Cisco Systems, Inc.

Cisco Adaptive Security Appliance Cryptographic Module (FPR 3100 Series)

FIPS 140-3 Non-Proprietary Security Policy

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This is Cisco Systems, Inc. non-proprietary security policy for the Cisco Adaptive Security Appliance Cryptographic Module (FPR 3100 Series) (hereinafter referred to as ASA or Module), version 9.20. The following details how this module meets the security requirements of FIPS 140-3, SP 800-140 and ISO/IEC 19790 for a Security Level 2 Hardware cryptographic module.

The security requirements cover areas related to the design and implementation of a cryptographic module. These areas include cryptographic module specification; cryptographic module interfaces; roles, services, and authentication; software/firmware security; operational environment; physical security; non-invasive security; sensitive security parameter management; self-tests; life-cycle assurance; and mitigation of other attacks. The following table indicates the actual security levels for each area of the cryptographic module.

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	2
12	Mitigation of other attacks	N/A
	Overall Level	2

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

This module is a multi-chip standalone hardware cryptographic module deployed under the Next-Generation Firewall (NGFW) with Adaptive Security Appliance (ASA). The module's operational environment is non-modifiable**.**

ASA delivers enterprise-class firewall for businesses, improving security at the Internet edge, high performance and throughput for demanding enterprise data centers. The ASA solution offers the combination of the industry's most deployed stateful firewall with a comprehensive range of next-generation network security services, intrusion prevention system (IPS), content security and secure unified communications, HTTPS/TLSv1.2, SSHv2, IPsec/IKEv2, SNMPv3 and Cryptographic Cipher Suite B using the ASA Cryptographic Module.

Module Type: Hardware

Module Embodiment: MultiChipStand

Module Characteristics [O]:

Cryptographic Boundary:

The cryptographic boundary is defined as the entire chassis unit's physical perimeter encompassing the "top," "front," "left," "right," "rear" and "bottom" surfaces of the case, and shown in the figures below and in the Physical Security section. The FPR 3105, FPR 3110, FPR 3120, FPR 3130 and FPR 3140 all have the same exterior appearance. Where they differ is in Firewall throughput, IPS throughput, IPsec VPN throughput and number of VPN peers allowed.

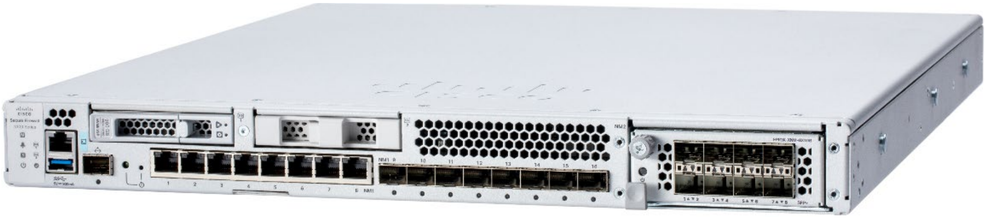


Figure 1 FPR 3105, 3110, 3120, 3130, 3140

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
FRP 3105	FPR-3105	9.20	AMD EPYC 7272 (Zen2) & NITROX-V, Marvell Semiconductor, NITROX	
FRP 3110	FPR-3110	9.20	AMD EPYC 7272 (Zen2) & NITROX-V, Marvell Semiconductor, NITROX	
FRP 3120	FPR-3120	9.20	AMD EPYC 7282 (Zen2) & NITROX-V, Marvell Semiconductor, NITROX	
FRP 3130	FPR-3130	9.20	AMD EPYC 7352 (Zen2) & NITROX-V, Marvell Semiconductor, NITROX	
FRP 3140	FPR-3140	9.20	AMD EPYC 7452 (Zen2) & NITROX-V, Marvell Semiconductor, NITROX	

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 for this module.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved Mode of Operation	The module is always in the approved mode of operation after initial operations are performed.	Approved	Approved mode indicator: "FIPS is currently enabled."

Table 3: Modes List and Description

The module has one approved mode of operation and is always in the approved mode of operation after initial operations are performed (See Section 11). The module does not claim implementation of a degraded mode of operation. Section 4 provides details on the service indicator implemented by the module.

2.5 Algorithms

Approved Algorithms:

CiscoSSL FOM Cryptographic Implementation

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A4446	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4446	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 192, 256	SP 800-38D

Algorithm	CAVP Cert	Properties	Reference
Counter DRBG	A4446	Prediction Resistance - Yes Mode - AES-128, AES-192, AES-256 Derivation Function Enabled - Yes	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-4)	A4446	Curve - P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A4446	Curve - P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A4446	Curve - P-256, P-384, P-521	FIPS 186-4
HMAC-SHA-1	A4446	Key Length - Key Length: 256-448 Increment 8	FIPS 198-1
HMAC-SHA2-224	A4446	Key Length - Key Length: 256-448 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4446	Key Length - Key Length: 256-448 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4446	Key Length - Key Length: 256-448 Increment 8	FIPS 198-1
HMAC-SHA2-512	A4446	Key Length - Key Length: 256-448 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4446	Domain Parameter Generation Methods - P-256, P-384, P-521	SP 800-56A Rev. 3
KAS-FFC-SSC Sp800-56Ar3	A4446	Domain Parameter Generation Methods - ffdhe2048, ffdhe3072, ffdhe4096, modp-2048, modp-3072, modp-4096	SP 800-56A Rev. 3
KDF IKEv2 (CVL)	A4446	Diffie-Hellman Shared Secret Length - Diffie-Hellman Shared Secret Length: 2048 Derived Keying Material Length - Derived Keying Material Length: 3072 Hash Algorithm - SHA-1	SP 800-135 Rev. 1
KDF SNMP (CVL)	A4446	Password Length - Password Length: 256, 64	SP 800-135 Rev. 1
KDF SSH (CVL)	A4446	Cipher - AES-128, AES-192, AES-256	SP 800-135 Rev. 1
RSA KeyGen (FIPS186-4)	A4446	Key Generation Mode - B.3.4 Modulo - 2048, 3072, 4096 Hash Algorithm - SHA2-256 Private Key Format - Standard	FIPS 186-4
RSA SigGen (FIPS186-4)	A4446	Signature Type - ANSI X9.31, PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
RSA SigVer (FIPS186-4)	A4446	Signature Type - ANSI X9.31, PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096	FIPS 186-4
Safe Primes Key Generation	A4446	Safe Prime Groups - modp-2048, modp-3072, modp-4096	SP 800-56A Rev. 3
SHA-1	A4446	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Algorithm	CAVP Cert	Properties	Reference
SHA2-224	A4446	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A4446	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A4446	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A4446	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
TLS v1.2 KDF RFC7627 (CVL)	A4446	Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1

Table 4: Approved Algorithms - CiscoSSL FOM Cryptographic Implementation

Marvell Cavium Nitrox V

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	C1026	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	C1026	Direction - Decrypt, Encrypt IV Generation - External Key Length - 128, 192, 256	SP 800-38D
Hash DRBG	C1026	Prediction Resistance - No Mode - SHA2-512	SP 800-90A Rev. 1
HMAC-SHA-1	C1026	-	FIPS 198-1
HMAC-SHA2-256	C1026	-	FIPS 198-1
HMAC-SHA2-384	C1026	-	FIPS 198-1
HMAC-SHA2-512	C1026	-	FIPS 198-1
SHA-1	C1026	Message Length - Message Length: 0-51200 Increment 8	FIPS 180-4
SHA2-256	C1026	Message Length - Message Length: 0-51200 Increment 8	FIPS 180-4
SHA2-384	C1026	Message Length - Message Length: 0-102400 Increment 8	FIPS 180-4
SHA2-512	C1026	Message Length - Message Length: 0-102400 Increment 8	FIPS 180-4

Table 5: Approved Algorithms - Marvell Cavium Nitrox V

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Key Type:Asymmetric	CiscoSSL FOM Cryptographic Implementation	The cryptographic module performs Cryptographic Key Generation (CKG) for asymmetric keys as per sections 4 and 5 in SP800-133rev2 (vendor affirmed) and FIPS

Name	Properties	Implementation	Reference
			140-3 IG D.H. A seed (i.e., the random value) used in asymmetric key generation is a direct output from SP800-90Arev1 CTR_DRBG (A4446) or HMAC_DRBG (C1026)

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:

N/A for this module.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
KAS-ECC-KeyGen (SSHv2)	KAS-KeyGen	KAS ECC keygen used in SSHv2 service		Counter DRBG Hash DRBG CKG
KAS-FFC-KeyGen (SSHv2)	KAS-KeyGen	KAS FFC keygen used in SSHv2 service		Counter DRBG Safe Primes Key Generation Hash DRBG CKG
KAS-ECC-KeyGen (TLSv1.2)	KAS-KeyGen	KAS ECC keygen used in TLSv1.2 service		Counter DRBG Hash DRBG CKG
KAS-FFC-KeyGen (TLSv1.2)	KAS-KeyGen	KAS FFC keygen used in TLSv1.2 service		Counter DRBG Safe Primes Key Generation Hash DRBG CKG
KAS-ECC-KeyGen (IKEv2)	KAS-KeyGen	KAS ECC keygen used in IKE v2 service		Counter DRBG Hash DRBG CKG
KAS-FFC-KeyGen (IKEv2)	KAS-KeyGen	KAS FFC keygen used in IKE v2 service		Counter DRBG Safe Primes Key Generation

Name	Type	Description	Properties	Algorithms
				Hash DRBG CKG
KAS-FFC (SSHv2)	KAS-Full	Key Agreement Scheme per SP800-56Arev3 with KDF SSH. The module's KAS (FFC) implementation is FIPS140-3 IG D.F Scenario 2 (path 2) compliant	Bit-strength Caveat:Provides between 112 to 152 bits of encryption strength	KDF SSH KAS-FFC-SSC Sp800-56Ar3 Domain Parameter Generation Methods:: modp-2048
KAS-ECC (SSHv2)	KAS-Full	Key Agreement Scheme per SP800-56Arev3 with KDF SSH. The module's KAS (FFC) implementation is FIPS140-3 IG D.F Scenario 2 (path 2) compliant	Bit-strength Caveat:Provides between 128 and 256 bits of encryption strength	KDF SSH KAS-ECC-SSC Sp800-56Ar3
KAS-FFC (TLSv1.2)	KAS-Full	Key Agreement Scheme per SP800-56Arev3 with TLS v1.2 KDF RFC7627. The module's KAS (FFC) implementation is FIPS140-3 IG D.F Scenario 2 (path 2) compliant	Bit-strength Caveat:Provides between 112 to 152 bits of encryption strength	TLS v1.2 KDF RFC7627 KAS-FFC-SSC Sp800-56Ar3 Domain Parameter Generation Methods:: modp-2048
KAS-ECC (TLSv1.2)	KAS-Full	Key Agreement Scheme per SP800-56Arev3 with KDF IKEv2. The module's KAS (ECC) implementation is FIPS140-3 IG D.F Scenario 2 (path 2) compliant	Bit-strength Caveat:Provides between 128 and 256 bits of encryption strength	TLS v1.2 KDF RFC7627 KAS-ECC-SSC Sp800-56Ar3

Name	Type	Description	Properties	Algorithms
KAS-ECC (IKEv2)	KAS-Full	Key Agreement Scheme per SP800-56Arev3 with KDF IKEv2. The module's KAS (ECC) implementation is FIPS140-3 IG D.F Scenario 2 (path 2) compliant	Bit-strength Caveat:Provides between 112 and 256 bits of encryption strength	KAS-ECC-SSC Sp800-56Ar3 KDF IKEv2
KAS-FFC (IKEv2)	KAS-Full	Key Agreement Scheme per SP800-56Arev3 with KDF IKEv2. The module's KAS (FFC) implementation is FIPS140-3 IG D.F Scenario 2 (path 2) compliant	Bit-strength Caveat:Provides between 112 and 152 bits of encryption strength	KAS-FFC-SSC Sp800-56Ar3 KDF IKEv2
KTS (TLSv1.2 with AES and HMAC)	KTS-Wrap	KTS via TLSv1.2 service by using AES and HMAC	Bit-strength Caveat:Provides between 128 and 256 bits of encryption strength	AES-CBC Key Length: 128, 256 HMAC-SHA-1 HMAC-SHA2-256 HMAC-SHA2-384 SHA-1 SHA2-256 SHA2-384
KTS (TLSv1.2 with AES-GCM)	KTS-Wrap	KTS via TLSv1.2 service by using AES-GCM	Bit-strength Caveat:Provides between 128 and 256 bits of encryption strength	AES-GCM Key Length: 128, 256 AES-CBC
KTS (SSHv2 with AES and HMAC)	KTS-Wrap	KTS via SSHv2 service by using AES and HMAC	Bit-strength Caveat:Provides between 128 and 256 bits of encryption strength	AES-CBC Key Length: 128, 256 HMAC-SHA-1 HMAC-SHA2-256 SHA-1 SHA2-256

Name	Type	Description	Properties	Algorithms
KTS (SSHv2 with AES-GCM)	KTS-Wrap	KTS via SSHv2 service by using AES-GCM	Bit-strength Caveat:Provides between 128 and 256 bits of encryption strength	AES-GCM Key Length: 128, 256 AES-CBC
RSA KeyGen (SSHv2, TLSv1.2, IKEv2)	AsymKeyPair-KeyGen	RSA KeyGen for SSHv2, TLSv1.2, and IKEv2 services		RSA KeyGen (FIPS186-4) Counter DRBG Hash DRBG
ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2)	AsymKeyPair-KeyGen	ECDSA KeyGen for TLSv1.2 and IKEv2 services		ECDSA KeyGen (FIPS186-4) Counter DRBG Hash DRBG
RSA SigGen (SSHv2, TLSv1.2, IKEv2)	DigSig-SigGen	RSA SigGen for SSHv2, TLSv1.2, and IKEv2 services		RSA SigGen (FIPS186-4)
ECDSA SigGen (SSHv2, TLSv1.2 and IKEv2)	DigSig-SigGen	ECDSA SigGen for TLSv1.2, and IKEv2 services		ECDSA SigGen (FIPS186-4)
RSA SigVer (SSHv2, TLSv1.2, and IKEv2)	DigSig-SigVer	RSA SigVer for SSHv2, TLSv1.2, and IKEv2 services		RSA SigVer (FIPS186-4)
ECDSA SigVer (SSHv2, TLSv1.2, and IKEv2)	DigSig-SigVer	ECDSA SigVer for TLSv1.2 and IKEv2 services		ECDSA SigVer (FIPS186-4)
Block Cipher (SSHv2)	BC-Auth BC-UnAuth	Block Cipher for SSHv2 service		AES-CBC Key Length: 128, 256 AES-GCM Key Length: 128, 256
Block Cipher (TLSv1.2)	BC-Auth BC-UnAuth	Block Cipher for TLSv1.2 service		AES-GCM Key Length: 128, 256 AES-CBC Key Length: 128, 256
Block Cipher (IPSec/IKEv2)	BC-Auth BC-UnAuth	Block Cipher for IPSec/IKEv2 service		AES-CBC AES-GCM AES-CBC AES-GCM
Block Cipher (SNMPv3)	BC-UnAuth	Block Cipher for SNMPv3 service		AES-CBC KDF SNMP

Name	Type	Description	Properties	Algorithms
MAC (SSHv2)	MAC	MAC for SSHv2 service		HMAC-SHA-1 HMAC-SHA2-256 SHA-1 SHA2-256
MAC (TLSv1.2)	MAC	Message Authentication for TLSv1.2 services		HMAC-SHA-1 HMAC-SHA2-256 HMAC-SHA2-384 SHA-1 SHA2-256 SHA2-384
MAC (IPSec/IKEv2)	MAC	Message Authentication for IPSec/IKEv2 services		HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-512 SHA2-256 SHA2-384 SHA2-512 HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-512 SHA2-256 SHA2-384 SHA2-512 HMAC-SHA-1 SHA-1
MAC (SNMPv3)	MAC	Message Authentication for SNMPv3 service		HMAC-SHA-1 SHA-1 KDF SNMP HMAC-SHA2-256 HMAC-SHA2-384 SHA2-256 SHA2-384 HMAC-SHA2-224 SHA2-224
Firmware Load Test	MAC	MAC for firmware load test		HMAC-SHA2-512

Name	Type	Description	Properties	Algorithms
SSHv2 Keying Materials Development	KAS-135KDF	SSHv2 session keying materials, used to derive SSHv2 session keys		KDF SSH
TLS Keying Materials Development	KAS-135KDF	TLS session keying materials, used to derive TLS session keys		TLS v1.2 KDF RFC7627
IKEv2 Keying Materials Development	KAS-135KDF	IKEv2 session keying materials, used to derive IKEv2 session keys		KDF IKEv2
SNMPv3 Keying Materials Development	KAS-135KDF	SNMPv3 session keying materials, used to derive SNMPv3 session keys		KDF SNMP
DRBG Function	DRBG	DRBG generation		Counter DRBG Hash DRBG

Table 7: Security Function Implementations

2.7 Algorithm Specific Information

There are some algorithm modes that were tested but not implemented by the module. Only the algorithms, modes, and key sizes that are implemented by the module are shown in this table.

The module's AES-GCM implementation conforms to Implementation Guidance C.H scenario #1 following RFC 5288 for TLS. The module is compatible with TLSv1.2 and provides support for the acceptable GCM cipher suites from SP 800-52 Rev1, Section 3.3.1. The operations of one of the two parties involved in the TLS key establishment scheme were performed entirely within the cryptographic boundary of the module being validated. The counter portion of the IV is set by the module within its cryptographic boundary. When the IV exhausts the maximum number of possible values for a given session key, the first party, client or server, to encounter this condition will trigger a handshake to establish a new encryption key. The keys for the client and server negotiated in the TLSv1.2 handshake process (client_write_key and server_write_key) are compared and the module aborts the session if the key values are identical. In case the module's power is lost and then restored, a new key for use with the AES GCM encryption/decryption shall be established.

The module uses RFC 7296 compliant IKEv2 to establish the shared secret SKEYSEED from which the AES GCM encryption keys are derived. When the IV exhausts the maximum number of possible values for a given session key, the first party, client or server, to encounter this condition will trigger a handshake to establish a new encryption key. Two keys established by IKEv2 for one security association (one key for encryption in each direction between the parties) are not identical and abort the session if they are. In case the module's power is lost and then restored, a new key for use with the AES GCM encryption/decryption shall be established.

No parts of SSH, TLS, IKE and SNMP protocols, other than the KDFs, have been tested by the CAVP and CMVP.

2.8 RBG and Entropy

Cert Number	Vendor Name
E3	Cisco Systems, Inc.

Table 8: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Cisco Jitter Entropy Source	Non-Physical	AMD EPYC 7272 (Zen2), AMD EPYC 7282 (Zen2), AMD EPYC 7352 (Zen2), AMD EPYC 7452 (Zen2)	4 bits	2 bits	A2810 (SHA3-256)

Table 9: Entropy Sources

2.9 Key Generation

The module generates RSA, ECDSA, ECDH, and DH asymmetric key pairs compliant with FIPS 186-4, using a NIST SP 800-90A CTR DRBG for random number generation. In accordance with FIPS 140-3 IG D.H, the cryptographic module performs CKG for asymmetric keys as per section 5.1 of NIST SP 800-133rev2 (vendor affirmed) by obtaining a random bit string directly from an approved DRBG. The random bit string supports the required security strength requested by the calling application (without any V, as described in Additional Comments 2 of IG D.H.).

2.10 Key Establishment

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

- KAS-FFC Shared Secret Computation: The module provides SP800-56Arev3 compliant key establishment according to FIPS 140-3 IG D.F scenario 2 path (2) with KAS-FFC shared secret computation. The shared secret computation provides between 112 and 152 bits of encryption strength.

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

2.11 Industry Protocols

The module supports SSHv2, TLS v1.2, SNMPv3 and IPsec/IKEv2 industrial protocols. Please refer to SSPs Table for more information.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
Ethernet Port, SFP (1G) port, SFP+ (10G) port, and Console Port	Data Input	Data input into the module for all the services defined in Approved Services Table, including TLSv1.2, SSHv2, SNMPv3 and IPsec/IKEv2 service data.
Ethernet Port, SFP (1G) port, SFP+ (10G) port and Console Port	Data Output	Data output from the module for all the services defined in Approved Services Table, including TLSv1.2, SSHv2, SNMPv3 and IPsec/IKEv2 service data.
Ethernet Port, SFP (1G) port, SFP+ (10G) port, Console Port and RESET	Control Input	Control Data input into the module for all the services defined in Approved Services Table, including TLSv1.2, SSHv2, SNMPv3 and IPsec/IKEv2 service data.
Ethernet Port, SFP (1G) port, SFP+ (10G) port, Console Port and LEDs	Status Output	Status Information output from the module.
N/A	Control Output	N/A
Power	Power	Provide the Power Supply to the module.

Table 10: Ports and Interfaces

The module's physical perimeter encompasses the case of the tested platform mentioned in Table 2. The module provides physical ports which are mapped to logical interfaces provided by the module (data input, data output, control input, control output and status output) as above.

4 Roles, Services, and Authentication

4.1 Authentication Methods

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
Password	The minimum length is eight (8) characters (94	Password Based	The probability that a random	The probability of successfully

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
	possible characters). The configuration supports at most ten failed attempts to authenticate in a one-minute period.		attempt will succeed or a false acceptance will occur is 1/(94^8) which is less than 1/1,000,000.	authenticating to the module within one minute is 10/(94^8), which is less than 1/100,000.
RSA-Based Certificate	The modules support RSA public-key based authentication mechanism using a minimum of RSA 2048 bits, which provides 112 bits of security strength. The probability that a random attempt will succeed is 1/(2^112) which is less than 1/1,000,000. For multiple attacks during a one-minute period, as the module at its highest can support at most 17,000 new sessions per second to authenticate in a one-minute period, the probability of successfully authenticating to the module within a one minute period is 17,000 * 60 = 1,020,000/(2^112), which is less than 1/100,000.	RSA SigVer (FIPS186-4) (A4446)	The probability that a random attempt will succeed is 1/(2^112). Please refer to Description section in this table for more details	the probability of successfully authenticating to the module within a one minute period is 17,000 * 60 = 1,020,000/(2^112). Please refer to Description section in this table for more details
ECDSA-Based Certificate	The modules support ECDSA public-key based authentication mechanism using a minimum of curve P-256, which provides 128 bits of security strength. The probability that a random attempt will succeed is 1/(2^128) which is less than 1/1,000,000. For	ECDSA SigVer (FIPS186-4) (A4446)	The probability that a random attempt will succeed is 1/(2^128) which is less than 1/1,000,000. Please refer to Description section in this	the probability of successfully authenticating to the module within a one minute period is 17,000 * 60 = 1,020,000/(2^128). Please refer to Description section in this table for more details

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
	multiple attacks during a one-minute period, as the module at its highest can support at most 17,000 new sessions per second to authenticate in a one-minute period, the probability of successfully authenticating to the module within a one minute period is 17,000 * 60 = 1,020,000/(2^128), which is less than 1/100,000.		table for more details	

Table 11: Authentication Methods

The module implements identity-based authentication. The module supports Crypto Officer role and the User role. The module also allows the concurrent operators.

4.2 Roles

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

Table 12: Roles

Unauthenticated Users can run the self-test service by power-cycling the module by removing the power and re-applying.

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Show Status	Provide Module’s current status (return	N/A	Command used to show Module's Status	Module's Operational Status	None	Crypto Officer User

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	codes and/or syslog messages)					
Show Version	Provide Module's name and version information	N/A	Command to show version	Module's ID and versioning information	None	Crypto Officer User
Perform Self-Tests	Perform Self-Tests (Pre-operational self-test and Conditional Self-Tests)	N/A	Command to trigger Self-Test	Status of the self-tests results	None	Crypto Officer User Unauthenticated
Perform Zeroization	Perform Zeroization	Syslog message	Command to zeroize the module	Status of the SSPs zeroization	None	Crypto Officer - DRBG Entropy Input: Z - DRBG Seed: Z - DRBG Internal State V value: Z - DRBG Key: Z - User Password: Z - Crypto Officer Password: Z - RADIUS Secret: Z - TACACS+ Secret: Z - Firmware Load Test Key: Z - SSH DH Private Key: Z - SSH DH Public Key: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- SSH Peer DH Public Key: Z - SSH DH Shared Secret: Z - SSH ECDH Private Key: Z - SSH ECDH Public Key: Z - SSH Peer ECDH Public Key: Z - SSH ECDH Shared Secret: Z - SSH RSA Private Key: Z - SSH RSA Public Key: Z - SSH ECDSA Private Key: Z - SSH ECDSA Public Key: Z - SSH Session Encryption Key: Z - SSH Session Authentication Key: Z - TLS DH Private Key: Z - TLS DH Public Key: Z - TLS Peer DH Public

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Key: Z - TLS DH Shared Secret: Z - TLS ECDH Private Key: Z - TLS ECDH Public Key: Z - TLS Peer ECDH Public Key: Z - TLS ECDH Shared Secret: Z - TLS ECDSA Private Key: Z - TLS ECDSA Public Key: Z - TLS RSA Private Key: Z - TLS RSA Public Key: Z - TLS Master Secret: Z - TLS Session Encryption Key: Z - TLS Session Authentication Key: Z - IPSec/IKE DH Private Key: Z - IPSec/IKE DH Public Key: Z - IPSec/IKE Peer DH

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Public Key: Z - IPSec/IKE DH Shared Secret: Z - IPSec/IKE ECDH Private Key: Z - IPSec/IKE ECDH Public Key: Z - IPSec/IKE Peer ECDH Public Key: Z - IPSec/IKE ECDH Shared Secret: Z - IPSec/IKE ECDSA Private Key: Z - IPSec/IKE ECDSA Public Key: Z - IPSec/IKE RSA Private Key: Z - IPSec/IKE RSA Public Key: Z - IPSec/IKE Pre-shared Secret: Z - SKEYSEED: Z - IPSec/IKE Session Encryption Key: Z - IPSec/IKE Authentication Key: Z - SNMPv3

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Shared Secret: Z - SNMPv3 Encryption Key: Z - SNMPv3 Authentication Key: Z
Configure Network	Sets configuration of the systems	None	Commands to configure the network	Status of the completion of network configuration status	None	Crypto Officer
Crypto Officer Authentication	CO Role Authentication	N/A	CO Authentication Request	Status of the CO authentication	None	Crypto Officer - Crypto Officer Password: W,Z
User Authentication	User Role Authentication	N/A	User role authentication request	Status of the User role authentication	None	User - User Password: W,Z
Configure Bypass Capability	Sets the Bypass capability	None	CLI Bypass commands	Status of the completion of Bypass capability configuration	None	Crypto Officer
Configure SSHv2 Function	Configure SSHv2 Function	Global Indicator and SSHv2 configuration success status message	Commands to configure SSHv2	Status of the completion of the SSHv2 configuration	KAS-FFC (SSHv2) KAS-ECC (SSHv2) KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) RSA KeyGen (SSHv2,	Crypto Officer - SSH DH Private Key: W,E - SSH DH Public Key: W,E - SSH Peer DH Public Key: W,E - SSH DH Shared Secret: W,E - SSH ECDH Private Key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					TLSv1.2, IKEv2) ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2) RSA SigGen (SSHv2, TLSv1.2, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2 and IKEv2) RSA SigVer (SSHv2, TLSv1.2, and IKEv2) ECDSA SigVer (SSHv2, TLSv1.2, and IKEv2) Block Cipher (SSHv2) MAC (SSHv2) KAS-ECC-KeyGen (SSHv2) KAS-FFC-KeyGen (SSHv2) DRBG Function SSHv2 Keying Materials Development	W,E - SSH ECDH Public Key: W,E - SSH Peer ECDH Public Key: W,E - SSH ECDH Shared Secret: W,E - SSH RSA Private Key: W,E - SSH RSA Public Key: W,E - SSH ECDSA Private Key: W,E - SSH ECDSA Public Key: W,E - SSH Session Encryption Key: W,E - SSH Session Authentication Key: W,E - DRBG Entropy Input: W,E - DRBG Seed: W,E - DRBG Internal State V value: W,E - DRBG Key: W,E
Configure HTTPS over	Configure HTTPS over	Global Indicator and HTTPS	Commands to configure TLSv1.2	Status of the completion of TLSv1.2	KAS-FFC (TLSv1.2) KAS-ECC (TLSv1.2)	Crypto Officer - TLS DH Private Key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TLSv1.2 Function	TLSv1.2 Function	over TLSv1.2 configuration success status message		configuration	KTS (TLSv1.2 with AES and HMAC) KTS (TLSv1.2 with AES-GCM) RSA KeyGen (SSHv2, TLSv1.2, IKEv2) ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2) RSA SigGen (SSHv2, TLSv1.2, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2 and IKEv2) RSA SigVer (SSHv2, TLSv1.2, and IKEv2) ECDSA SigVer (SSHv2, TLSv1.2, and IKEv2) Block Cipher (TLSv1.2) MAC (TLSv1.2) KAS-ECC-KeyGen (TLSv1.2) KAS-FFC-KeyGen	W,E - TLS DH Public Key: W,E - TLS Peer DH Public Key: W,E - TLS DH Shared Secret: W,E - TLS ECDH Private Key: W,E - TLS ECDH Public Key: W,E - TLS Peer ECDH Public Key: W,E - TLS ECDH Shared Secret: W,E - TLS ECDSA Private Key: W,E - TLS ECDSA Public Key: W,E - TLS RSA Private Key: W,E - TLS RSA Public Key: W,E - TLS Master Secret: W,E - TLS Session Encryption Key: W,E - TLS Session Authentication Key: W,E - DRBG Entropy

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					(TLSv1.2) TLS Keying Materials Development DRBG Function	Input: W,E - DRBG Seed: W,E - DRBG Internal State V value: W,E - DRBG Key: W,E
Configure IPsec/IKEv2 Function	Configure IPSec/IKEv2 Function	Global Indicator with IPsec/IKEv2 configuration success status message	Commands to configure IPsec/IKEv2	Status of the completion of IPsec/IKEv2 configuration	KAS-ECC (IKEv2) KAS-FFC (IKEv2) RSA KeyGen (SSHv2, TLSv1.2, IKEv2) ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2) RSA SigGen (SSHv2, TLSv1.2, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2 and IKEv2) RSA SigVer (SSHv2, TLSv1.2, and IKEv2) ECDSA SigVer (SSHv2, TLSv1.2, and IKEv2) Block Cipher (IPSec/IKEv2) MAC (IPSec/IKE	Crypto Officer - IPSec/IKE DH Private Key: W,E - IPSec/IKE DH Public Key: W,E - IPSec/IKE Peer DH Public Key: W,E - IPSec/IKE DH Shared Secret: W,E - IPSec/IKE ECDH Private Key: W,E - IPSec/IKE ECDH Public Key: W,E - IPSec/IKE Peer ECDH Public Key: W,E - IPSec/IKE ECDH Shared Secret: W,E - IPSec/IKE ECDSA Private Key: W,E - IPSec/IKE ECDSA Public Key: W,E - IPSec/IKE

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					v2) KAS-ECC-KeyGen (IKEv2) KAS-FFC-KeyGen (IKEv2) IKEv2 Keying Materials Development DRBG Function	RSA Private Key: W,E - IPSec/IKE RSA Public Key: W,E - IPSec/IKE Pre-shared Secret: W,E - SKEYSEED: W,E - IPSec/IKE Session Encryption Key: W,E - IPSec/IKE Authentication Key: W,E - DRBG Entropy Input: W,E - DRBG Seed: W,E - DRBG Internal State V value: W,E - DRBG Key: W,E
Run SSHv2 Function	Execute SSHv2 Function	Global Indicator and successful SSHv2 log message	Initiate SSHv2 tunnel establishment	Status of SSHv2 tunnel establishment	KAS-FFC (SSHv2) KAS-ECC (SSHv2) KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) RSA KeyGen (SSHv2, TLSv1.2, IKEv2) ECDSA KeyGen	Crypto Officer - SSH DH Private Key: W,E - SSH DH Public Key: W,E - SSH Peer DH Public Key: W,E - SSH DH Shared Secret: W,E - SSH ECDH Private Key: W,E - SSH ECDH Public Key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					(SSHv2, TLSv1.2 and IKEv2) RSA SigGen (SSHv2, TLSv1.2, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2 and IKEv2) RSA SigVer (SSHv2, TLSv1.2, and IKEv2) ECDSA SigVer (SSHv2, TLSv1.2, and IKEv2) Block Cipher (SSHv2) MAC (SSHv2) KAS-ECC-KeyGen (SSHv2) KAS-FFC-KeyGen (SSHv2) DRBG Function SSHv2 Keying Materials Development	- SSH Peer ECDH Public Key: W,E - SSH ECDH Shared Secret: W,E - SSH RSA Private Key: W,E - SSH RSA Public Key: W,E - SSH ECDSA Private Key: W,E - SSH ECDSA Public Key: W,E - SSH Session Encryption Key: W,E - SSH Session Authentication Key: W,E - DRBG Entropy Input: W,E - DRBG Seed: W,E - DRBG Internal State V value: W,E - DRBG Key: W,E User - SSH DH Private Key: W,E - SSH DH Public Key: W,E - SSH Peer DH Public

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Key: W,E - SSH DH Shared Secret: W,E - SSH ECDH Private Key: W,E - SSH ECDH Public Key: W,E - SSH Peer ECDH Public Key: W,E - SSH ECDH Shared Secret: W,E - SSH RSA Private Key: W,E - SSH RSA Public Key: W,E - SSH ECDSA Private Key: W,E - SSH ECDSA Public Key: W,E - SSH Session Encryption Key: W,E - SSH Session Authentication Key: W,E - DRBG Entropy Input: W,E - DRBG Seed: W,E - DRBG Internal State V value: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- DRBG Key: W,E
Run HTTPS over TLSv1.2 Function	Execute HTTPS over TLSv1.2 function	Global Indicator and successful HTTPS over TLSv1.2 log message	Initiate TLSv1.2 tunnel establishment request	Status of TLSv1.2 tunnel establishment	KAS-FFC (TLSv1.2) KAS-ECC (TLSv1.2) KTS (TLSv1.2 with AES and HMAC) KTS (TLSv1.2 with AES-GCM) RSA KeyGen (SSHv2, TLSv1.2, IKEv2) ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2) RSA SigGen (SSHv2, TLSv1.2, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2 and IKEv2) RSA SigVer (SSHv2, TLSv1.2, and IKEv2) ECDSA SigVer (SSHv2, TLSv1.2, and IKEv2) Block Cipher (TLSv1.2) MAC	Crypto Officer - TLS DH Private Key: W,E - TLS DH Public Key: W,E - TLS Peer DH Public Key: W,E - TLS DH Shared Secret: W,E - TLS ECDH Private Key: W,E - TLS ECDH Public Key: W,E - TLS Peer ECDH Public Key: W,E - TLS ECDH Shared Secret: W,E - TLS ECDSA Private Key: W,E - TLS ECDSA Public Key: W,E - TLS RSA Private Key: W,E - TLS RSA Public Key: W,E - TLS Master Secret: W,E - TLS Session Encryption Key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					(TLSv1.2) KAS-ECC-KeyGen (SSHv2) KAS-FFC-KeyGen (SSHv2) DRBG Function SSHv2 Keying Materials Development	- TLS Session Authentication Key: W,E - DRBG Entropy Input: W,E - DRBG Seed: W,E - DRBG Internal State V value: W,E - DRBG Key: W,E User - TLS DH Private Key: W,E - TLS DH Public Key: W,E - TLS Peer DH Public Key: W,E - TLS DH Shared Secret: W,E - TLS ECDH Private Key: W,E - TLS ECDH Public Key: W,E - TLS Peer ECDH Public Key: W,E - TLS ECDH Shared Secret: W,E - TLS ECDSA Private Key: W,E - TLS ECDSA Public Key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS RSA Private Key: W,E - TLS RSA Public Key: W,E - TLS Master Secret: W,E - TLS Session Encryption Key: W,E - TLS Session Authentication Key: W,E - DRBG Entropy Input: W,E - DRBG Seed: W,E - DRBG Internal State V value: W,E - DRBG Key: W,E
Run IPSec/IKEv2 Function	Execute IPsec/IKEv2 Function	Global Indicator and succesful IPsec/IKEv2 log message	Initiate IPsec/IKEv2 tunnel establishment request	Status of IPSec/IKEv2 tunnel establishment	KAS-ECC (IKEv2) KAS-FFC (IKEv2) RSA KeyGen (SSHv2, TLSv1.2, IKEv2) ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2) RSA SigGen (SSHv2, TLSv1.2, IKEv2) ECDSA SigGen (SSHv2,	Crypto Officer - IPSec/IKE DH Private Key: W,E - IPSec/IKE DH Public Key: W,E - IPSec/IKE Peer DH Public Key: W,E - IPSec/IKE DH Shared Secret: W,E - IPSec/IKE ECDH Private Key: W,E - IPSec/IKE ECDH Public Key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					TLSv1.2 and IKEv2) RSA SigVer (SSHv2, TLSv1.2, and IKEv2) ECDSA SigVer (SSHv2, TLSv1.2, and IKEv2) Block Cipher (IPSec/IKEv2) MAC (IPSec/IKEv2) KAS-ECC-KeyGen (IKEv2) KAS-FFC-KeyGen (IKEv2) IKEv2 Keying Materials Development DRBG Function	W,E - IPSec/IKE Peer ECDH Public Key: W,E - IPSec/IKE ECDH Shared Secret: W,E - IPSec/IKE ECDSA Private Key: W,E - IPSec/IKE ECDSA Public Key: W,E - IPSec/IKE RSA Private Key: W,E - IPSec/IKE RSA Public Key: W,E - IPSec/IKE Pre-shared Secret: W,E - SKEYSEED: W,E - IPSec/IKE Session Encryption Key: W,E - IPSec/IKE Authentication Key: W,E - DRBG Entropy Input: W,E - DRBG Seed: W,E - DRBG Internal State V value: W,E - DRBG Key: W,E User - IPSec/IKE

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						DH Private Key: W,E - IPSec/IKE DH Public Key: W,E - IPSec/IKE Peer DH Public Key: W,E - IPSec/IKE DH Shared Secret: W,E - IPSec/IKE ECDH Private Key: W,E - IPSec/IKE ECDH Public Key: W,E - IPSec/IKE Peer ECDH Public Key: W,E - IPSec/IKE ECDH Shared Secret: W,E - IPSec/IKE ECDSA Private Key: W,E - IPSec/IKE ECDSA Public Key: W,E - IPSec/IKE RSA Private Key: W,E - IPSec/IKE RSA Public Key: W,E - IPSec/IKE Pre-shared Secret: W,E - SKEYSEED: W,E - IPSec/IKE

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Session Encryption Key: W,E - IPSec/IKE Authentication Key: W,E - DRBG Entropy Input: W,E - DRBG Seed: W,E - DRBG Internal State V value: W,E - DRBG Key: W,E
Configure SNMPv3 Function	Configure SNMPv3 Function	Global Indicator and SNMPv3 configuration success status message	Commands to configure SNMPv3	Status of the completion of SNMPv3 configuration	Block Cipher (SNMPv3) MAC (SNMPv3) SNMPv3 Keying Materials Development	Crypto Officer - SNMPv3 Shared Secret: W,E - SNMPv3 Encryption Key: W,E - SNMPv3 Authentication Key: W,E
Run SNMPv3 Function	Execute SNMPv3 Function	Global Indicator and successful SNMPv3 log message	Initiate SNMPv3 tunnel establishment request	Status of SNMPv3 tunnel establishment	Block Cipher (SNMPv3) MAC (SNMPv3) SNMPv3 Keying Materials Development	Crypto Officer User
Firmware Load Test	Execute the Firmware Load Test	Global indicator and successful Firmware Loading status message	Commands to load new firmware image	Outcome of the Firmware Load Test	Firmware Load Test	Crypto Officer - Firmware Load Test Key: R

Table 13: Approved Services

4.4 Non-Approved Services

N/A for this module.

4.5 External Software/Firmware Loaded

The module also supports the firmware load test by using HMAC-SHA2-512 (HMAC Cert. #A4446) 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 to the module before loading the 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. Any firmware loaded into the module that is not shown on the module certificate, is out of scope of this validation and requires a separate FIPS 140-3 validation.

4.6 Bypass Actions and Status

The module implements Bypass service. The operator shall assume Crypto Officer role and configure the Bypass capability. The module will conduct two independent internal actions activate the capability to prevent the inadvertent bypass of plaintext data due to a single error. To verify the module is in bypass state, the Crypto Officer needs to issue commands "show access-list" and "show ipsec sa" to verify the module is in bypass state.

4.7 Cryptographic Output Actions and Status

The module implements Self-initiated cryptographic output capability without external operator request. The Crypto Officer shall configure self-initiated cryptographic output capability. Prior to executing the self-initiated cryptographic output capability, the module conducts two independent internal actions to activate the capability to prevent the inadvertent output due to a single error.

4.8 Additional Information

The module supports unauthenticated service. The unauthenticated User/Operators can trigger the self-test service by power-cycling the module.

5 Software/Firmware Security

5.1 Integrity Techniques

The module is provided in the form of binary executable code. To ensure firmware security, the module is protected by RSA 2048 bits with SHA2-512 (RSA Cert. #A4446) algorithm. A Firmware Integrity Test Key (non-SSP) was preloaded to the module's binary at the factory and used for firmware integrity test only at the pre-operational self-test. The module uses the RSA

2048 bits modulus public key to verify the digital signature. If the firmware integrity test fails, the module would enter to an Error state with all crypto functionality inhibited.

5.2 Initiate on Demand

Integrity test is performed as part of the Pre-Operational Self-Tests. It is automatically executed at power-on. The operator can power-cycle or reboot the tested platform to initiate the firmware integrity test on-demand.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Limited

7 Physical Security

7.1 Mechanisms and Actions Required

Mechanism	Inspection Frequency	Inspection Guidance
Tamper labels (9) with Part number: AIR-AP-FIPSKIT=	Recommend 30 Days	Visible inspection of platform for residual evidence of tampering
Opacity shield (1) with Part number: FPR3K-FIPS-KIT=	Recommend 30 Days	Visible inspection of platform for evidence of tampering, removal or access

Table 14: Mechanisms and Actions Required

The module utilizes a production-grade enclosure and removable cover along with tamper evidence labels as the physical security mechanisms.

Appling Tamper Evidence Labels

Step 1: Turn off and unplug the module.

Step 2: Clean the chassis of any grease, dirt, oil or any other material other than the surface coating from manufacture before applying the tamper evident labels. Alcohol-based cleaning pads are recommended for this purpose.

Step 3: Apply a label to cover the module as shown in the figures below.

The tamper evident labels are produced from a special thin gauge vinyl with self-adhesive backing. Any attempt to open the module will damage the tamper evident labels or the material of the security appliance cover. Because the tamper evident labels have non-repeated serial numbers, they may be inspected for damage and compared against the applied serial numbers to verify that the security appliance has not been tampered with. Tamper evident labels can also be inspected for signs of tampering, which include the following: curled corners, rips, and slices. The word “FIPS” may appear if the label was peeled back.

7.2 User Placed Tamper Seals

Number: Nine (9)

Placement:

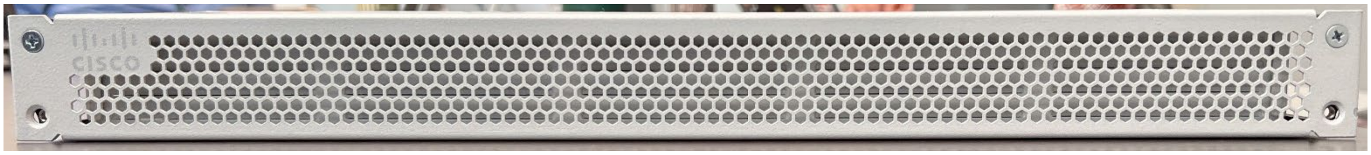


Figure 2 Module's front view opacity shield

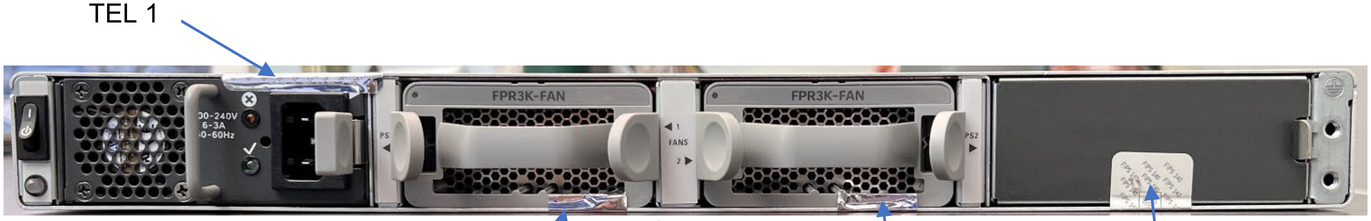


Figure 3 Module's back view

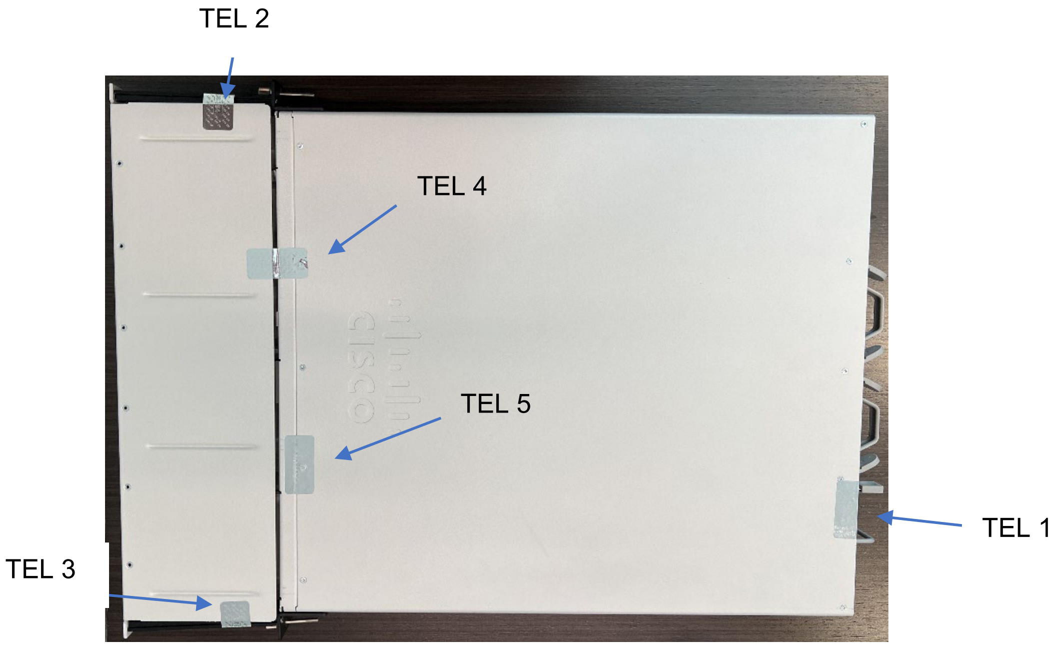


Figure 4 Module's top view with opacity shield

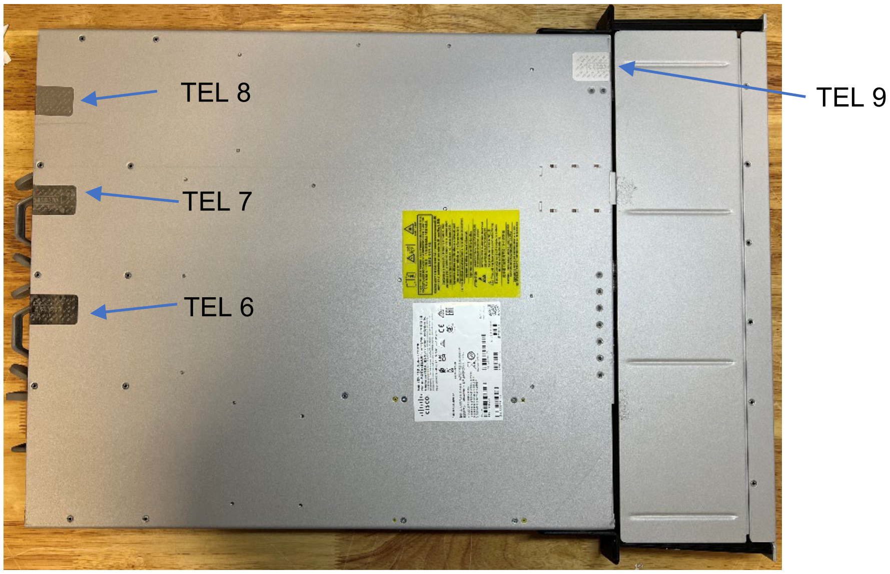


Figure 5 Module's bottom view with opacity shield

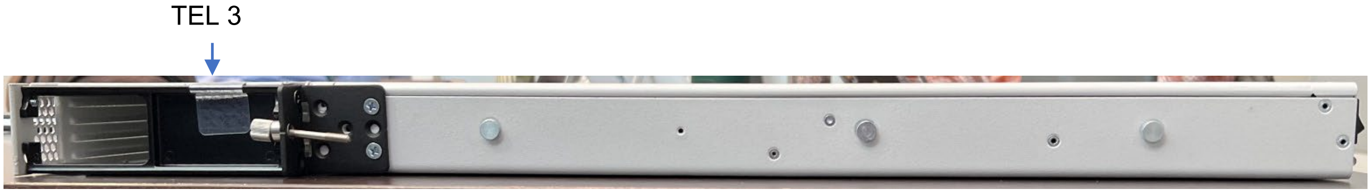


Figure 6 Module's left view with opacity shield

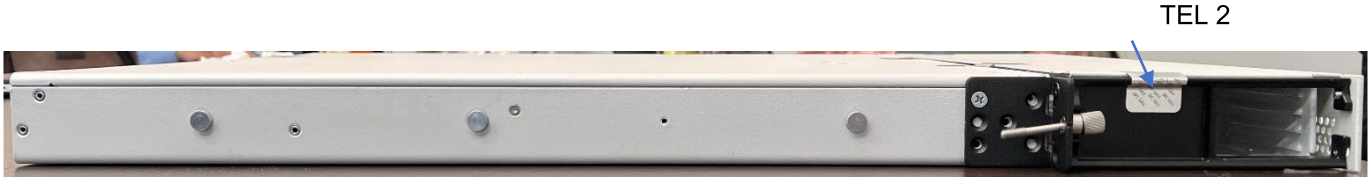


Figure 7 Module's right view with opacity shield

Surface Preparation: Clean the chassis of any grease, dirt, or oil before applying the tamper evident labels. Alcohol-based cleaning pads are recommended for this purpose.

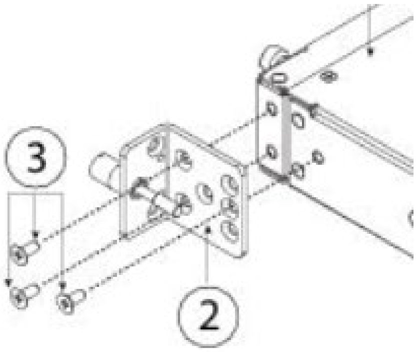
Operator Responsible for Securing Unused Seals: Must be stored in a secure location under controlled access

7.3 Filler Panels

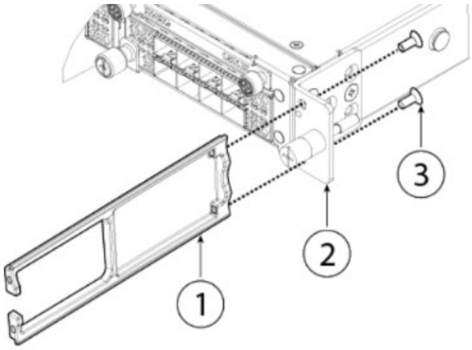
3105, 3110, 3120, 3130, 3140 Opacity Shield

FPR3K-FIPS-KIT=

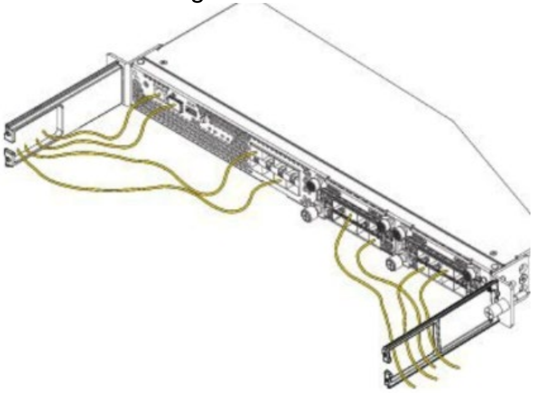
Step 1: Attach the Slide Rail Locking Bracket, #2 in diagram to the Side of the Chassis using the countersink screws #3 in diagram.



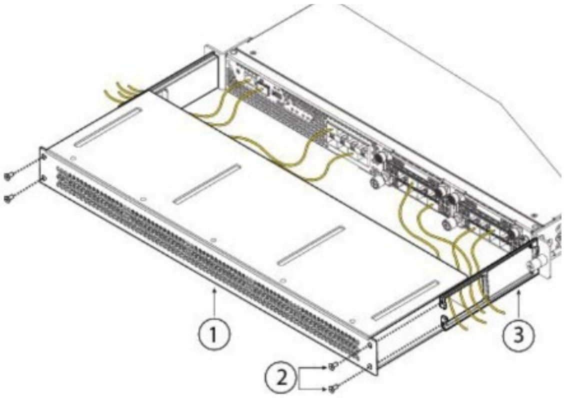
Step 2: Attach the Cable Management Bracket (#1) to the Slide Rail Locking Bracket (#2) using the countersink screws (#3)



Step 3: Route the Cables through the Cable Management Brackets



Step 4: Attach the FIPS Opacity Shield (#1) to the Cable Management Brackets (#3) using the countersink screws (#2)



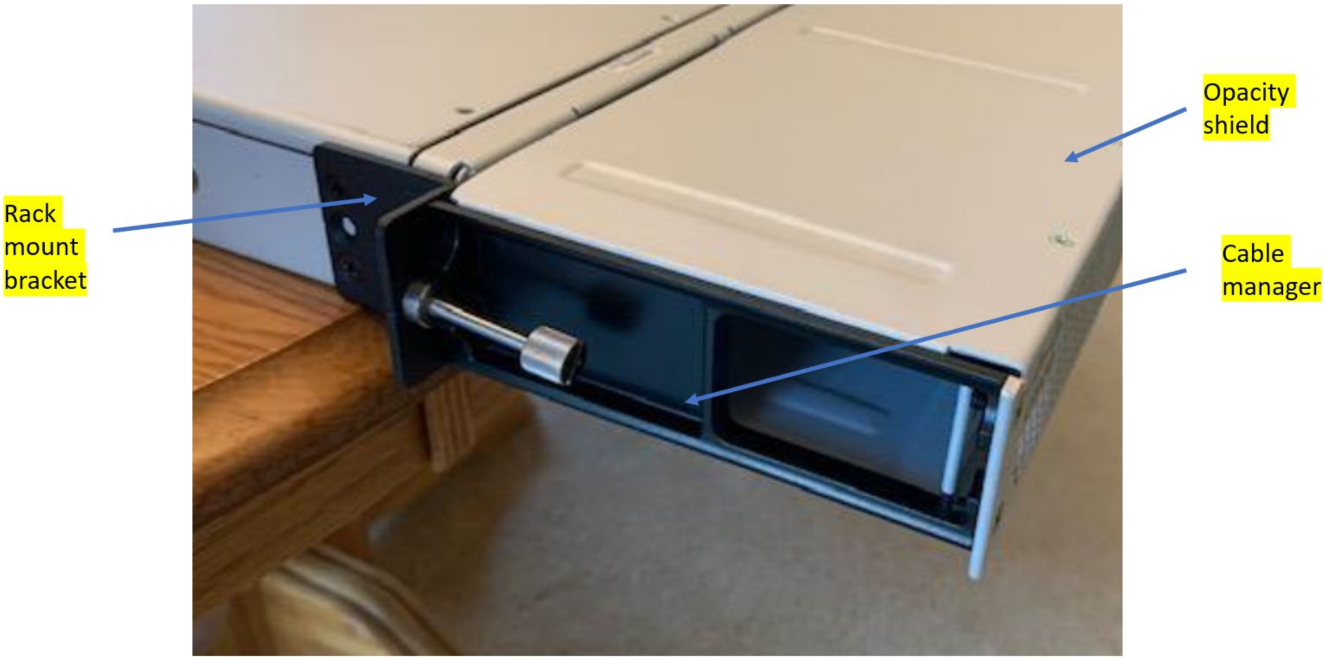


Figure 8 Opacity Shield Brackets

8 Non-Invasive Security

N/A for this module.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
DRAM	Volatile Memory	Dynamic
Flash	Non-Volatile Memory	Static

Table 15: Storage Areas

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Password/Secret Input via TLS encrypted by GCM	External (Outside of the Module's Boundary)	Module	Encrypted	Automated	Electronic	KTS (TLSv1.2 with AES-GCM)
Password/Secret Input via TLS encrypted by AES and HMAC	External (Outside of the Module's Boundary)	Module	Encrypted	Automated	Electronic	KTS (TLSv1.2 with AES and HMAC)
Peer Public Key Input	External (Outside of the Module's Boundary)	Module	Plaintext	Automated	Electronic	
Module Public Key Output	Module	External (Outside of the Module's Boundary)	Plaintext	Automated	Electronic	
Password/Secret Input via SSHv2 encrypted by GCM	External (Outside of the Module's Boundary)	Module	Encrypted	Automated	Electronic	KTS (SSHv2 with AES-GCM)
Password/Secret Input via SSHv2 encrypted by AES and HMAC	External (Outside of the Module's Boundary)	Module	Encrypted	Automated	Electronic	KTS (SSHv2 with AES and HMAC)

Table 16: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Zeroization Command	CO issues zeroization service	the zeroization command will erase all SSPs stored in the DRAM or in the Flash of the module.	'configure factory-default'

Table 17: SSP Zeroization Methods

Please note that the Firmware Load Test Key is only used for Firmware Load Test Authentication and not subject to the zeroization requirement.

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
DRBG Entropy Input	Used to seed the DRBG	384 bits - at least 256 bits	Entropy Input - CSP			DRBG Function
DRBG Seed	Used in DRBG Generation	256 bits - 256 bits	DRBG Seed - CSP			DRBG Function
DRBG Internal State V value	Used in DRBG Generation	256 bits - 256 bits	DRBG Internal State V value - CSP			DRBG Function
DRBG Key	Used in DRBG Generation	256 bits - 256 bits	DRBG Key - CSP			DRBG Function
User Password	User authentication	8-30 Characters - 8-30 Characters	Authentication Data - CSP			
Crypto Officer Password	Crypto Officer authentication	8-30 Characters - 8-30 Characters	Authentication Data - CSP			
RADIUS Secret	RADIUS Server Authentication	16 Characters - 16 Characters	Authentication Data - CSP			

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
TACACS+ Secret	TACACS+ Authentication	16 Characters - 16 Characters	Authentication Data - CSP			
Firmware Load Test Key	Used for Firmware Load Test	112 bits - 112 bits	Public Key - CSP			Firmware Load Test
SSH DH Private Key	Used to derive the SSH DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112-152 bits	Private Key - CSP	KAS-FFC-KeyGen (SSHv2)		KAS-FFC (SSHv2)
SSH DH Public Key	Used to derive SSH DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112-152 bits	Public Key - PSP		KAS-FFC-KeyGen (SSHv2)	KAS-FFC (SSHv2)
SSH Peer DH Public Key	Used to derive SSH DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112-152 bits	Public Key - PSP			KAS-FFC (SSHv2)
SSH DH Shared Secret	Used to derive SSH Session Encryption Keys, SSH Session Authentication Keys	MODP-2048, MODP-3072, MODP-4096 - 112-152 bits	Shared Secret - CSP		KAS-FFC (SSHv2)	KAS-FFC (SSHv2)
SSH ECDH Private Key	Used to derive the SSH ECDH Shared Secret	Curves: 256, 384, 521 bits - 128 to 256 bits	Private Key - CSP	KAS-ECC-KeyGen (SSHv2)		KAS-ECC (SSHv2)
SSH ECDH Public Key	Used to derive SSH ECDHE	Curves: 256, 384, 521 bits -	Public Key - PSP		KAS-ECC-KeyGen (SSHv2)	KAS-ECC (SSHv2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	Shared Secret	128-256 bits				
SSH Peer ECDH Public Key	Used to derive SSH DH Shared Secret	Curves: 256, 384, 521 bits - 128 to 256 bits	Public Key - PSP			KAS-ECC (SSHv2)
SSH ECDH Shared Secret	Used to derive SSH Session Encryption Keys, SSH Session Authentication Keys	Curves: 256, 384, 521 bits - 128 to 256 bits	Shared Secret - CSP		KAS-ECC (SSHv2)	KAS-ECC (SSHv2)
SSH RSA Private Key	Used for SSH session authentication	Modulus 2048 and 3072 bits - 112-128 bits	Private Key - CSP	RSA KeyGen (SSHv2, TLSv1.2, IKEv2)		RSA SigVer (SSHv2, TLSv1.2, and IKEv2)
SSH RSA Public Key	Used for SSH sessions aiuthentication	Modulus 2048 and 3072 bits - 112-128 bits	Public Key - PSP		RSA KeyGen (SSHv2, TLSv1.2, IKEv2)	RSA SigVer (SSHv2, TLSv1.2, and IKEv2)
SSH ECDSA Private Key	Used for SSH session authentication	Curves: 256, 384, 521 bits - 128 to 256 bits	Private Key - CSP	ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2)		ECDSA SigGen (SSHv2, TLSv1.2 and IKEv2)
SSH ECDSA Public Key	Used for SSH sessions aiuthentication	Curves: 256, 384, 521 bits - 128 to 256 bits	Public Key - PSP		ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2)	ECDSA SigVer (SSHv2, TLSv1.2, and IKEv2)
SSH Session Encryption Key	Used for SSH Session confidentiality protection	128-256 bits - 128-256 bits	Session Key - CSP		SSHv2 Keying Materials Development	Block Cipher (SSHv2)
SSH Session Authentication Key	Used for SSH Session integrity protection	At least 160 bits - At least 160 bits	Session Key - CSP		SSHv2 Keying Materials Development	MAC (SSHv2)
TLS DH Private Key	Used to Derive TLS	Modulus: 2048,	Private Key - CSP	KAS-FFC-		KAS-FFC (TLSv1.2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	DH Shared Secret	3072, 4096 bits - 128-152 bits		KeyGen (TLSv1.2)		
TLS DH Public Key	Used to Derive TS DH Shared Secret	Modulus: 2048, 3072, or 4096 bits - 128-152 bits	Public Key - PSP		KAS-FFC-KeyGen (TLSv1.2)	KAS-FFC (TLSv1.2)
TLS Peer DH Public Key	Used to derive IKE DH Shared Secret	Modulus: 2048, 3072, or 4096 bits - 128-152 bits	Public Key - PSP			KAS-FFC (TLSv1.2)
TLS DH Shared Secret	Used to Derive TLS Session Encryption Key and TLS Session Authentication Key	Modulus 2048, 3072, or 4096 - 128-152 bits	Shared Secret - CSP		KAS-FFC (TLSv1.2)	KAS-FFC (TLSv1.2)
TLS ECDH Private Key	Used to Derive TLS ECDH Shared Secret	Curves P-256, P-384, and P-521 - 128-256 bits	Private Key - CSP	KAS-ECC-KeyGen (TLSv1.2)		KAS-ECC (TLSv1.2)
TLS ECDH Public Key	Used to Derive TS ECDH Shared Secret	Curves P-256, P-384, and P-521 - 128-256 bits	Public Key - PSP		KAS-ECC-KeyGen (TLSv1.2)	KAS-ECC (TLSv1.2)
TLS Peer ECDH Public Key	Used to derive IKE ECDH Shared Secret	Curves: P-256, P-384, P-521 - 128-256 bits	Public Key - PSP			KAS-ECC (TLSv1.2)
TLS ECDH Shared Secret	Used to Derive TLS Session Encryption Key and	Curves p-256, P-384, P-521 -	Shared Secret - CSP		KAS-ECC (TLSv1.2)	KAS-ECC (TLSv1.2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	TLS Session Authentication Key	128-256 bits				
TLS ECDSA Private Key	Used to support CO and Admin HTTPS interfaces	Curves P-256, P-384, P-521 - 128-256 bits	Private Key - CSP	ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2)		ECDSA SigGen (SSHv2, TLSv1.2 and IKEv2)
TLS ECDSA Public Key	Used to support CO and User HTTPS Interfaces	Curves P-256, P-384, P-521 - 128-256 bits	Public Key - PSP		ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2)	ECDSA SigVer (SSHv2, TLSv1.2, and IKEv2)
TLS RSA Private Key	Used to support CO and Admin HTTPS Interfaces	Modulus 2048 and 3072 bits - 112-128 bits	Private Key - CSP	RSA KeyGen (SSHv2, TLSv1.2, IKEv2)		RSA SigVer (SSHv2, TLSv1.2, and IKEv2)
TLS RSA Public Key	Used to support CO and User HTTPS interfaces	Modulus 2048 and 3072 bits - 112-128 bits	Public Key - PSP		RSA KeyGen (SSHv2, TLSv1.2, IKEv2)	RSA SigVer (SSHv2, TLSv1.2, and IKEv2)
TLS Master Secret	Used to protect HTTPS Session. Pre-master secret	At least 112 bits - At least 112 bits	Master Secret - CSP		TLS Keying Materials Development	TLS Keying Materials Development
TLS Session Encryption Key	Used to protect HTTPS Session. TLS Master secret	128-256 bits - 128-256 bits	Session Key - CSP		TLS Keying Materials Development	Block Cipher (TLSv1.2)
TLS Session Authentication Key	Used to protect HTTPS Session. TLS master secret	at least 112 bits - at least 112 bits	Session Key - CSP		TLS Keying Materials Development	MAC (TLSv1.2)
IPSec/IKE DH Private Key	Used to derive IPSec/IKE	MODP-2048, MODP-3072,	Private Key - CSP	KAS-FFC-KeyGen (IKEv2)		KAS-FFC (IKEv2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	DH Shared Secret	MODP-4096 - 112-152 bits				
IPSec/IKE DH Public Key	Used to derive IPSec/IKE DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112-152 bits	Public Key - PSP		KAS-FFC-KeyGen (IKEv2)	KAS-FFC (IKEv2)
IPSec/IKE Peer DH Public Key	Used to derive IPSec/IKE DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112-152 bits	Public Key - PSP			KAS-FFC (IKEv2)
IPSec/IKE DH Shared Secret	Used to derive IPSec/IKE Session Encryption Keys, IPSec/IKE Authentication Keys	MODP-2048, MODP-3072, MODP-4096 - 112-152 bits	Shared Secret - CSP		KAS-FFC (IKEv2)	KAS-FFC (IKEv2)
IPSec/IKE ECDH Private Key	Used to derive IPSec/IKE ECDH Shared Secrets	Curves P-256, P-384, P-521 - 128-256 bits	Private Key - CSP	KAS-ECC-KeyGen (IKEv2)		KAS-ECC (IKEv2)
IPSec/IKE ECDH Public Key	Used to derive IPSec/IKE ECDH Shared Secrets	Curves P-256, P-384, P-521 - 128-256 bits	Public Key - PSP		KAS-ECC-KeyGen (IKEv2)	KAS-ECC (IKEv2)
IPSec/IKE Peer ECDH Public Key	Used to derive IPSec/IKE ECDH Shared Secrets	Curves P-256, P-384, P-521 - 128-256 bits	Public Key - PSP			KAS-ECC (IKEv2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
IPSec/IKE ECDH Shared Secret	Used to derive IPSec/IKE ECDH Shared Secrets	Curves P-256, P-384, P-521 - 128-256 bits	Shared Secret - CSP		KAS-ECC (IKEv2)	KAS-ECC (IKEv2)
IPSec/IKE ECDSA Private Key	Used for IPSec/IKE peer authentication	Curves P-256, P-384, P-521 - 128-256 bits	Private Key - CSP	ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2)		ECDSA SigGen (SSHv2, TLSv1.2 and IKEv2)
IPSec/IKE ECDSA Public Key	Used for IPSec/IKE peer authentication	Curves P-256, P-384, P-521 - 128-256 bits	Public Key - PSP		ECDSA KeyGen (SSHv2, TLSv1.2 and IKEv2)	ECDSA SigVer (SSHv2, TLSv1.2, and IKEv2)
IPSec/IKE RSA Private Key	Used for IPSec/IKE peer authentication	Modulus 2048 or 3072 - 112 or 128 bits	Private Key - CSP	RSA KeyGen (SSHv2, TLSv1.2, IKEv2)		RSA SigGen (SSHv2, TLSv1.2, IKEv2)
IPSec/IKE RSA Public Key	Used for IPSec/IKE peer authentication	Modulus 2048 or 3072 - 112 or 128 bits	Public Key - PSP		RSA KeyGen (SSHv2, TLSv1.2, IKEv2)	RSA SigVer (SSHv2, TLSv1.2, and IKEv2)
IPSec/IKE Pre-shared Secret	Used for IPSec/IKE peer authentication	16-32 bytes characters - 16-32 bytes characters	shared secret - CSP			IKEv2 Keying Materials Development
SKEYSEED	Keying material used to derive the IPSec/IKE Session Encryption Key and IPSec/IKE Authentication Key	160 bits - 160 bits	Keying Material - CSP		IKEv2 Keying Materials Development	IKEv2 Keying Materials Development
IPSec/IKE Session	Used to secure	128-256 bits -	Session Key - CSP		IKEv2 Keying	Block Cipher

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Encryption Key	IPSec/IKEv2 session confidentiality	128-256 bits			Materials Development	(IPSec/IKEv2)
IPSec/IKE Authentication Key	Used to secure IPSec/IKEv2 session integrity	at least 160 bits - at least 160 bits	Session Key - CSP		IKEv2 Keying Materials Development	MAC (IPSec/IKEv2)
SNMPv3 Shared Secret	Used for SNMPv3 user authentication	8-32 characters - N/A	Authentication Secret - CSP			IKEv2 Keying Materials Development
SNMPv3 Encryption Key	Used to protect SNMPv3 traffic confidentiality	128 bits - 128 bits	Encryption Key - CSP		SNMPv3 Keying Materials Development	Block Cipher (SNMPv3)
SNMPv3 Authentication Key	Used to secure SNMPv3 traffic integrity	At least 112 bits - At least 112 bits	Authentication Key - CSP		SNMPv3 Keying Materials Development	MAC (SNMPv3)

Table 18: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
DRBG Entropy Input		DRAM:Plaintext	Until Reboot	Zeroization Command	DRBG Seed:Used With DRBG Internal State V value:Used With DRBG Key:Used With
DRBG Seed		DRAM:Plaintext	Until Reboot	Zeroization Command	DRBG Entropy Input:Used With DRBG Internal State V value:Used With DRBG Key:Used With
DRBG Internal State V value		DRAM:Plaintext	Until Reboot	Zeroization Command	DRBG Entropy Input:Used With DRBG Seed:Used With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					DRBG Key:Used With
DRBG Key		DRAM:Plaintext	Until Reboot	Zeroization Command	DRBG Entropy Input:Used With DRBG Seed:Used With DRBG Internal State V value:Used With
User Password	Password/Secret Input via TLS encrypted by GCM Password/Secret Input via TLS encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by GCM Password/Secret Input via SSHv2 encrypted by AES and HMAC	Flash:Encrypted		Zeroization Command	
Crypto Officer Password	Password/Secret Input via TLS encrypted by GCM Password/Secret Input via TLS encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by GCM Password/Secret Input via SSHv2 encrypted by	Flash:Encrypted		Zeroization Command	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	AES and HMAC				
RADIUS Secret	Password/Secret Input via TLS encrypted by GCM Password/Secret Input via TLS encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by GCM Password/Secret Input via SSHv2 encrypted by AES and HMAC	Flash:Plaintext		Zeroization Command	
TACACS+ Secret	Password/Secret Input via TLS encrypted by GCM Password/Secret Input via TLS encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by GCM Password/Secret Input via SSHv2 encrypted by AES and HMAC	Flash:Plaintext		Zeroization Command	
Firmware Load Test Key		Flash:Plaintext		N/A	
SSH DH Private Key		DRAM:Plaintext	While SSH tunnel is on	Zeroization Command	SSH DH Public Key:Paired With SSH Peer DH

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Public Key:Used With
SSH DH Public Key	Module Public Key Output	DRAM:Plaintext	While SSH tunnel is on	Zeroization Command	SSH DH Private Key:Paired With
SSH Peer DH Public Key	Peer Public Key Input	DRAM:Plaintext	While SSH tunnel is on	Zeroization Command	SSH DH Private Key:Used With
SSH DH Shared Secret		DRAM:Plaintext	While SSH tunnel is on	Zeroization Command	SSH DH Private Key:Derived From SSH DH Public Key:Derived From
SSH ECDH Private Key		DRAM:Plaintext	While SSH tunnel is on	Zeroization Command	SSH ECDH Public Key:Paired With SSH Peer ECDH Public Key:Used With
SSH ECDH Public Key	Module Public Key Output	DRAM:Plaintext	While SSH tunnel is on	Zeroization Command	SSH ECDH Private Key:Paired With
SSH Peer ECDH Public Key	Peer Public Key Input	DRAM:Plaintext	While SSH tunnel is on	Zeroization Command	SSH ECDH Private Key:Used With
SSH ECDH Shared Secret		DRAM:Plaintext	While SSH tunnel is on	Zeroization Command	SSH ECDH Private Key:Derived From SSH ECDH Public Key:Derived From
SSH RSA Private Key		Flash:Plaintext		Zeroization Command	SSH RSA Public Key:Paired With SSH Peer RSA Public Key:Used With
SSH RSA Public Key	Module Public Key Output	Flash:Plaintext		Zeroization Command	SSH RSA Private Key:Paired With
SSH ECDSA Private Key		Flash:Plaintext		Zeroization Command	SSH ECDSA Public Key:Paired With
SSH ECDSA Public Key	Module Public Key Output	Flash:Plaintext		Zeroization Command	SSH ECDSA Private Key:Paired With
SSH Session Encryption Key		DRAM:Plaintext	While SSH tunnel is on	Zeroization Command	SSH Session Authentication Key:Used With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
SSH Session Authentication Key		DRAM:Plaintext	While SSH tunnel is on	Zeroization Command	SSH Session Encryption Key:Used With
TLS DH Private Key		DRAM:Plaintext	While TLS tunnel is on	Zeroization Command	TLS DH Public Key:Paired With TLS Peer DH Public Key:Used With
TLS DH Public Key	Module Public Key Output	DRAM:Plaintext	While TLS tunnel is on	Zeroization Command	TLS DH Private Key:Paired With
TLS Peer DH Public Key	Peer Public Key Input	DRAM:Plaintext	while TLS tunnel is on	Zeroization Command	TLS DH Private Key:Used With
TLS DH Shared Secret		DRAM:Plaintext	While TLS tunnel is on	Zeroization Command	TLS ECDH Private Key:Derived From TLS Peer ECDH Public Key:Derived From
TLS ECDH Private Key		DRAM:Plaintext	While TLS tunnel is on	Zeroization Command	TLS ECDH Public Key:Paired With TLS Peer ECDH Public Key:Used With
TLS ECDH Public Key	Module Public Key Output	DRAM:Plaintext	While TLS tunnel is on	Zeroization Command	TLS ECDH Private Key:Paired With
TLS Peer ECDH Public Key	Peer Public Key Input	DRAM:Plaintext	while TLS tunnel is on	Zeroization Command	TLS ECDH Private Key:Used With
TLS ECDH Shared Secret		DRAM:Plaintext	While TLS tunnel is on	Zeroization Command	TLS ECDH Private Key:Derived From TLS Peer ECDH Public Key:Derived From
TLS ECDSA Private Key		Flash:Plaintext		Zeroization Command	TLS ECDSA Public Key:Paired With
TLS ECDSA Public Key	Module Public Key Output	Flash:Plaintext		Zeroization Command	TLS ECDSA Private Key:Paired With
TLS RSA Private Key		Flash:Plaintext		Zeroization Command	TLS RSA Public Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
TLS RSA Public Key	Module Public Key Output	Flash:Plaintext		Zeroization Command	TLS RSA Private Key:Paired With
TLS Master Secret		DRAM:Plaintext	While TLS tunnel is on	Zeroization Command	TLS ECDH Shared Secret:Derived From
TLS Session Encryption Key		DRAM:Plaintext	While TLS tunnel is on	Zeroization Command	TLS Session Authentication Key:Used With
TLS Session Authentication Key		DRAM:Plaintext	While TLS tunnel is on	Zeroization Command	TLS Session Encryption Key:Used With
IPSec/IKE DH Private Key		DRAM:Plaintext	While IPSec/IKEv2 tunnel is on	Zeroization Command	IPSec/IKE DH Public Key:Paired With IPSec/IKE Peer DH Public Key:Used With
IPSec/IKE DH Public Key	Module Public Key Output	DRAM:Plaintext	While IPSec/IKEv2 tunnel is on	Zeroization Command	IPSec/IKE DH Private Key:Paired With
IPSec/IKE Peer DH Public Key	Peer Public Key Input	DRAM:Plaintext	while IPSec/IKE tunnel is on	Zeroization Command	IPSec/IKE DH Private Key:Used With
IPSec/IKE DH Shared Secret		DRAM:Plaintext	While IPSec/IKEv2 tunnel is on	Zeroization Command	SKEYSEED:Used With
IPSec/IKE ECDH Private Key		DRAM:Plaintext	While IPSec/IKEv2 tunnel is on	Zeroization Command	IPSec/IKE ECDH Public Key:Paired With IPSec/IKE Peer ECDH Public Key:Used With
IPSec/IKE ECDH Public Key	Module Public Key Output	DRAM:Plaintext	While IPSec/IKEv2 tunnel is on	Zeroization Command	IPSec/IKE ECDH Private Key:Paired With
IPSec/IKE Peer ECDH Public Key	Peer Public Key Input	DRAM:Plaintext	While IPSec/IKEv2 tunnel is on	Zeroization Command	IPSec/IKE ECDH Private Key:Used With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
IPSec/IKE ECDH Shared Secret		DRAM:Plaintext	While IPSec/IKE v2 tunnel is on	Zeroization Command	SKEYSEED:Used With
IPSec/IKE ECDSA Private Key		Flash:Plaintext		Zeroization Command	IPSec/IKE ECDSA Public Key:Paired With
IPSec/IKE ECDSA Public Key	Module Public Key Output	Flash:Plaintext		Zeroization Command	IPSec/IKE ECDSA Private Key:Paired With
IPSec/IKE RSA Private Key		Flash:Plaintext		Zeroization Command	IPSec/IKE RSA Public Key:Paired With
IPSec/IKE RSA Public Key	Module Public Key Output	Flash:Plaintext		Zeroization Command	IPSec/IKE RSA Private Key:Paired With
IPSec/IKE Pre-shared Secret		DRAM:Plaintext	While IPSec/IKE v2 tunnel is on	Zeroization Command	SKEYSEED:Derived to
SKEYSEED		DRAM:Plaintext	While IPSec/IKE v2 tunnel is on	Zeroization Command	IPSec/IKE DH Shared Secret:Derived From IPSec/IKE ECDH Shared Secret:Derived From IPSec/IKE Pre-shared Secret:Derived From
IPSec/IKE Session Encryption Key		DRAM:Plaintext	While IPSec/IKE v2 tunnel is on	Zeroization Command	IPSec/IKE DH Shared Secret:Derived From IPSec/IKE ECDH Shared Secret:Derived From
IPSec/IKE Authentication Key		DRAM:Plaintext	While IPSec/IKE v2 tunnel is on	Zeroization Command	IPSec/IKE DH Shared Secret:Derived From IPSec/IKE ECDH Shared Secret:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
SNMPv3 Shared Secret	Password/Secret Input via TLS encrypted by GCM Password/Secret Input via TLS encrypted by AES and HMAC Password/Secret Input via SSHv2 encrypted by GCM Password/Secret Input via SSHv2 encrypted by AES and HMAC	DRAM:Plaintext	While SNMPv3 tunnel is on	Zeroization Command	SNMPv3 Encryption Key:Derive To SNMPv3 Authentication Key:Derive To
SNMPv3 Encryption Key		DRAM:Plaintext	While SNMPv3 tunnel is on	Zeroization Command	SNMPv3 Shared Secret:Derived From
SNMPv3 Authentication Key		DRAM:Plaintext	While SNMPv3 tunnel is on	Zeroization Command	SNMPv3 Shared Secret:Derived From SNMPv3 Encryption Key:Used With

Table 19: SSP Table 2

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
RSA SigVer (FIPS186-4) (A4446)	RSA SigVer 2048 bits with SHA2-512	KAT	SW/FW Integrity	Module is in normal state	RSA SigVer
Pre-Operational Bypass Test	N/A	N/A	Bypass	Module is in normal state	N/A

Table 20: Pre-Operational Self-Tests

The module performs the following self-tests, including the pre-operational self-tests and Conditional self-tests. Prior to the module providing any data output via the data output interface, the module performs and passes the pre-operational self-tests. Following the successful pre-operational self-tests, the module executes the Conditional Cryptographic Algorithm Self-tests (CASTs). If anyone of the self-tests fails, the module transitions into an error state and outputs the error message via the module's status output interface. While the module is in the error state, all data through the data output interface and all cryptographic operations are disabled. The error state can only be cleared by reloading the module. All self-tests must be completed successfully before the module transitions to the operational state.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CBC (A4446)	256 bits	KAT	CAST	Module is in normal state	Encrypt	Power Up
AES-CBC (A4446)	256 bits	KAT	CAST	Module is in normal state	Decrypt	Power Up
AES-GCM (A4446)	256 bits	KAT	CAST	Module is in normal state	Authenticated Encrypt	Power Up
AES-GCM (A4446)	256 bits	KAT	CAST	Module is in normal state	Authenticated Decrypt	Power Up
Counter DRBG (A4446)	AES-128	KAT	CAST	Module is in normal state	Instantiate KAT	Power Up
Counter DRBG (A4446)	AES-128	KAT	CAST	Module is in normal state	Generate KAT	Power Up
Counter DRBG (A4446)	AES-128	KAT	CAST	Module is in normal state	Reseed KAT	Power Up
ECDSA SigGen (FIPS186-4) (A4446)	P-256 curve with SHA2-256	KAT	CAST	Module is in normal state	ECDSA SigGen KAT	Power Up
ECDSA SigVer (FIPS186-4) (A4446)	P-256 curve with SHA2-256	KAT	CAST	Module is in normal state	ECDSA SigVer KAT	Power Up
HMAC-SHA-1 (A4446)	SHA-1	KAT	CAST	Module is in normal state	HMAC-SHA-1	Power Up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-256 (A4446)	SHA2-256	KAT	CAST	Module is in normal state	HMAC-SHA2-256	Power Up
HMAC-SHA2-384 (A4446)	SHA2-384	KAT	CAST	Module is in normal state	HMAC-SHA2-384	Power Up
HMAC-SHA2-512 (A4446)	SHA2-512	KAT	CAST	Module is in normal state	HMAC-SHA2-512	Power Up
KAS-ECC-SSC Sp800-56Ar3 (A4446)	P-256 Curve	KAT	CAST	Module is in normal state	Primitive Z KAT	Power Up
KAS-FFC-SSC Sp800-56Ar3 (A4446)	MODP-2048	KAT	CAST	Module is in normal state	Primitive Z KAT	Power Up
RSA SigGen (FIPS186-4) (A4446)	2048 bit modulus with SHA2-256	KAT	CAST	Module is in normal state	RSA SigGen KAT	Power Up
RSA SigVer (FIPS186-4) (A4446)	2048 bit modulus with SHA2-256	KAT	CAST	Module is in normal state	RSA SigVer KAT	Power Up
KDF IKEv2 (A4446)	N/A	KAT	CAST	Module is in normal state	N/A	Power Up
KDF SNMP (A4446)	N/A	KAT	CAST	Module is in normal state	N/A	Power Up
KDF SSH (A4446)	N/A	KAT	CAST	Module is in normal state	N/A	Power Up
TLS v1.2 KDF RFC7627 (A4446)	N/A	KAT	CAST	Module is in normal state	N/A	Power Up
SHA-1 (A4446)	Message Length: 0-65536 Increment 8	KAT	CAST	Module is in normal state	N/A	Power Up
AES-CBC (C1026)	128 bits	KAT	CAST	Module is in normal state	Encrypt KAT	Power Up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CBC (C1026)	128 bits	KAT	CAST	Module is in normal state	Decrypt KAT	Power Up
AES-GCM (C1026)	128 bits	KAT	CAST	Module is in normal state	Encrypt KAT	Power Up
AES-GCM (C1026)	128 bits	KAT	CAST	Module is in normal state	Decrypt KAT	Power Up
Hash DRBG (C1026)	SHA2-512	KAT	CAST	Module is in normal state	Instantiate KAT	Power Up
Hash DRBG (C1026)	SHA2-512	KAT	CAST	Module is in normal state	Generate KAT	Power Up
Hash DRBG (C1026)	SHA2-512	KAT	CAST	Module is in normal state	Reseed KAT	Power Up
HMAC-SHA-1 (C1026)	SHA-1	KAT	CAST	Module is in normal state	HMAC-SHA-1	Power Up
HMAC-SHA2-256 (C1026)	SHA2-256	KAT	CAST	Module is in normal state	HMAC-SHA2-256	Power Up
HMAC-SHA2-384 (C1026)	SHA2-384	KAT	CAST	Module is in normal state	HMAC-SHA2-384	Power Up
HMAC-SHA2-512 (C1026)	SHA2-512	KAT	CAST	Module is in normal state	HMAC-SHA2-512	Power Up
SHA-1 (C1026)	Message Length: 0-51200 Increment 8	KAT	CAST	Module is in normal state	SHA-1	Power Up
ECDSA KeyGen (FIPS186-4) (A4446)	Curve P-256 with SHA2-256	PCT	PCT	Module is in normal state	ECDSA	Performs all required pair-wise consistency tests on the newly generated key pairs before the first operational use.
RSA KeyGen	2048 bit Modulus	PCT	PCT	Module is in normal state	RSA	Performs all required pair-wise

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
(FIPS186-4) (A4446)						consistency tests on the newly generated key pairs before the first operational use.
KAS-ECC-SSC Sp800-56Ar3 (A4446)	Curve P-256 with SHA2-256	PCT	PCT	Module is in normal state	N/A	Performs all required pair-wise consistency tests on the newly generated key pairs before the first operational use.
KAS-FFC-SSC Sp800-56Ar3 (A4446)	MODP-2048	PCT	PCT	Module is in normal state	N/A	Performs all required pair-wise consistency tests on the newly generated key pairs before the first operational use.
HMAC-SHA2-512 (A4446)	HMAC-SHA2-512	KAT	SW/FW Load	Module is in normal state	N/A	When firmware has been uploaded to the module
Conditional Bypass	N/A	N/A	Bypass	Module is in normal state	N/A	Performs conditional bypass test before first operational use of bypass service

Table 21: Conditional Self-Tests

The module performs on-demand self-tests initiated by the operator, by powering off and powering the module back on. The full suite of self-tests is then executed. The same procedure may be employed by the operator to perform periodic self-tests.

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigVer (FIPS186-4) (A4446)	KAT	SW/FW Integrity	Recommend 60 Days	Reboot
Pre-Operational Bypass Test	N/A	Bypass	Recommend 60 Days	Reboot

Table 22: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CBC (A4446)	KAT	CAST	Recommend 60 Days	Reboot
AES-CBC (A4446)	KAT	CAST	Recommend 60 Days	Reboot
AES-GCM (A4446)	KAT	CAST	Recommend 60 Days	Reboot
AES-GCM (A4446)	KAT	CAST	Recommend 60 Days	Reboot
Counter DRBG (A4446)	KAT	CAST	Recommend 60 Days	Reboot
Counter DRBG (A4446)	KAT	CAST	Recommend 60 Days	Reboot
Counter DRBG (A4446)	KAT	CAST	Recommend 60 Days	Reboot
ECDSA SigGen (FIPS186-4) (A4446)	KAT	CAST	Recommend 60 Days	Reboot
ECDSA SigVer (FIPS186-4) (A4446)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA-1 (A4446)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA2-256 (A4446)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA2-384 (A4446)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA2-512 (A4446)	KAT	CAST	Recommend 60 Days	Reboot
KAS-ECC-SSC Sp800-56Ar3 (A4446)	KAT	CAST	Recommend 60 Days	Reboot
KAS-FFC-SSC Sp800-56Ar3 (A4446)	KAT	CAST	Recommend 60 Days	Reboot

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigGen (FIPS186-4) (A4446)	KAT	CAST	Recommend 60 Days	Reboot
RSA SigVer (FIPS186-4) (A4446)	KAT	CAST	Recommend 60 Days	Reboot
KDF IKEv2 (A4446)	KAT	CAST	Recommend 60 Days	Reboot
KDF SNMP (A4446)	KAT	CAST	Recommend 60 Days	Reboot
KDF SSH (A4446)	KAT	CAST	Recommend 60 Days	Reboot
TLS v1.2 KDF RFC7627 (A4446)	KAT	CAST	Recommend 60 Days	Reboot
SHA-1 (A4446)	KAT	CAST	Recommend 60 Days	Reboot
AES-CBC (C1026)	KAT	CAST	Recommend 60 Days	Reboot
AES-CBC (C1026)	KAT	CAST	Recommend 60 Days	Reboot
AES-GCM (C1026)	KAT	CAST	Recommend 60 Days	Reboot
AES-GCM (C1026)	KAT	CAST	Recommend 60 Days	Reboot
Hash DRBG (C1026)	KAT	CAST	Recommend 60 Days	Reboot
Hash DRBG (C1026)	KAT	CAST	Recommend 60 Days	Reboot
Hash DRBG (C1026)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA-1 (C1026)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA2-256 (C1026)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA2-384 (C1026)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA2-512 (C1026)	KAT	CAST	Recommend 60 Days	Reboot
SHA-1 (C1026)	KAT	CAST	Recommend 60 Days	Reboot
ECDSA KeyGen (FIPS186-4) (A4446)	PCT	PCT	Recommend 60 Days	Reboot
RSA KeyGen (FIPS186-4) (A4446)	PCT	PCT	Recommend 60 Days	Reboot

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KAS-ECC-SSC Sp800-56Ar3 (A4446)	PCT	PCT	Recommend 60 Days	Reboot
KAS-FFC-SSC Sp800-56Ar3 (A4446)	PCT	PCT	Recommend 60 Days	Reboot
HMAC-SHA2-512 (A4446)	KAT	SW/FW Load	N/A	N/A
Conditional Bypass	N/A	Bypass	N/A	N/A

Table 23: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Error State	If self-test tests fail, the module is put into an error state	Self-test failure	Reboot the module	System Halt

Table 24: Error States

If any of the above-mentioned self-tests fail, the module reports the error and enters the Error state. In the Error State, no cryptographic services are provided, and data output is prohibited. The only method to recover from the error state is to reboot the module and perform the self-tests, including the pre-operational firmware integrity test and the conditional CASTs. The module will only enter into the operational state after successfully passing the pre-operational firmware integrity test and the conditional CASTs.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The validated module firmware was installed onto the respective test platforms listed in Table 2 above. The Crypto Officer must configure and enforce the following initialization steps:

Step 1: The Crypto Officer must install opacity shields as described in section 7 above.

Step 2: The Crypto Officer must apply tamper evidence labels as described in section 7 above.

Step 3: The Crypto Officer must securely store any unused tamper evidence labels.

Note: Each module has a Type A USB 2.0 port, but it is considered to be disabled once the Crypto Officer has applied the TEL #9.

Step 4: Crypto Officer performs the following configurations:

```
ciscoasa# configure terminal
```

Note, the Crypto Officer needs to connect the platform to cisco.com to obtain the license for ASA from Cisco.

```
ciscoasa(config)# license smart register idtoken [token data]
ciscoasa(config)#license smart
ciscoasa(config-smart-lic)# show license all
Smart Licensing Status
=====================
Smart Licensing is ENABLED
```

-OR-

```
ciscoasa(config-smart-lic)# show license summary
Smart Licensing is ENABLED
Registration:
```

Step 5: Enable “Approved Mode” to allow the module to startup the cryptographic module, such as run power-on self-tests and bypass test by using the following command:

```
ciscoasa(config)# fips enable
```

Note: Startup operational mode will not take effect until you save configuration and reboot the device

Rebooting the device will force new self-test

Step 6: Crypto Officer can verify the version installed and running

```
ciscoasa(config)# show version
```

Step 7: Crypto Officer will need to configure ASA

```
ciscoasa> en
ciscoasa# conf t
ciscoasa(config)#
```

Step 8: Assign users a Privilege Level of 1.

Step 9: Configure IP address for unit and all distant endpoints.

Step 10: Define RADIUS and TACACS+ shared secret keys that are at least 8 characters long and secure traffic between the security module and the RADIUS/TACACS+ server via secure (IPSec, TLS) tunnel.

Note: Perform this step only if RADIUS/TACAS+ is configured, otherwise skip over and proceed to next step.

Step 11: Configure the security module so that any remote connections via Telnet are secured through IPSec.

Step 12: Configure the security module so that only approved algorithms are used for IPsec tunnels.

Step 13: Configure the security module so that error messages can only be viewed by Crypto Officer.

Step 14: Disable the TFTP server.

Step 15: Disable HTTP for performing system management in approved mode of operation. HTTPS with TLS should always be used for Web-based management.

Step 16: Ensure that installed digital certificates are signed using approved algorithms.

11.2 Administrator Guidance

No specific Administrator guidance.

11.3 Non-Administrator Guidance

No specific Non-Administrator guidance.

12 Mitigation of Other Attacks

N/A for this module.