Integrity	On Demand	Manually or Scheduled

Table 21: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES CCM Decrypt	KAT	CAST	On Demand	Manually or Scheduled
AES CCM Encrypt	KAT	CAST	On Demand	Manually or Scheduled
AES GCM Decrypt	KAT	CAST	On Demand	Manually or Scheduled
AES GCM Encrypt	KAT	CAST	On Demand	Manually or Scheduled
AES-ECB Decrypt	KAT	CAST	On Demand	Manually or Scheduled
DRBG	KAT	CAST	On Demand	Manually or Scheduled
ECDSA / KAS-ECC	PCT	PCT	On session	On session
ECDSA Sign	KAT	CAST	On Demand	Manually or Scheduled
ECDSA Verify	KAT	CAST	On Demand	Manually or Scheduled
Firmware Load Test	FW Load Test	SW/FW Load	On session	On session
HMAC-SHA-1	KAT	CAST	On Demand	Manually or Scheduled
HMAC-SHA2-224	KAT	CAST	On Demand	Manually or Scheduled
HMAC-SHA2-256	KAT	CAST	On Demand	Manually or Scheduled
HMAC-SHA2-384	KAT	CAST	On Demand	Manually or Scheduled
HMAC-SHA2-512	KAT	CAST	On Demand	Manually or Scheduled
KAS-FFC	PCT	PCT	On session	On session
RSA Keygen	PCT	PCT	On session	On session
RSA Sign	KAT	CAST	On Demand	Manually or Scheduled
RSA Verify	KAT	CAST	On Demand	Manually or Scheduled
SHA-1	KAT	CAST	On Demand	Manually or Scheduled
SHA2-256	KAT	CAST	On Demand	Manually or Scheduled

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
SHA2-384	KAT	CAST	On Demand	Manually or Scheduled
SHA2-512	KAT	CAST	On Demand	Manually or Scheduled
SP 800-135rev1 IKEv2 KDF with SHA-256	KAT	CAST	On Demand	Manually or Scheduled
SP 800-135rev1 SSH KDF with SHA-256	KAT	CAST	On Demand	Manually or Scheduled
SP 800-135rev1 TLS 1.2 with SHA-256 KDF	KAT	CAST	On Demand	Manually or Scheduled
SP 800-56A Rev 3 Assurance Tests	Critical Functions	Critical Function	On session	On session
SP 800-56Ar3 KAS-ECC-SSC	KAT	CAST	On Demand	Manually or Scheduled
SP 800-56Ar3 KAS-FFC-SSC	KAT	CAST	On Demand	Manually or Scheduled
SP 800-90B APT Health Tests on Entropy Source	Fault detection	CAST	On Demand	Manually or Scheduled
SP 800-90B RCT Health Tests on Entropy Source	Fault detection	CAST	On Demand	Manually or Scheduled

Table 22: Conditional Periodic Information

From Web Cryptik

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Conditional Firmware Load Test Failure	Signature verification fails on firmware load	Signature verification failure	N/A	System prints Invalid image message.
Conditional Pairwise Consistency or Critical Functions Test Failure	Module fails a PCT or critical functions test	PCT / Critical functions test	Reset session	System log prints an error message.
Self-Test / Integrity Test Failure	Module fails a self-test or integrity test	Self-test or Integrity Test failure	Reboot Module or Factory Reset	FIPS-CC mode failure. <Algorithm test> failed.

Table 23: Error States

From Web Cryptik

10.5 Operator Initiation of Self-Tests

Perform a power cycle or via the ‘Self-Tests’ service by entering CLI command “request restart system”.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The following procedure will put the modules into the Approved mode of operation:

- Install physical kit opacity shields and tamper evidence seals according to the Physical Security Policy section. Physical kits must be correctly installed to operate in the Approved mode of operation. The tamper evidence seals and opacity shields shall be installed for the module to operate in the Approved mode of operation.
- During initial boot up, break the boot sequence via the console port connection (by pressing the main button when instructed to do so) to access the main menu.
- Select “Continue.”
- Select the “Set FIPS-CC Mode” option to enter the Approved mode.
- Select “Enable FIPS-CC Mode”.
- When prompted, select “Reboot” and the module will re-initialize and continue into Approved mode.
- The module will reboot.
- In Approved mode, the console port is available as a status output port.
- Once the module has finished booting, the Crypto Officer can authenticate using the default credentials that come with the module
    - Once authenticated, the module will automatically require the operator to change their password; and the default credential is overwritten

The module will automatically indicate the Approved mode of operation in the following manner:

- Status output interface will indicate “**** FIPS-CC MODE ENABLED ****” via the CLI session.
- Status output interface will indicate “FIPS-CC mode enabled successfully” via the console port.

Should one or more power-up self-tests fail, the Approved mode of operation will not be achieved. Feedback will consist of:

- The module will output “FIPS-CC failure”
- The module will reboot and enter a state in which the reason for the reboot can be determined.
- To determine which self-test caused the system to reboot into the error state, connect the console cable and follow the on-screen instructions to view the self-test output.

Note: Disabling Approved mode causes a complete factory reset, which is described in the Zeroization section below.

Failure to follow the installation/instructions steps in Section 11.1 would result in the module operating in a non-compliant state

11.2 Administrator Guidance

https://docs.paloaltonetworks.com/content/dam/techdocs/en_US/pdf/advanced-wildfire/wildfire-appliance.pdf

11.3 Non-Administrator Guidance

N/A

11.4 Design and Rules

In Approved mode, the following rules shall apply:

The operator should not enable or use TLSv1.3

- Checked via CLI using “show profiles” command

If using RADIUS, it must be configured using TLS.

- Checked via CLI using “show shared” command

11.5 End of Life

The following procedure will zeroize the module:

- Access the module’s CLI via SSH, and command the module to enter maintenance mode; the module will reboot
    - Note: Establish a serial connection to the console port
- After reboot, select “Continue.”
- Select “Factory Reset”
- The module will perform a zeroization, and provide the following message once complete:
    - “Factory Reset Status: Success”

Note: Following the completion of this procedure, the module will be placed back into an uninitialized state.