Cisco Systems, Inc.

Cisco Adaptive Security Appliance Virtual Cryptographic Module

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 Virtual Cryptographic Module (hereinafter referred to as ASAv or the Module), firmware version 9.20(3). 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 1 firmware hybrid 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	1
2	Cryptographic module specification	1
3	Cryptographic module interfaces	1
4	Roles, services, and authentication	1
5	Software/Firmware security	1
6	Operational environment	1
7	Physical security	1
8	Non-invasive security	N/A
9	Sensitive security parameter management	1
10	Self-tests	1
11	Life-cycle assurance	1
12	Mitigation of other attacks	N/A
	Overall Level	1

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

This module is a multi-chip standalone firmware hybrid cryptographic module deployed as the virtualized version of the Cisco Adaptive Security Appliance (ASA) with underlying operating system identified as Linux 4 (also referred to as Firepower eXtensible Operating System or FX-OS throughout this document). 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. This 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, SSHv2, HTTPS/TLSv1.2, IPsec/IKEv2, SNMPv3 and Cryptographic Cipher Suite B.

Module Type: Firmware-hybrid

Module Embodiment: MultiChipStand

Module Characteristics:

Cryptographic Boundary:

The cryptographic module (red dash box) is a non-modifiable, multi-chip standalone firmware hybrid cryptographic module providing cryptographic support which takes data in and out from the host application via the API.

The block diagram below shows the boundary of the Tested Operational Environment's Physical Perimeter (TOEPP) being defined as the physical perimeter of the tested platform enclosure around which everything runs. The cryptographic boundary is the module (red dash box) and its interfaces with the operational environment.

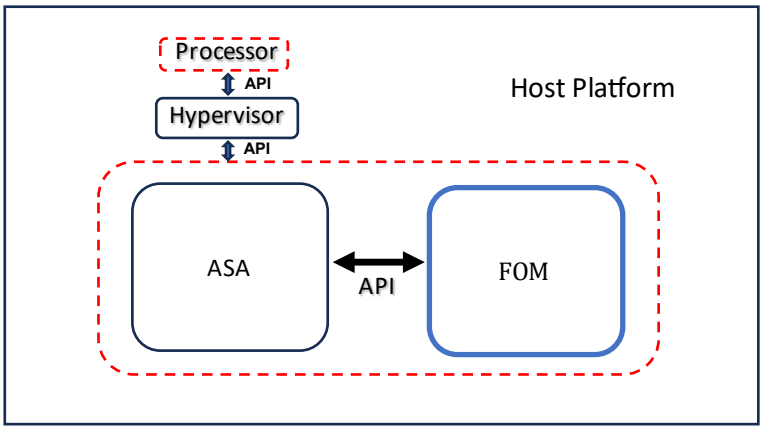

Figure 1 Block Diagram

The Block Diagram above comprises the following components

- Processor: Chip on the tested platforms handle all processes.
- API: Host API between hypervisor and processor
- Hypervisor: VMWare ESXi 7.0
- API: Host API between hypervisor and the ASA Module
- ASA: Adaptive Security Application
- API: Guest API between the ASA Module and FOM Crypto library
- FOM: Cisco FIPS Object Module (FOM Crypto Library)

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

N/A for this module.

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

Package or File Name	Software/ Firmware Version	Features	Integrity Test
asav9-20-3.zip	9.20(3)		RSA 2048 SigVer with SHA2-512

Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets)

Tested Module Identification – Hybrid Disjoint Hardware:

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
Intel Xeon Platinum 8160 (Skylake)	1.0	N/A	Intel Xeon Platinum 8160 (Skylake)	

Table 3: Tested Module Identification – Hybrid Disjoint Hardware

Tested Operational Environments - Software, Firmware, Hybrid:

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Linux 4 (FX-OS) on VMware ESXi 7.0	UCS C220 M5 SFF Server	Intel Xeon Platinum 8160 (Skylake)	Yes	VMware ESXi 7.0	9.20(3)

Table 4: Tested Operational Environments - Software, Firmware, Hybrid

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

Operating System	Hardware Platform
Linux 4 (FX-OS)	UCS C220 M6 SFF Server w/ESXi 7.0
Linux 4 (FX-OS)	UCS C220 M7 SFF Server w/ESXi 7.0
Linux 4 (FX-OS)	UCS C225 M6 SFF Server w/ESXi 7.0
Linux 4 (FX-OS)	UCS C240 M5 SFF Server w/ESXi 7.0
Linux 4 (FX-OS)	UCS C240 M6 SFF Server w/ESXi 7.0
Linux 4 (FX-OS)	UCS C480 M5 SFF Server w/ESXi 7.0
Linux 4 (FX-OS)	UCS-E1100D M6 SFF Server w/ESXi 7.0

Table 5: Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid

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	The module is always in the approved mode of operation after initial operations are performed.	Approved	Approved mode indicator: "FIPS is currently enabled."

Table 6: Modes List and Description

The module has one Approved mode of operation and does not implement a Non-Approved mode of operation. Once the module is configured in the Approved mode of operation by following the steps in section 11 of this document, the module will only operate in the Approved mode of operation. The module doesn't claim the implementation of a degraded mode operation.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A4595	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38A
AES-GCM	A4595	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 256	SP 800-38D
Counter DRBG	A4595	Prediction Resistance - Yes Mode - AES-256 Derivation Function Enabled - Yes	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-4)	A4595	Curve - P-256, P-384, P-521 Secret Generation Mode - Testing Candidates	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A4595	Curve - P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A4595	Curve - P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-4
HMAC-SHA-1	A4595	Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA2-224	A4595	Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4595	Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4595	Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A4595	Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4595	Domain Parameter Generation Methods - P-256, P-384, P-521 Scheme -	SP 800-56A Rev. 3

Algorithm	CAVP Cert	Properties	Reference
		ephemeralUnified - KAS Role - initiator, responder	
KAS-FFC-SSC Sp800-56Ar3	A4595	Domain Parameter Generation Methods - ffdhe2048, ffdhe3072, ffdhe4096, modp-2048, modp-3072, modp-4096 Scheme - dhEphem - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDF IKEv2 (CVL)	A4595	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)	A4595	Password Length - Password Length: 256, 64	SP 800-135 Rev. 1
KDF SSH (CVL)	A4595	Cipher - AES-128, AES-192, AES-256	SP 800-135 Rev. 1
RSA KeyGen (FIPS186-4)	A4595	Key Generation Mode - B.3.4 Modulo - 2048, 3072	FIPS 186-4
RSA SigGen (FIPS186-4)	A4595	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072	FIPS 186-4
RSA SigVer (FIPS186-4)	A4595	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072	FIPS 186-4
Safe Primes Key Generation	A4595	Safe Prime Groups - ffdhe2048, ffdhe3072, ffdhe4096, modp-2048, modp-3072, modp-4096	SP 800-56A Rev. 3
SHA-1	A4595	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-224	A4595	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A4595	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A4595	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A4595	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
TLS v1.2 KDF RFC7627 (CVL)	A4595	Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1

Table 7: Approved Algorithms

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Key Type:Asymmetric	N/A	The Module performs Cryptographic Key Generation (CKG) for asymmetric keys as detailed by example 1 in section 4 and section 5 of SP800-133r2

Table 8: 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-FFC (SSHv2)	CKG KAS-Full	Full KAS-FFC Key Agreement used for SSHv2 service	Caveat:Key establishment methodology provides between 112 and 152 bits of security strength IG : IG D.F Path 2, Scenario 2, Split Key Confirmation : No Key Derivation : IG 2.4.B SP 800-135rev1 CVL	KAS-FFC-SSC Sp800-56Ar3: (A4595) Domain Parameter Generation: MODP-2048, MODP-3072, MODP-4096 Safe Primes Key Generation: (A4595) KDF SSH: (A4595) Counter DRBG: (A4595) CKG: () Key Type: Asymmetric
KAS-ECC (SSHv2)	CKG KAS-Full	Full KAS-ECC Key Agreement used for SSHv2 service	Caveat:Key establishment methodology provides between 128 and 256 bits of security strength IG : IG D.F Scenario 2, Path 2, Split Key Confirmation : No Key Derivation : IG 2.4.B SP 800-135rev1 CVL	KAS-ECC-SSC Sp800-56Ar3: (A4595) Curves: P-256, P-384, P-521 KDF SSH: (A4595) Counter DRBG: (A4595) CKG: () Key Type: Asymmetric

Name	Type	Description	Properties	Algorithms
KAS-FFC (TLSv1.2)	CKG KAS-Full	Full KAS-FFC Key Agreement used for TLSv1.2 service	Caveat:Key establishment methodology provides between 112 and 152 bits of security strength IG : IG D.F Path 2, Scenario 2, Split Key Confirmation : No Key Derivation : IG 2.4.B SP 800-135rev1 CVL	KAS-FFC-SSC Sp800-56Ar3: (A4595) Domain Parameter Generation: ffdhe2048, ffdhe3072, ffdhe4096 Safe Primes Key Generation: (A4595) TLS v1.2 KDF RFC7627: (A4595) Counter DRBG: (A4595) CKG: () Key Type: Asymmetric
KAS-ECC (TLSv1.2)	CKG KAS-Full	Full KAS-ECC Key Agreement used for TLSv1.2 service	Caveat:Key establishment methodology provides between 128 and 256 bits of security strength IG : IG D.F Scenario 2, Path 2, Split Key Confirmation : No Key Derivation : IG 2.4.B SP 800-135rev1 CVL	KAS-ECC-SSC Sp800-56Ar3: (A4595) Curves: P-256, P-384, P-521 TLS v1.2 KDF RFC7627: (A4595) Counter DRBG: (A4595) CKG: () Key Type: Asymmetric
KAS-FFC (IKEv2)	CKG KAS-Full	Full KAS-FFC Key Agreement used for IKEv2 service	Caveat:Key establishment methodology provides between 112 and 152 bits of security strength IG : IG D.F Path 2, Scenario 2, Split Key Confirmation : No Key Derivation : IG 2.4.B SP 800-135rev1 CVL	KAS-FFC-SSC Sp800-56Ar3: (A4595) Domain Parameter Generation: MODP-2048, MODP-3072, MODP-4096 Safe Primes Key Generation: (A4595) KDF IKEv2: (A4595) Counter DRBG: (A4595) CKG: ()

Name	Type	Description	Properties	Algorithms
				Key Type: Asymmetric
KAS-ECC (IKEv2)	CKG KAS-Full	Full KAS-ECC Key Agreement used for IKEv2 service	Caveat:Key establishment methodology provides between 128 and 256 bits of security strength IG : IG D.F Scenario 2, Path 2, Split Key Confirmation : No Key Derivation : IG 2.4.B SP 800-135rev1 CVL	KAS-ECC-SSC Sp800-56Ar3: (A4595) Curves: P-256, P-384, P-521 KDF IKEv2: (A4595) Counter DRBG: (A4595) CKG: () Key Type: Asymmetric
KTS (TLSv1.2 with AES and HMAC)	KTS-Unwrap KTS-Wrap	KTS via TLSv1.2 service by using AES and HMAC	Caveat:Key establishment methodology provides 128 or 256 bits of security strength Standard:SP 800-38F IG D.G:"combination" method: use any approved symmetric encryption mode together with an approved authentication method	AES-CBC: (A4595) Key Length: 128, 256 HMAC-SHA-1: (A4595) HMAC-SHA2-256: (A4595) HMAC-SHA2-384: (A4595) SHA-1: (A4595) SHA2-256: (A4595) SHA2-384: (A4595)
KTS (TLSv1.2 with AES-GCM)	KTS-Unwrap KTS-Wrap	KTS via TLSv1.2 service by using AES-GCM	Caveat:Key establishment methodology provides 128 or 256 bits of security strength Standard:SP 800-38F IG D.G:method: use of any approved authenticated symmetric encryption mode	AES-GCM: (A4595) Key Length: 128, 256

Name	Type	Description	Properties	Algorithms
KTS (SSHv2 with AES and HMAC)	KTS-Unwrap KTS-Wrap	KTS via SSHv2 service by using AES and HMAC	Caveat:Key establishment methodology provides 128 or 256 bits of security strength Standard:SP 800-38F IG D.G:"combination" method: use any approved symmetric encryption mode together with an approved authentication method	AES-CBC: (A4595) Key Length: 128, 256 HMAC-SHA-1: (A4595) HMAC-SHA2-256: (A4595) SHA-1: (A4595) SHA2-256: (A4595)
KTS (SSHv2 with AES-GCM)	KTS-Unwrap KTS-Wrap	KTS via SSHv2 service by using AES-GCM	Caveat:Key establishment methodology provides 128 or 256 bits of security strength Standard:SP 800-38F IG D.G:method: use of any approved authenticated symmetric encryption mode	AES-GCM: (A4595) Key Length: 128, 256
RSA KeyGen (SSHv2, TLSv1.2, IKEv2)	AsymKeyPair-KeyGen CKG	RSA KeyGen for SSHv2, TLSv1.2, and IKEv2 services		RSA KeyGen (FIPS186-4): (A4595) Modulus: 2048, 3072 bits Counter DRBG: (A4595) CKG: () Key Type: Asymmetric
ECDSA KeyGen (SSHv2, TLSv1.2, IKEv2)	AsymKeyPair-KeyGen CKG	ECDSA KeyGen for SSHv2, TLSv1.2, and IKEv2 services		ECDSA KeyGen (FIPS186-4): (A4595) Curves: P-256, P-384, P-521 Counter DRBG: (A4595)

Name	Type	Description	Properties	Algorithms
				CKG: () Key Type: Asymmetric
RSA SigGen (SSHv2, TLSv1.2, IKEv2)	DigSig-SigGen	RSA SigGen for SSHv2, TLSv1.2, and IKEv2 services		RSA SigGen (FIPS186-4): (A4595) Modulus: 2048, 3072 bits
ECDSA SigGen (SSHv2, TLSv1.2, IKEv2)	DigSig-SigGen	ECDSA SigGen for SSHv2, TLSv1.2, and IKEv2 services		ECDSA SigGen (FIPS186-4): (A4595) Curves: P-256, P-384, P-521
RSA SigVer (SSHv2, TLSv1.2, IKEv2)	DigSig-SigVer	RSA SigVer for SSHv2, TLSv1.2, and IKEv2 services		RSA SigVer (FIPS186-4): (A4595) Modulus: 2048, 3072 bits
ECDSA SigVer (SSHv2, TLSv1.2, IKEv2)	DigSig-SigVer	ECDSA SigVer for SSHv2, TLSv1.2, and IKEv2 services		ECDSA SigVer (FIPS186-4): (A4595) Curves: P-256, P-384, P-521
SSHv2 Session Encrypt/Decrypt	BC-Auth BC-UnAuth	SSHv2 session protection.	Bit-strength Caveat:Provides between 112 and 152 bits of encryption strength when keys derived by KAS-FFC; Provides between 128 and 256 bits of encryption strength when keys derived by KAS-ECC	AES-CBC: (A4595) Key Length: 128, 256 AES-GCM: (A4595) Key Length: 128, 256
SSHv2 Session Authentication	MAC	SSHv2 Session Authentication.		SHA-1: (A4595) SHA2-256: (A4595) HMAC-SHA-1: (A4595) HMAC-SHA2-256: (A4595)
SSHv2 Keying Materials Development	KAS-135KDF	SSHv2 session keying materials, used to derive SSHv2 session keys.		KDF SSH: (A4595)

Name	Type	Description	Properties	Algorithms
TLSv1.2 Session Encrypt/Decrypt	BC-Auth BC-UnAuth	TLSv1.2 session protection	Bit-strength Caveat:Provides between 112 and 152 bits of encryption strength when keys derived by KAS-FFC; Provides between 128 and 256 bits of encryption strength when keys derived by KAS-ECC	AES-CBC: (A4595) Key Length: 128, 256 AES-GCM: (A4595) Key Length: 128, 256
TLSv1.2 Session Authentication	MAC	TLSv1.2 session authentication.		SHA-1: (A4595) SHA2-256: (A4595) SHA2-384: (A4595) HMAC-SHA-1: (A4595) HMAC-SHA2-256: (A4595) HMAC-SHA2-384: (A4595)
TLSv1.2 Keying Materials Development	KAS-135KDF	TLSv1.2 session keying materials, used to derive TLS session keys.		TLS v1.2 KDF RFC7627: (A4595)
IPsec/IKEv2 Session Encrypt/Decrypt	BC-Auth BC-UnAuth	IPsec/IKEv2 session protection	Bit-strength Caveat:Provides between 112 and 152 bits of encryption strength when keys derived by KAS-FFC; Provides between 128 and 256 bits of encryption strength when keys derived by KAS-ECC	AES-CBC: (A4595) Key Length: 128, 256 AES-GCM: (A4595) Key Length: 128, 256
IPsec/IKEv2 Session Authentication	MAC	IPsec/IKEv2 session authentication.		SHA2-256: (A4595) SHA2-384: (A4595) SHA2-512:

Name	Type	Description	Properties	Algorithms
				(A4595) HMAC-SHA2-256: (A4595) HMAC-SHA2-384: (A4595) HMAC-SHA2-512: (A4595)
IPsec/IKEv2 Keying Materials Development	KAS-135KDF	IPsec/IKEv2 session keying materials, used to derive IPsec/IKEv2 session keys.		KDF IKEv2: (A4595)
SNMPv3 Session Encrypt/Decrypt	BC-UnAuth	SNMPv3 session protection.	Bit-strength Caveat:Provides 128 or 256 bits of encryption strength	AES-CBC: (A4595) Key Length: 128, 256
SNMPv3 Session Authentication	MAC	SNMPv3 session authentication.		SHA-1: (A4595) SHA2-224: (A4595) SHA2-256: (A4595) SHA2-384: (A4595) HMAC-SHA-1: (A4595) HMAC-SHA2-224: (A4595) HMAC-SHA2-256: (A4595) HMAC-SHA2-384: (A4595)
SNMPv3 Keying Materials Development	KAS-135KDF	SNMPv3 session keying materials, used to derive SNMPv3 session keys.		KDF SNMP: (A4595)
DRBG Function	DRBG	Used for DRBG generation		Counter DRBG: (A4595)

Table 9: Security Function Implementations

2.7 Algorithm Specific Information

- 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 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. 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. 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. 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. 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. 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.

- In accordance with FIPS 140-3 IG D.H, the cryptographic module performs Cryptographic Key Generation as per section 5 in SP800-133rev2. The resulting generated seed used in the asymmetric key generation is the unmodified output from SP800-90Arev1 DRBG.

- The module was algorithm tested based on the FIPS 186-4 standard Digital Signatures. According to IG C.K, this module is 186-5 compliant as all 186-4 CAVP tests performed are mathematically identical to the 186-5 CAVP tests. The Module does not support 186-4 DSA or RSA X9.31 for Signature Generation or Signature Verification.

2.8 RBG and Entropy

Cert Number	Vendor Name
E3	Cisco

Table 10: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Cisco Jitter Entropy Source	Non-Physical	Intel Xeon Platinum 8160 (Skylake)	256 bits	Full entropy	A2810 (SHA3-256)

Table 11: Entropy Sources

The module employs a Deterministic Random Bit Generator (DRBG) implementation based on SP800-90Arev1. This DRBG is used internally by the module (e.g. to generate symmetric keys, seeds for asymmetric key pairs, and random numbers for security functions).

The DRBG implemented is an AES-256 Counter DRBG, seeded by the entropy source described in the table above. The Counter DRBG utilizes the Derivation Function and employs prediction resistance.

The DRBG is instantiated with a 384-bits long entropy input (corresponding to 384 bits of entropy). Additionally, the DRBG is reseeded with a 256-bits long entropy input (corresponding to 256 bits of entropy).

2.9 Key Generation

The module implements Cryptographic Key Generation (CKG, vendor affirmed), compliant with SP 800- 133r2. When random values are required, they are obtained from the SP 800-90Ar1 approved DRBG, compliant with Section 4 of SP 800-133r2. The following methods are implemented:

- Direct generation of symmetric keys: compliant with SP 800-133rev2, Section 6.1.
- Safe primes key pair generation: compliant with SP 800-133rev2, Section 5.2, which maps to SP 800-56Arev3. The method described in Section 5.6.1.1.4 of SP 800-56Ar3 (“Testing Candidates”) is used.
- RSA key pair generation: compliant with SP 800-133rev2, Section 5.1, which maps to FIPS 186-4. The method described in Appendix B.3 of FIPS 186-4 (“Probable Primes”) is used.
- ECC (ECDH and ECDSA) key pair generation: compliant with SP 800-133r2, Section 5.1, which maps to FIPS 186-4. The method described in Appendix B.4 of FIPS 186-4 (“Testing Candidates”) is used. Note that this generation method is also used to generate ECDH key pairs.

Additionally, the module implements the following key derivation methods:

- SNMPv3, SSHv2 KDF, TLS 1.2 KDF, IKEv2 KDF: compliant with SP 800-135r1. These implementations shall only be used to generate secret keys in the context of the SNMPv3, SSHv2, TLSv1.2 and IKEv2 KDF protocols, respectively.

Intermediate key generation values are not output from the module and are explicitly zeroized after processing the service

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.
- The module supports the use of the safe primes defined in RFC 4419 (SSH), RFC 7919 (TLS) and RFC 3526 (IKE). Note that the module only implements domain parameter generation, key pair generation and verification, and shared secret computation.
    - SSH (RFC 4419):
        - MODP-2048 (ID = 14)

        - MODP-3072 (ID = 15)
        - MODP-4096 (ID = 16)
    - TLS (RFC 7919):
        - ffdhe2048 (ID = 256)
        - ffdhe3072 (ID = 257)
        - ffdhe4096 (ID = 258)
    - IKE (RFC 3526):
        - MODP-2048 (ID = 14)
        - MODP-3072 (ID = 15)
        - MODP-4096 (ID = 16)

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.

The module also provides the following key transport mechanisms:

- Key wrapping using AES-GCM with a security strength of 128 or 256 bits.
- Key wrapping using AES-CBC with a security strength of 128 or 256 bits with HMAC-SHA-1, HMAC-SHA2-256 or HMAC-SHA2-384.

2.11 Industry Protocols

The module supports SSHv2, TLSv1.2, IPsec/IKEv2 and SNMPv3 industrial protocols. No parts of SSHv2, TLSv1.2, IPsec/IKEv2 or SNMPv3 protocols, other than the KDFs, have been tested by the CAVP and CMVP. 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
N/A	Data Input	Arguments for an API that provide the data to be used for processed by the module.
N/A	Data Output	Arguments output from an API call.
N/A	Control Input	Arguments for an API call used to control and configure module operation.
N/A	Control Output	N/A
N/A	Status Output	Return values, and/or log messages.
N/A	Power	Provide the Power Supply to the module.

Table 12: Ports and Interfaces

The module's physical perimeter encompasses the case of the tested platform mentioned in Table 2. The module provides its logical interfaces via Application Programming Interface (API) calls. The logical interfaces provided by the module are mapped onto the FIPS 140-3 interfaces (data input, data output, control input, control output and status output).

4 Roles, Services, and Authentication

4.1 Authentication Methods

N/A for this module.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Role	Crypto Officer	None

Table 13: Roles

The module supports Crypto Officer (CO) role. The module does not allow concurrent operators. The Crypto Officer is implicitly assumed based on the service requested.

4.3 Approved Services

The following tables detail the types of approved services available to each role in approved mode of operation, the types of access for each role and the Keys or SSPs they affect.

- Generate G
- Read Access R
- Write Access W
- Execute Access E
- Zeroize Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Show Status	Provide Module's current status	None	API command to show status.	Module's current status.	None	Crypto Officer
Show Version	Provide Module's name/ID and versioning information.	None	API command "show version"	Module's name "ASAv Adaptive Security Virtual Appliance" and versioning information	None	Crypto Officer
Perform Self-Tests	Perform Self-Tests (Pre-operational self-tests and	None	API commands to conduct on-demand	Status of the self-tests results.	None	Crypto Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	Conditional Self-Tests)		Self-Tests.			
Perform Zeroization	Perform Zeroization.	None	API commands to conduct Zeroization operation or Power down the tested platform.	Status of the SSPs zeroization.	None	Crypto Officer - DRBG Entropy Input: Z - DRBG Seed: Z - DRBG Internal State V value: Z - DRBG Key: Z - SSH DH Private Key: Z - SSH DH Public Key: Z - 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:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						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 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 RSA Private Key: Z - TLS RSA

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Public Key: Z - TLS ECDSA Private Key: Z - TLS ECDSA Public Key: Z - TLS Master Secret: Z - TLS Session Encryption Key: Z - TLS Session Authentication Key: Z - IPsec/IKEv2 DH Private Key: Z - IPsec/IKEv2 DH Public Key: Z - IPsec/IKEv2 Peer DH Public Key: Z - IPsec/IKEv2 DH Shared Secret: Z - IPsec/IKEv2 ECDH Private Key: Z - IPsec/IKEv2 ECDH Public Key: Z -

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						IPsec/IKEv2 Peer ECDH Public Key: Z - IPsec/IKEv2 ECDH Shared Secret: Z - IPsec/IKEv2 RSA Private Key: Z - IPsec/IKEv2 RSA Public Key: Z - IPsec/IKEv2 ECDSA Private Key: Z - IPsec/IKEv2 ECDSA Public Key: Z - IPsec/IKEv2 Pre-Shared Key: Z - SKEYSEED: Z - IPsec/IKEv2 Session Encryption Key: Z - IPsec/IKEv2 Authentication Key: Z - SNMPv3 Authentication/ Privacy Password: Z - SNMPv3

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Encryption Key: Z - SNMPv3 Authentication Key: Z - RADIUS Secret: Z - TACACS+ Secret: Z
Configure Network	Sets configuration of the systems.	None	API commands to configure the module.	Status of the completion of network related configuration.	None	Crypto Officer
Configure Bypass capability	Sets the Bypass capability	None	API commands to configure the Bypass capability.	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.	API commands to configure SSHv2.	Status of the completion of SSHv2 configuration.	KTS (TLSv1.2 with AES and HMAC) KTS (TLSv1.2 with AES-GCM) KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) RSA KeyGen (SSHv2, TLSv1.2, IKEv2) ECDSA KeyGen (SSHv2, TLSv1.2, IKEv2)	Crypto Officer - SSH RSA Private Key: G,W,E - SSH RSA Public Key: G,R,W - SSH ECDSA Private Key: G,W,E - SSH ECDSA Public Key: G,R,W - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - DRBG Internal

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					DRBG Function	State V value: G,W,E - DRBG Key: G,W,E - RADIUS Secret: W - TACACS+ Secret: W
Configure HTTPS over TLSv1.2 Function	Configure HTTPS over TLSv1.2 Function.	Global Indicator and HTTPS over TLSv1.2 configuration success status message.	API commands to configure HTTPS over TLSv1.2	Status of the completion of HTTPS over TLSv1.2 configuration.	KTS (TLSv1.2 with AES and HMAC) KTS (TLSv1.2 with AES-GCM) KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) RSA KeyGen (SSHv2, TLSv1.2, IKEv2) ECDSA KeyGen (SSHv2, TLSv1.2, IKEv2) DRBG Function	Crypto Officer - TLS RSA Private Key: G,W,E - TLS RSA Public Key: G,R,W - TLS ECDSA Private Key: G,W,E - TLS ECDSA Public Key: G,R,W - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - DRBG Internal State V value: G,W,E - DRBG Key: G,W,E
Configure IPsec/IKEv2 Functions	Configure IPsec/IKEv2 Functions	Global Indicator with IPsec/IKEv2 configuration success status message.	API commands to configure IPsec/IKEv2.	Status of the completion of IPsec/IKEv2 secure tunnel configuration.	KTS (TLSv1.2 with AES and HMAC) KTS (TLSv1.2 with AES-GCM) KTS (SSHv2 with AES	Crypto Officer - IPsec/IKEv2 RSA Private Key: G,W,E - IPsec/IKEv2 RSA Public Key: G,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					and HMAC) KTS (SSHv2 with AES-GCM) RSA KeyGen (SSHv2, TLSv1.2, IKEv2) ECDSA KeyGen (SSHv2, TLSv1.2, IKEv2) DRBG Function	- IPsec/IKEv2 ECDSA Private Key: G,W,E - IPsec/IKEv2 ECDSA Public Key: G,W,E - IPsec/IKEv2 Pre-Shared Key: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - DRBG Internal State V value: G,W,E - DRBG Key: G,W,E
Configure SNMPv3 Function	Configure SNMPv3 Function	Global Indicator and SNMPv3 configuration success status message.	API commands to configure SNMPv3.	Status of the completion of SNMPv3 configuration.	KTS (TLSv1.2 with AES and HMAC) KTS (TLSv1.2 with AES-GCM) KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) SNMPv3 Keying Materials Development	Crypto Officer - SNMPv3 Authentication/ Privacy Password: W,E - SNMPv3 Encryption Key: G,W,E - SNMPv3 Authentication Key: G,W,E
Run SSHv2 Function	Execute SSHv2 Function	Global Indicator and	API commands to	Status of SSHv2 secure	KAS-FFC (SSHv2) KAS-ECC	Crypto Officer - SSH DH

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		Successful SSHv2 log message.	execute SSHv2 service.	tunnel establishment.	(SSHv2) KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) RSA SigGen (SSHv2, TLSv1.2, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2, IKEv2) RSA SigVer (SSHv2, TLSv1.2, IKEv2) ECDSA SigVer (SSHv2, TLSv1.2, IKEv2) SSHv2 Session Encrypt/Decrypt SSHv2 Session Authentication SSHv2 Keying Materials Development DRBG Function	Private Key: G,W,E - SSH DH Public Key: G,R,W - SSH Peer DH Public Key: W,E - SSH DH Shared Secret: G,W,E - SSH ECDH Private Key: G,W,E - SSH ECDH Public Key: G,R,W - SSH Peer ECDH Public Key: W,E - SSH ECDH Shared Secret: G,W,E - SSH RSA Private Key: G,W,E - SSH RSA Public Key: G,R,W - SSH ECDSA Private Key: G,W,E - SSH ECDSA Public Key: G,R,W - SSH Session Encryption Key: G,W,E - SSH Session

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Authentication Key: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - DRBG Internal State V value: G,W,E - DRBG Key: G,W,E - RADIUS Secret: W,E - TACACS+ Secret: W,E
Run HTTPS over TLSv1.2 Function	Execute HTTPS over TLSv1.2 Function.	Global Indicator and Successful HTTPS over TLSv1.2 log message.	API command to execute HTTPS over TLSv1.2 service.	Status of HTTPS over TLSv1.2 establishment.	KAS-FFC (TLSv1.2) KAS-ECC (TLSv1.2) KTS (TLSv1.2 with AES and HMAC) KTS (TLSv1.2 with AES-GCM) RSA SigGen (SSHv2, TLSv1.2, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2, IKEv2) RSA SigVer (SSHv2, TLSv1.2, IKEv2) ECDSA SigVer (SSHv2,	Crypto Officer - TLS DH Private Key: G,W,E - TLS DH Public Key: G,R,W - TLS Peer DH Public Key: W,E - TLS DH Shared Secret: G,W,E - TLS ECDH Private Key: G,W,E - TLS ECDH Public Key: G,R,W - TLS Peer ECDH Public Key: W,E - TLS

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					TLSv1.2, IKEv2) TLSv1.2 Session Encrypt/Decrypt TLSv1.2 Session Authentication TLSv1.2 Keying Materials Development DRBG Function	ECDH Shared Secret: G,W,E - TLS RSA Private Key: G,W,E - TLS RSA Public Key: G,R,W - TLS ECDSA Private Key: G,W,E - TLS ECDSA Public Key: G,R,W - TLS Master Secret: G,W,E - TLS Session Encryption Key: G,W,E - TLS Session Authentication Key: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - DRBG Internal State V value: G,W,E - DRBG Key: G,W,E
Run IPsec/IKEv2 Functions	Execute IPsec/IKEv2 Functions	Global Indicator and Successfu	API command to execute	Status of IPsec/IKEv2 secure tunnel	KAS-FFC (IKEv2) KAS-ECC (IKEv2)	Crypto Officer - IPsec/IKEv2

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		l IPsec/IKEv2 log message.	IPsec/IKEv2	establishment	RSA SigGen (SSHv2, TLSv1.2, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2, IKEv2) RSA SigVer (SSHv2, TLSv1.2, IKEv2) ECDSA SigVer (SSHv2, TLSv1.2, IKEv2) IPsec/IKEv2 Session Encrypt/Decrypt IPsec/IKEv2 Session Authentication IPsec/IKEv2 Keying Materials Development DRBG Function	DH Private Key: G,W,E - IPsec/IKEv2 DH Public Key: G,R,W - IPsec/IKEv2 Peer DH Public Key: W,E - IPsec/IKEv2 DH Shared Secret: G,W,E - IPsec/IKEv2 ECDH Private Key: G,W,E - IPsec/IKEv2 ECDH Public Key: G,R,W - IPsec/IKEv2 Peer ECDH Public Key: W,E - IPsec/IKEv2 ECDH Shared Secret: G,W,E - IPsec/IKEv2 RSA Private Key: G,W,E - IPsec/IKEv2 RSA Public Key: G,W,E - IPsec/IKEv2 ECDSA Private Key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						G,W,E - IPsec/IKEv2 ECDSA Public Key: G,W,E - IPsec/IKEv2 Pre-Shared Key: G,W,E - SKEYSEED: G,W,E - IPsec/IKEv2 Session Encryption Key: G,W,E - IPsec/IKEv2 Authentication Key: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - DRBG Internal State V value: G,W,E - DRBG Key: G,W,E
Run SNMPv3 Functions	Execute SNMPv3 Function.	Global Indicator and Successful SNMPv3 log message.	API command to execute SNMPv3 service.	Status of SNMPv3 service.	SNMPv3 Session Encrypt/Decrypt SNMPv3 Session Authentication SNMPv3 Keying Materials Development	Crypto Officer - SNMPv3 Authentication/ Privacy Password: W,E - SNMPv3 Encryption Key: G,W,E - SNMPv3 Authenticati

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						on Key: G,W,E

Table 14: Approved Services

4.4 Non-Approved Services

N/A for this module.

4.5 External Software/Firmware Loaded

N/A for this module.

4.6 Bypass Actions and Status

The module implements alternating Bypass service. Traffic output from the module's data output interface can be cryptographically protected via IPSec/IKE VPN, or passed as plaintext (Bypass state), depending on the VPN tunnel establishment on the dedicated data output interface. The operator shall assume Crypto Officer role so as to configure IPSec/IKE VPN capability. If no IPSec/IKE VPN was configured, after running two independent internal actions, Module would enter the Bypass state.

Before the module executes the Bypass service (sending out plaintext traffic via the data output interface), the module would conduct two independent internal actions to prevent the inadvertent bypass of plaintext data due to a single error. The Crypto Officer can use commands "show access-list" and "show crypto ipsec sa" to verify the module's Bypass status. In Bypass tests fail, the module would enter an error state, and drop the traffic.

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 operator 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 library is protected by RSA 2048 SigVer with SHA2-512 (RSA and SHA2-512 Cert. #A4595)

signature calculated at build time. At crypto module library initialization, the signature is recalculated and compared to the hardcoded build-time generated signature value. If at load time the signature does not match, the crypto module library exits with error. If failure occurs during self-test, all crypto functionality is disabled.

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 integrity test on-demand.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Non-Modifiable

The module is a firmware hybrid module, which is operated in a non-modifiable operational environment per FIPS 140-3 level 1 specifications. The module's firmware version running on each tested platform is 9.20(3).

The module has control over its own SSPs. The process and memory management functionality of the host device's OS prevent unauthorized access to plaintext private and secret keys, intermediate key generation values and other SSPs by external processes during module execution. The module only allows access to SSPs through its well-defined API. The operational environments provide the capability to separate individual application processes from each other by preventing uncontrolled access to CSPs and uncontrolled modifications of SSPs regardless of whether this data is in the process memory or stored on persistent storage within the operational environment. Processes that are spawned by the module are owned by the module and are not owned by external processes/operators.

7 Physical Security

7.1 Mechanisms and Actions Required

Mechanism	Inspection Frequency	Inspection Guidance
Production grade components	N/A	N/A

Table 15: Mechanisms and Actions Required

The module is running on the multi-chip standalone production grade platform to meet physical security requirements from FIPS 140-3 level 1. The module's Tested Operational Environment's Physical Perimeter (TOEPP) is drawn at the casing of the tested platforms in Table 3. The module's tested platforms consist of production-grade components.

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 provided by the ESXi host for the module temporary.	Dynamic
Flash	Non-Volatile memory provided by the ESXi host for the module to retain memory across power-cycles.	Static

Table 16: Storage Areas

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
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	
Secret Input via SSHv2 encrypted by GCM	External (Outside of the Module's Boundary)	Module	Encrypted	Automated	Electronic	KTS (SSHv2 with AES-GCM)
Public key Output via SSHv2 encrypted by GCM	Module	External (Outside of the Module's Boundary)	Encrypted	Automated	Electronic	KTS (SSHv2 with AES-GCM)
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)
Public key Output via SSHv2 encrypted by AES and HMAC	Module	External (Outside of the Module's Boundary)	Encrypted	Automated	Electronic	KTS (SSHv2 with AES and HMAC)

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

Table 17: 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 and Flash of the module.	`configure factory-default`
Session Termination	Zeroization upon session termination	Session termination will automatically zeroize all session based temporary SSPs	Terminate session
Reboot	Zeroization upon rebooting the module	Reboot to zeroize all temporary SSPs stored in volatile memory	Reboot

Table 18: SSP Zeroization Methods

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

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
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
RADIUS Secret	RADIUS Server Authentication	16 Characters - 128 bits	Authentication Data - CSP			
TACACS+ Secret	TACACS+ Authentication	16 Characters - 128 bits	Authentication Data - CSP			
SSH DH Private Key	Used to derive the SSH DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112 to 152 bits	Private Key - CSP	KAS-FFC (SSHv2)		KAS-FFC (SSHv2)
SSH DH Public Key	Used to derive SSH DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112 to 152 bits	Public Key - PSP		KAS-FFC (SSHv2)	
SSH Peer DH Public Key	Used to derive SSH DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112 to 152 bits	Public Key - PSP			KAS-FFC (SSHv2)
SSH DH Shared Secret	Used to derive SSH Session Encryption Keys, SSH Session	MODP-2048, MODP-3072, MODP-4096 -	Shared Secret - CSP		KAS-FFC (SSHv2)	SSHv2 Keying Materials Development

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	Authentication Keys	112 to 152 bits				
SSH ECDH Private Key	Used to derive the SSH ECDH Shared Secret	Curves: P-256, P-384, P-521 - 128 to 256 bits	Private Key - CSP	KAS-ECC (SSHv2)		KAS-ECC (SSHv2)
SSH ECDH Public Key	Used to derive the SSH ECDH Shared Secret	Curves: P-256, P-384, P-521 - 128 to 256 bits	Public Key - PSP		KAS-ECC (SSHv2)	
SSH Peer ECDH Public Key	Used to derive SSH DH Shared Secret	Curves: P-256, P-384, P-521 - 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: P-256, P-384, P-521 - 128 to 256 bits	Shared Secret - CSP		KAS-ECC (SSHv2)	SSHv2 Keying Materials Development
SSH RSA Private Key	Used for SSH session authentication	Modulus 2048 and 3072 bits - 112 or 128 bits	Private Key - CSP	RSA KeyGen (SSHv2, TLSv1.2, IKEv2)		RSA SigGen (SSHv2, TLSv1.2, IKEv2)
SSH RSA Public Key	Used for SSH session authentication	Modulus 2048 and 3072 bits - 112 or 128 bits	Public Key - PSP		RSA KeyGen (SSHv2, TLSv1.2, IKEv2)	
SSH ECDSA Private Key	Used for SSH session authentication	Curves: P-256, P-384, P-521 - 128 to 256 bits	Private Key - CSP	ECDSA KeyGen (SSHv2, TLSv1.2, IKEv2)		ECDSA SigGen (SSHv2, TLSv1.2, IKEv2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
SSH ECDSA Public Key	Used for SSH session authentication	Curves: P-256, P-384, P-521 - 128 to 256 bits	Public Key - PSP		ECDSA KeyGen (SSHv2, TLSv1.2, IKEv2)	
SSH Session Encryption Key	Used for SSH session confidentiality protection	128, 256 bits - 112 to 256 bits	Symmetric Key - CSP		SSHv2 Keying Materials Development	SSHv2 Session Encrypt/Decrypt
SSH Session Authentication Key	Used for SSH Session integrity protection	At least 160 bits - 112 to 256 bits	Session Key - CSP		SSHv2 Keying Materials Development	SSHv2 Session Authentication
TLS DH Private Key	Used to Derive TLS DH Shared Secret	ffdhe2048, ffdhe3072, ffdhe4096 - 112 to 152 bits	Private Key - CSP	KAS-FFC (TLSv1.2)		KAS-FFC (TLSv1.2)
TLS DH Public Key	Used to Derive TLS DH Shared Secret	ffdhe2048, ffdhe3072, ffdhe4096 - 112 to 152 bits	Public Key - PSP		KAS-FFC (TLSv1.2)	
TLS Peer DH Public Key	Used to derive TLS DH Shared Secret	ffdhe2048, ffdhe3072, ffdhe4096 - 112 to 152 bits	Public Key - PSP			KAS-FFC (TLSv1.2)
TLS DH Shared Secret	Used to Derive TLS Session Encryption Key and TLS Session	ffdhe2048, ffdhe3072, ffdhe4096 - 112	Shared Secret - CSP		KAS-FFC (TLSv1.2)	TLSv1.2 Keying Materials Development

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	Authentication Key	to 152 bits				
TLS ECDH Private Key	Used to Derive TLS ECDH Shared Secret	Curves: P-256, P-384, P-521 - 128 to 256 bits	Private Key - CSP	KAS-ECC (TLSv1.2)		KAS-ECC (TLSv1.2)
TLS ECDH Public Key	Used to Derive TLS ECDH Shared Secret	Curves: P-256, P-384, P-521 - 128 to 256 bits	Public Key - PSP		KAS-ECC (TLSv1.2)	
TLS Peer ECDH Public Key	Used to derive TLS ECDH Shared Secret	Curves: P-256, P-384, P-521 - 128 to 256 bits	Public Key - PSP			KAS-ECC (TLSv1.2)
TLS ECDH Shared Secret	Used to Derive TLS Session Encryption Key and TLS Session Authentication Key	Curves: P-256, P-384, P-521 - 128 to 256 bits	Shared Secret - CSP		KAS-ECC (TLSv1.2)	TLSv1.2 Keying Materials Development
TLS RSA Private Key	Used to support CO HTTPS interfaces	Modulus 2048 and 3072 bits - 112 or 128 bits	Private Key - CSP	RSA KeyGen (SSHv2, TLSv1.2, IKEv2)		RSA SigGen (SSHv2, TLSv1.2, IKEv2)
TLS RSA Public Key	Used to support CO HTTPS interfaces	Modulus 2048 and 3072 bits - 112 or 128 bits	Public Key - PSP		RSA KeyGen (SSHv2, TLSv1.2, IKEv2)	
TLS ECDSA Private Key	Used to support CO HTTPS interfaces	Curves: P-256, P-384, P-521 - 128 to 256 bits	Private Key - CSP	ECDSA KeyGen (SSHv2, TLSv1.2, IKEv2)		ECDSA SigGen (SSHv2, TLSv1.2, IKEv2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
TLS ECDSA Public Key	Used to support CO HTTPS interfaces	Curves: P-256, P-384, P-521 - 128 to 256 bits	Public Key - PSP		ECDSA KeyGen (SSHv2, TLSv1.2, IKEv2)	
TLS Master Secret	Used to protect HTTPS Session	384 bits - 384 bits	Master Secret - CSP		TLSv1.2 Keying Materials Development	TLSv1.2 Session Encrypt/Decrypt TLSv1.2 Session Authentication
TLS Session Encryption Key	Used to protect HTTPS Session	128, 256 bits - 112 to 256 bits	Symmetric Key - CSP		TLSv1.2 Keying Materials Development	TLSv1.2 Session Encrypt/Decrypt
TLS Session Authentication Key	Used to authenticate HTTPS Session	160, 256, 384 bits - 112 to 256 bits	Message Authentication Key - CSP		TLSv1.2 Keying Materials Development	TLSv1.2 Session Authentication
IPsec/IKEv2 DH Private Key	Used to derive IPsec/IKEv2 DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112 to 152 bits	Private Key - CSP	KAS-FFC (IKEv2)		KAS-FFC (IKEv2)
IPsec/IKEv2 DH Public Key	Used to derive IPsec/IKEv2 DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112 to 152 bits	Public Key - PSP		KAS-FFC (IKEv2)	
IPsec/IKEv2 Peer DH Public Key	Used to derive IPsec/IKEv2 DH Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112 to 152 bits	Public Key - PSP			KAS-FFC (IKEv2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
IPsec/IKEv2 DH Shared Secret	Used to derive IPsec/IKEv2 Session Shared Secret	MODP-2048, MODP-3072, MODP-4096 - 112 to 152 bits	Shared Secret - CSP		KAS-FFC (IKEv2)	IPsec/IKEv2 Keying Materials Development
IPsec/IKEv2 ECDH Private Key	Used to derive IPsec/IKEv2 ECDH Shared Secret	Curves: P-256, P-384, P-521 - 128 to 256 bits	Private key - CSP	KAS-ECC (IKEv2)		KAS-ECC (IKEv2)
IPsec/IKEv2 ECDH Public Key	Used to derive IPsec/IKEv2 ECDH Shared Secret	Curves: P-256, P-384, P-521 - 128 to 256 bits	Public Key - PSP		KAS-ECC (IKEv2)	
IPsec/IKEv2 Peer ECDH Public Key	Used to derive IPsec/IKEv2 ECDH Shared Secret	Curves: P-256, P-384, P-521 - 128 to 256 bits	Public Key - PSP			KAS-ECC (IKEv2)
IPsec/IKEv2 ECDH Shared Secret	Used to derive IPsec/IKEv2 ECDH Shared Secret	Curves: P-256, P-384, P-521 - 128 to 256 bits	Shared Secret - CSP		KAS-ECC (IKEv2)	IPsec/IKEv2 Keying Materials Development
IPsec/IKEv2 RSA Private Key	Used for IPsec/IKEv2 authentication	Modulus 2048 and 3072 bits - 112 or 128 bits	Private Key - CSP	RSA KeyGen (SSHv2, TLSv1.2, IKEv2)		RSA SigGen (SSHv2, TLSv1.2, IKEv2)
IPsec/IKEv2 RSA Public Key	Used for IPsec/IKEv2 authentication	Modulus 2048 and 3072 bits - 112 or 128 bits	Public Key - PSP		RSA KeyGen (SSHv2, TLSv1.2, IKEv2)	
IPsec/IKEv2 ECDSA Private Key	Used for IPsec/IKEv2	Curves: P-256, P-384, P-521 -	Private Key - CSP	ECDSA KeyGen (SSHv2,		ECDSA SigGen (SSHv2,

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	authentication	128 to 256 bits		TLSv1.2, IKEv2)		TLSv1.2, IKEv2)
IPsec/IKEv2 ECDSA Public Key	Used for IPsec/IKEv2 authentication	Curves: P-256, P-384, P-521 - 128 to 256 bits	Public Key - PSP		ECDSA KeyGen (SSHv2, TLSv1.2, IKEv2)	
IPsec/IKEv2 Pre-Shared Key	Used for IPsec/IKEv2 authentication	16-32 characters - 128 to 256 bits	Shared Secret - CSP			
SKEYSEED	Keying material used to derive the IPSec/IKE Session Encryption Key and IPSec/IKE Authentication Key	160 bits - 112 to 256 bits	Keying Material - CSP		IPsec/IKEv2 Keying Materials Development	IPsec/IKEv2 Session Encrypt/Decrypt IPsec/IKEv2 Session Authentication
IPsec/IKEv2 Session Encryption Key	Used to secure IPsec/IKEv2 session confidentiality	128, 256 bits - 112 to 256 bits	Symmetric Key - CSP		IPsec/IKEv2 Keying Materials Development	IPsec/IKEv2 Session Encrypt/Decrypt
IPsec/IKEv2 Authentication Key	Used to secure IPsec/IKEv2 session authentication	at least 160 bits - 112 to 256 bits	Message Authentication Key - CSP		IPsec/IKEv2 Keying Materials Development	IPsec/IKEv2 Session Authentication
SNMPv3 Authentication/ Privacy Password	Used for SNMPv3 user authentication	8-32 characters - 64 to 256 bits	Authentication Password - CSP			
SNMPv3 Encryption Key	Used for SNMPv3 confidentiality	128 bits - 128 bits	Symmetric Key - CSP		SNMPv3 Keying Materials Development	SNMPv3 Session Encrypt/Decrypt

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
SNMPv3 Authentication Key	Used for SNMPv3 authentication	At least 112 bits - At least 112 bits	Authentication Key - CSP		SNMPv3 Keying Materials Development	SNMPv3 Session Authentication

Table 19: SSP Table 1

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

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	HMAC Secret Input via TLS encrypted by GCM Secret Input via TLS encrypted by AES and HMAC				
TACACS+ Secret	Secret Input via SSHv2 encrypted by GCM Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via TLS encrypted by GCM Secret Input via TLS encrypted by AES and HMAC	Flash:Encrypted		Zeroization Command	
SSH DH Private Key		DRAM:Plaintext	While SSH session is active	Zeroization Command Session Termination Reboot	SSH DH Public Key:Paired With SSH Peer DH Public Key:Used With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
SSH DH Public Key	Module Public Key Output	DRAM:Plaintext	While SSH session is active	Zeroization Command Session Termination Reboot	SSH DH Private Key:Paired With
SSH Peer DH Public Key	Peer Public Key Input	DRAM:Plaintext	While SSH session is active	Zeroization Command Session Termination Reboot	SSH DH Private Key:Used With
SSH DH Shared Secret		DRAM:Plaintext	While SSH session is active	Zeroization Command Session Termination Reboot	SSH DH Private Key:Derived From SSH Peer DH Public Key:Derived From
SSH ECDH Private Key		DRAM:Plaintext	While SSH session is active	Zeroization Command Session Termination Reboot	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 session is active	Zeroization Command Session Termination Reboot	SSH ECDH Private Key:Paired With
SSH Peer ECDH Public Key	Peer Public Key Input	DRAM:Plaintext	While SSH session is active	Zeroization Command Session Termination Reboot	SSH ECDH Private Key:Used With
SSH ECDH Shared Secret		DRAM:Plaintext	While SSH session is active	Zeroization Command Session Termination Reboot	SSH ECDH Private Key:Derived From SSH Peer ECDH Public Key:Derived From
SSH RSA Private Key		Flash:Plaintext		Zeroization Command	SSH RSA Public Key:Paired With
SSH RSA Public Key	Module Public Key Output	Flash:Plaintext		Zeroization Command	SSH RSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Secret Input via SSHv2 encrypted by GCM Public key Output via SSHv2 encrypted by GCM Secret Input via SSHv2 encrypted by AES and HMAC Public key Output via SSHv2 encrypted by AES and HMAC Secret Input via TLS encrypted by GCM Public key Output via TLS encrypted by GCM Secret Input via TLS encrypted by AES and				

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	HMAC Public key Output via TLS encrypted by AES and HMAC				
SSH ECDSA Private Key		Flash:Plaintext		Zeroization Command	SSH ECDSA Public Key:Paired With
SSH ECDSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypted by GCM Public key Output via SSHv2 encrypted by GCM Secret Input via SSHv2 encrypted by AES and HMAC Public key Output via SSHv2 encrypted by AES and HMAC Secret Input via TLS	Flash:Plaintext		Zeroization Command	SSH ECDSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	encrypted by GCM Public key Output via TLS encrypted by GCM Secret Input via TLS encrypted by AES and HMAC Public key Output via TLS encrypted by AES and HMAC				
SSH Session Encryption Key		DRAM:Plaintext	While SSH session is active	Zeroization Command Session Termination Reboot	SSH Session Authentication Key:Used With
SSH Session Authentication Key		DRAM:Plaintext	While SSH session is active	Zeroization Command Session Termination Reboot	SSH Session Encryption Key:Used With
TLS DH Private Key		DRAM:Plaintext	While TLS session is active	Zeroization Command Session Termination Reboot	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 session is active	Zeroization Command Session Terminatio	TLS DH Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				n Reboot	
TLS Peer DH Public Key	Peer Public Key Input	DRAM:Plaintext	While TLS session is active	Zeroization Command Session Termination Reboot	TLS DH Private Key:Used With
TLS DH Shared Secret		DRAM:Plaintext	While TLS session is active	Zeroization Command Session Termination Reboot	TLS DH Private Key:Derived From TLS Peer DH Public Key:Derived From
TLS ECDH Private Key		DRAM:Plaintext	While TLS session is active	Zeroization Command Session Termination Reboot	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 session is active	Zeroization Command Session Termination Reboot	TLS ECDH Private Key:Paired With
TLS Peer ECDH Public Key	Peer Public Key Input	DRAM:Plaintext	While TLS session is active	Zeroization Command Session Termination Reboot	TLS ECDH Private Key:Used With
TLS ECDH Shared Secret		DRAM:Plaintext	While TLS session is active	Zeroization Command Session Termination Reboot	TLS ECDH Private Key:Derived From TLS Peer ECDH Public Key:Derived From
TLS RSA Private Key		Flash:Plaintext		Zeroization Command	TLS RSA Public Key:Paired With
TLS RSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypte	Flash:Plaintext		Zeroization Command	TLS RSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	d by GCM Public key Output via SSHv2 encrypted by GCM Secret Input via SSHv2 encrypted by AES and HMAC Public key Output via SSHv2 encrypted by AES and HMAC Secret Input via TLS encrypted by GCM Public key Output via TLS encrypted by GCM Secret Input via TLS encrypted by AES and HMAC Public key Output				

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	via TLS encrypted by AES and HMAC				
TLS ECDSA Private Key		Flash:Plaintext		Zeroization Command	TLS ECDSA Public Key:Paired With
TLS ECDSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypted by GCM Public key Output via SSHv2 encrypted by GCM Secret Input via SSHv2 encrypted by AES and HMAC Public key Output via SSHv2 encrypted by AES and HMAC Secret Input via TLS encrypted by GCM Public key	Flash:Plaintext		Zeroization Command	TLS ECDSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Output via TLS encrypted by GCM Secret Input via TLS encrypted by AES and HMAC Public key Output via TLS encrypted by AES and HMAC				
TLS Master Secret		DRAM:Plaintext	While TLS session is active	Zeroization Command Session Termination Reboot	TLS DH Shared Secret:Derived From TLS ECDH Shared Secret:Derived From
TLS Session Encryption Key		DRAM:Plaintext	While TLS session is active	Zeroization Command Session Termination Reboot	TLS Session Authentication Key:Used With TLS Master Secret:Derived From
TLS Session Authentication Key		DRAM:Plaintext	While TLS session is active	Zeroization Command Session Termination Reboot	TLS Session Encryption Key:Used With TLS Master Secret:Derived From
IPsec/IKEv2 DH Private Key		DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	IPsec/IKEv2 DH Public Key:Paired With IPsec/IKEv2 Peer DH Public Key:Used With
IPsec/IKEv2 DH Public Key	Module Public Key Output	DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Terminatio	IPsec/IKEv2 DH Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				n Reboot	
IPsec/IKEv2 Peer DH Public Key	Peer Public Key Input	DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	IPsec/IKEv2 DH Private Key:Used With
IPsec/IKEv2 DH Shared Secret		DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	SKEYSEED:Used With
IPsec/IKEv2 ECDH Private Key		DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	IPsec/IKEv2 ECDH Public Key:Paired With IPsec/IKEv2 Peer ECDH Public Key:Used With
IPsec/IKEv2 ECDH Public Key	Module Public Key Output	DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	IPsec/IKEv2 ECDH Private Key:Paired With
IPsec/IKEv2 Peer ECDH Public Key	Peer Public Key Input	DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	IPsec/IKEv2 ECDH Private Key:Used With
IPsec/IKEv2 ECDH Shared Secret		DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	IPsec/IKEv2 ECDH Private Key:Derived From IPsec/IKEv2 Peer ECDH Public Key:Derived From SKEYSEED:Used With
IPsec/IKEv2 RSA Private Key		Flash:Plaintext		Zeroization Command	IPsec/IKEv2 RSA Public Key:Paired With
IPsec/IKEv2 RSA Public Key	Module Public Key Output Secret	Flash:Plaintext		Zeroization Command	IPsec/IKEv2 RSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Input via SSHv2 encrypted by GCM Public key Output via SSHv2 encrypted by GCM Secret Input via SSHv2 encrypted by AES and HMAC Public key Output via SSHv2 encrypted by AES and HMAC Secret Input via TLS encrypted by GCM Public key Output via TLS encrypted by GCM Secret Input via TLS encrypted by AES and HMAC				

Name	Input - Output	Storage	Storage Duration	Zeroizatio n	Related SSPs
	Public key Output via TLS encrypte d by AES and HMAC				
IPsec/IKEv2 ECDSA Private Key		Flash:Plaintext		Zeroization Command	IPsec/IKEv2 ECDSA Public Key:Paired With
IPsec/IKEv2 ECDSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypte d by GCM Public key Output via SSHv2 encrypte d by GCM Secret Input via SSHv2 encrypte d by AES and HMAC Public key Output via SSHv2 encrypte d by AES and HMAC Secret Input via TLS encrypte	Flash:Plaintext		Zeroization Command	IPsec/IKEv2 ECDSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	d by GCM Public key Output via TLS encrypted by GCM Secret Input via TLS encrypted by AES and HMAC Public key Output via TLS encrypted by AES and HMAC				
IPsec/IKEv2 Pre-Shared Key	Secret Input via SSHv2 encrypted by GCM Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via TLS encrypted by GCM Secret Input via TLS encrypted by AES	Flash:Plaintext		Zeroization Command	SKEYSEED:Derived to

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	and HMAC				
SKEYSEED		DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	IPsec/IKEv2 DH Shared Secret:Derived From IPsec/IKEv2 ECDH Shared Secret:Derived From IPsec/IKEv2 Pre-Shared Secret:Derived From
IPsec/IKEv2 Session Encryption Key		DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	SKEYSEED:Derived From
IPsec/IKEv2 Authentication Key		DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	SKEYSEED:Derived From
SNMPv3 Authentication / Privacy Password	Secret Input via SSHv2 encrypted by GCM Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via TLS encrypted by GCM Secret Input via TLS	Flash:Plaintext		Zeroization Command	SNMPv3 Encryption Key:Derived to SNMPv3 Authentication Key:Derived to

Name	Input - Output	Storage	Storage Duration	Zeroizatio n	Related SSPs
	encrypte d by AES and HMAC				
SNMPv3 Encryption Key		DRAM:Plaintex t	While SNMPv3 session is active	Zeroization Command Session Terminatio n Reboot	SNMPv3 Shared Secret:Derived From SNMPv3 Authentication Key:Used With
SNMPv3 Authentication Key		DRAM:Plaintex t	While SNMPv3 session is active	Zeroization Command Session Terminatio n Reboot	SNMPv3 Shared Secret:Derived From SNMPv3 Encryption Key:Used With

Table 20: SSP Table 2

9.5 Transitions

- SHA-1
  - The module includes an implementation of SHA-1 for hashing and digital signature verification. This implementation will be non-Approved for all uses starting January 1, 2031

- FIPS 186-4/186-5
  - As of February 5, 2024, the CMVP does not accept module submissions that implement DSA or RSA X9.31 in the approved mode, other than for signature verification which is approved for legacy use. This module does not implement DSA or RSA X9.31 for signature generation and therefore is unaffected by the current transition from 186-4 to 186-5. As detailed in section 2.7, the CAVP testing performed on the 186-4 algorithms is mathematically similar to the testing performed on the 186-5 algorithms and therefore this module claims compliance with 186-5. This means that no timeline exists in which any of the implemented algorithms will transition from approved to non-approved.

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
RSA SigVer (FIPS186-4) (A4595)	RSA 2048 SigVer 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 21: Pre-Operational Self-Tests

The module performs the following self-tests, including Pre-operational 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). The self-test success or failure results are an output of the return value of the library load API call, which is functioning as the self-test status indicator. 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 encrypt KAT (A4595)	256 bits	KAT	CAST	Module is in normal state	Encrypt	Power up
AES-CBC decrypt KAT (A4595)	256 bits	KAT	CAST	Module is in normal state	Decrypt	Power up
AES-GCM authenticated encrypt KAT (A4595)	256 bits	KAT	CAST	Module is in normal state	Authenticated Encrypt	Power up
AES-GCM authenticated decrypt KAT (A4595)	256 bits	KAT	CAST	Module is in normal state	Authenticated Decrypt	Power up
Counter DRBG Instantiate/Generate/Reseed KAT (A4595)	AES-128	KAT	CAST	Module is in normal state	Instantiate, Generate, and Reseed KATs	Power up
ECDSA SigGen (FIPS186-4) KAT (A4595)	Curve P-256 with SHA2-256	KAT	CAST	Module is in normal state	ECDSA SigGen KAT	Power up
ECDSA SigVer (FIPS186-4) KAT (A4595)	Curve P-256 with SHA2-256	KAT	CAST	Module is in normal state	ECDSA SigVer KAT	Power up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Entropy Source RCT Start-up Health Tests	Repetition Count Test (RCT)	RCT	CAST	Module is in normal state	N/A	Power up
Entropy Source APT Start-up Health Tests	Adaptive Proportion Test (APT)	APT	CAST	Module is in normal state	N/A	Power up
Entropy Source RCT Continuous Health Tests	Repetition Count Test (RCT)	RCT	CAST	Module is in normal state	N/A	Performed continuously as entropy source is active
Entropy Source APT Continuous Health Tests	Adaptive Proportion Test (APT)	APT	CAST	Module is in normal state	N/A	Performed continuously as entropy source is active
HMAC-SHA-1 KAT (A4595)	SHA-1	KAT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA2-224 KAT (A4595)	SHA2-224	KAT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA2-256 KAT (A4595)	SHA2-256	KAT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA2-384 KAT (A4595)	SHA2-384	KAT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA2-512 KAT (A4595)	SHA2-512	KAT	CAST	Module is in normal state	N/A	Power up
KAS-ECC-SSC Sp800-56Ar3 KAT (A4595)	Curve P-256	KAT	CAST	Module is in normal state	Primitive Z KAT	Power up
KAS-FFC-SSC Sp800-56Ar3 KAT (A4595)	MODP-2048	KAT	CAST	Module is in	Primitive Z KAT	Power up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
				normal state		
KDF IKEv2 KAT (A4595)	N/A	KAT	CAST	Module is in normal state	N/A	Power up
KDF SNMP KAT (A4595)	N/A	KAT	CAST	Module is in normal state	N/A	Power up
KDF SSH KAT (A4595)	N/A	KAT	CAST	Module is in normal state	N/A	Power up
RSA SigGen (FIPS186-4) KAT (A4595)	2048 bit modulus with SHA2-256	KAT	CAST	Module is in normal state	RSA SigGen KAT	Power up
RSA SigVer (FIPS186-4) KAT (A4595)	2048 bit modulus with SHA2-256	KAT	CAST	Module is in normal state	RSA SigVer KAT	Power up
TLS v1.2 KDF RFC7627 KAT (A4595)	N/A	KAT	CAST	Module is in normal state	N/A	Power up
ECDSA KeyGen (FIPS186-4) PCT (A4595)	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.
KAS-ECC-SSC Sp800-56Ar3 PCT (A4595)	Curve P-256 with SHA2-256	PCT	PCT	Module is in normal state	N/A	Performs all required pair-wise consistency tests on

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						the newly generated key pairs before the first operational use.
KAS-FFC-SSC Sp800-56Ar3 PCT (A4595)	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.
RSA KeyGen (FIPS186-4) PCT (A4595)	2048 bit modulus	PCT	PCT	Module is in normal state	RSA	Performs all required pair-wise consistency tests on the newly generated key pairs before the first operational use.
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 22: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigVer (FIPS186-4) (A4595)	KAT	SW/FW Integrity	Recommend 60 Days	Reboot
Pre-Operational Bypass Test	N/A	Bypass	Recommend 60 Days	Reboot

Table 23: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CBC encrypt KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
AES-CBC decrypt KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
AES-GCM authenticated encrypt KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
AES-GCM authenticated decrypt KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
Counter DRBG Instantiate/Generate/Reseed KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
ECDSA SigGen (FIPS186-4) KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
ECDSA SigVer (FIPS186-4) KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
Entropy Source RCT Start-up Health Tests	RCT	CAST	Recommend 60 Days	Reboot
Entropy Source APT Start-up Health Tests	APT	CAST	Recommend 60 Days	Reboot
Entropy Source RCT Continuous Health Tests	RCT	CAST	N/A	N/A
Entropy Source APT Continuous Health Tests	APT	CAST	N/A	N/A
HMAC-SHA-1 KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA2-224 KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA2-256 KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA2-384 KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
HMAC-SHA2-512 KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
KAS-ECC-SSC Sp800-56Ar3 KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
KAS-FFC-SSC Sp800-56Ar3 KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KDF IKEv2 KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
KDF SNMP KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
KDF SSH KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
RSA SigGen (FIPS186-4) KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
RSA SigVer (FIPS186-4) KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
TLS v1.2 KDF RFC7627 KAT (A4595)	KAT	CAST	Recommend 60 Days	Reboot
ECDSA KeyGen (FIPS186-4) PCT (A4595)	PCT	PCT	Recommend 60 Days	Reboot
KAS-ECC-SSC Sp800-56Ar3 PCT (A4595)	PCT	PCT	Recommend 60 Days	Reboot
KAS-FFC-SSC Sp800-56Ar3 PCT (A4595)	PCT	PCT	Recommend 60 Days	Reboot
RSA KeyGen (FIPS186-4) PCT (A4595)	PCT	PCT	Recommend 60 Days	Reboot
Conditional Bypass	N/A	Bypass	Recommend 60 Days	Reboot

Table 24: Conditional Periodic Information

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.

10.4 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 25: 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 integrity test and the conditional CASTs. The module will only enter into the operational state after successfully passing the pre-operational integrity test and the conditional CASTs.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The module meets all the Level 1 requirements for FIPS 140-3. Operating this module without maintaining the following settings will remove the module from the approved mode of operation. Any firmware that is not shown on the module certificate, is out of scope of this validation and requires a separate FIPS 140-3 validation. The Crypto Officer must configure and enforce the following initialization steps:

Step 1: 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 2. Crypto officer shall perform zeroization operation if the module was previously used before the approved mode configuration.

configure factory-default

Step 3: Enable approved mode of operation 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 4: Crypto Officer can verify the version installed and running the following command.

ciscoasa(config)# show version

Step 5: Crypto Officer will need to issue the following commands to configure module.

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

Step 6: Configure IP address for unit and all distant endpoints.

Step 7: Define RADIUS and TACACS+ shared secret keys that are at least 16 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 8: Configure the security module so that any remote connections via Telnet are secured through IPSec connection by using the following commands

crypto map interface
access-list
protocol esp encryption aes
protocol esp integrity sha-256

Note: If the destined IP address is not within access-list, after running two internal independent actions defined in section 4.6 above, the module would enter the Bypass state.

Step 9: Configure the security services by using the algorithms from section 2.5 Approved Algorithms table in this document for all security connections (SSHv2, TLSv1.2, SNMPv3 and IPSec/IKEv2). Note the module will reject any configuration with algorithms not listed in Approved Algorithm Table after the module is operated in approved mode.

Here is an example of configuring the approved algorithms for the security services:

SSH:
ssh cipher encryption custom aes128-gcm@openssh.com
ssh cipher integrity custom hmac-sha2-256
ssh key-exchange group ecdh-sha2-nistp256

TLSv1.2:
ssl cipher tlsv1.2 ecdhe-rsa-aes128-sha

SNMPv3:
snmp-server user <SNMP username> <group name> v3 auth sha <auth password> priv aes 128 <priv password>

IKEv2:
crypto ikev2 policy <policy number>
encryption aes
integrity sha256
group 14

IPsec:
crypto ipsec ikev2 ipsec-proposal <name your proposal>
protocol esp encryption aes
protocol esp integrity sha-256

Step 10: Disable the TFTP server by following the commands:

policy-map global_policy
class inspection_default
no inspect tftp

Step 11: Disable HTTP for performing system management in approved mode of operation by using the command:

no http server enable

HTTPS with TLSv1.2 should always be used for Web-based management.

Step 12: Save the configuration.

write memory

Step 13: Reboot the module.

reload

Step 14: Check the Module's name, version and approved service status by using the following commands:

Output the modules name/version:
show version
Output the modules approved mode of operation status:
show fips

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.