Ruckus Wireless LLC

Ruckus Networks SmartZone (vSZ)

FIPS 140-3 Non-Proprietary Security Policy

Table of Contents

1 General 5
1.1 Overview 5
1.2 Security Levels 5
2 Cryptographic Module Specification 5
2.1 Description 5
2.2 Tested and Vendor Affirmed Module Version and Identification 6
2.3 Excluded Components 7
2.4 Modes of Operation 7
2.5 Algorithms 7
2.6 Security Function Implementations 10
2.7 Algorithm Specific Information 15
2.8 RBG and Entropy 16
2.9 Key Generation 17
2.10 Key Establishment 17
2.11 Industry Protocols 17
3 Cryptographic Module Interfaces 18
3.1 Ports and Interfaces 18
4 Roles, Services, and Authentication 18
4.1 Authentication Methods 18
4.2 Roles 20
4.3 Approved Services 20
4.4 Non-Approved Services 42
4.5 External Software/Firmware Loaded 42
4.6 Bypass Actions and Status 42
4.7 Cryptographic Output Actions and Status 42
4.8 Additional Information 43
5 Software/Firmware Security 43
5.1 Integrity Techniques 43
5.2 Initiate on Demand 43
6 Operational Environment 43
6.1 Operational Environment Type and Requirements 43
7 Physical Security 43
8 Non-Invasive Security 44
9 Sensitive Security Parameters Management 44
9.1 Storage Areas 44

9.2 SSP Input-Output Methods 44
9.3 SSP Zeroization Methods 45
9.4 SSPs 45
9.5 Transitions 65
10 Self-Tests 65
10.1 Pre-Operational Self-Tests 65
10.2 Conditional Self-Tests 65
10.3 Periodic Self-Test Information 70
10.4 Error States 71
10.5 Operator Initiation of Self-Tests 73
11 Life-Cycle Assurance 73
11.1 Installation, Initialization, and Startup Procedures 73
11.2 Administrator Guidance 74
11.3 Non-Administrator Guidance 74
12 Mitigation of Other Attacks 74

List of Tables

Table 1: Security Levels 5
Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets) 6
Table 3: Tested Operational Environments - Software, Firmware, Hybrid 7
Table 4: Modes List and Description 7
Table 5: Approved Algorithms - Ruckus Kernel Crypto Implementation 8
Table 6: Approved Algorithms - Ruckus OpenSSL Crypto Implementation 9
Table 7: Vendor-Affirmed Algorithms 9
Table 8: Security Function Implementations 15
Table 9: Entropy Certificates 16
Table 10: Entropy Sources 17
Table 11: Ports and Interfaces 18
Table 12: Authentication Methods 20
Table 13: Roles 20
Table 14: Approved Services 42
Table 15: Storage Areas 44
Table 16: SSP Input-Output Methods 45
Table 17: SSP Zeroization Methods 45
Table 18: SSP Table 1 52
Table 19: SSP Table 2 65
Table 20: Pre-Operational Self-Tests 65
Table 21: Conditional Self-Tests 70
Table 22: Pre-Operational Periodic Information 70
Table 23: Conditional Periodic Information 71
Table 24: Error States 73

List of Figures

Figure 1: Block Diagram 6

1 General

1.1 Overview

SmartZone™ network controllers deliver unified wired and wireless management of Ruckus Wi-Fi Access Points and Switches. It is designed to meet the scale requirements of large enterprises as well as telecommunications service providers. It can be deployed as a physical or virtual appliance depending on the customer's network architecture. The virtual SmartZone (vSZ) solution has broad hypervisor support and can run on virtual appliances or on popular cloud infrastructure-as-a-service providers.

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	3
5	Software/Firmware security	1
6	Operational environment	1
7	Physical security	N/A
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 is Ruckus Wireless LLC. non-proprietary security policy for Ruckus Networks SmartZone (vSZ) (hereinafter referred to as vSZ or Module), version 7.1.1.3. The following describes how this module meets the security requirements of FIPS 140-3, SP 800-140 and ISO/IEC 19790 for a Security Level 1 software cryptographic module. The Module, is a Network Functions Virtualization (NFV) based WLAN Controller for customers requiring a carrier-class solution that can be deployed on-premises or hosted in hyperscaler environments. It supports the full range of the WLAN Controller features offered by the industry leading physical controllers, while also enabling the deployment of highly scalable and resilient wireless LANs.

Module Type: Software

Module Embodiment: Multi-Chip Standalone

Cryptographic Boundary:

The Module is defined as a multi-chip standalone software module. Figure 1 below depicts the cryptographic boundary (orange color area) and the physical perimeter defined as the tested platform's hard case enclosure around which everything runs. The cryptographic boundary includes all of the software components of the cryptographic libraries. The physical perimeter is the Tested Operational Environment's Physical Perimeter (TOEPP) on which the Module runs. The Module performs no communication other than with the calling application (the process that invokes the Module services).

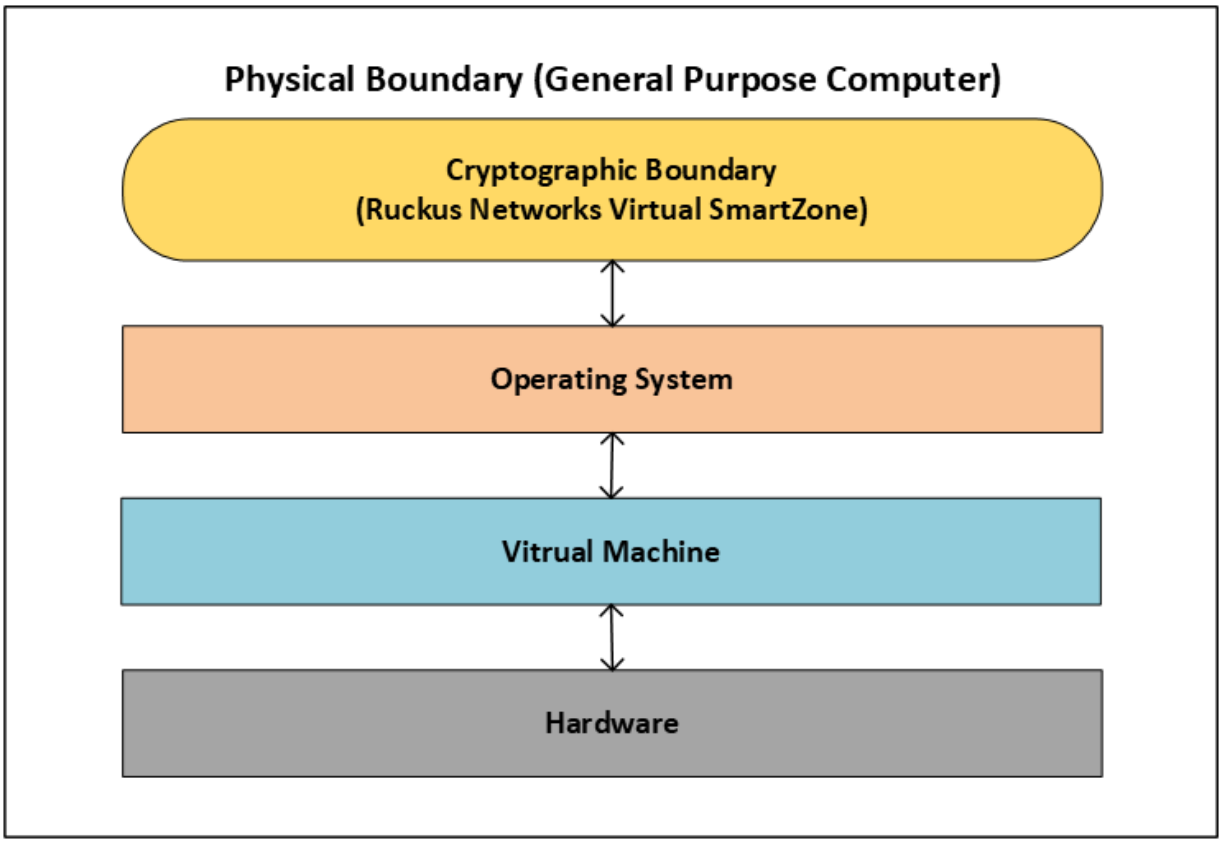

Figure 1: Block Diagram

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
vscg-7.1.1.3	7.1.1.3		RSA 4096 SigVer with SHA2-384

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

Tested Module Identification – Hybrid Disjoint Hardware:

N/A for this module.

Tested Operational Environments - Software, Firmware, Hybrid:

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
SmartZone OS 7.1.1.3	Dell PowerEdge R660xs	Intel Xeon Silver 4410Y	Yes	VMware ESXi 8.0	7.1.1.3
SmartZone OS 7.1.1.3	Dell PowerEdge R660xs	Intel Xeon Silver 4410Y	No	VMware ESXi 8.0	7.1.1.3

Table 3: Tested Operational Environments - Software, Firmware, Hybrid

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 compliance is Enable"

Table 4: Modes List and Description

After the Module is configured in the Approved mode of operation as outlined in Section 11 of this document, it will be ready to operate in the Approved mode. The Module does not claim the implementation of a degraded mode operation.

2.5 Algorithms

Approved Algorithms:

Ruckus Kernel Crypto Implementation

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A6719	Direction - Decrypt, Encrypt Key Length - 128, 192	SP 800-38A
HMAC-SHA-1	A6719	Key Length - Key Length: 160	FIPS 198-1
HMAC-SHA2-256	A6719	Key Length - Key Length: 256	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-384	A6719	Key Length - Key Length: 384	FIPS 198-1
HMAC-SHA2-512	A6719	Key Length - Key Length: 512	FIPS 198-1
SHA-1	A6719	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A6719	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A6719	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A6719	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 5: Approved Algorithms - Ruckus Kernel Crypto Implementation

Ruckus OpenSSL Crypto Implementation

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A6720	Key Length - 128, 256	SP 800-38A
AES-CFB128	A6720	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CTR	A6720	Key Length - 128, 256	SP 800-38A
AES-ECB	A6720	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A6720	Key Length - 128, 256	SP 800-38D
Counter DRBG	A6720	Prediction Resistance - No, Yes Mode - AES-128, AES-192, AES-256 Derivation Function Enabled - No, Yes	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-5)	A6720	Curve - P-256, P-384	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A6720	Curve - P-256, P-384	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A6720	Curve - P-256, P-384	FIPS 186-5
HMAC-SHA-1	A6720	Key Length - Key Length: 160	FIPS 198-1
HMAC-SHA2-256	A6720	Key Length - Key Length: 256	FIPS 198-1
HMAC-SHA2-384	A6720	Key Length - Key Length: 384	FIPS 198-1
HMAC-SHA2-512	A6720	Key Length - Key Length: 512	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A6720	Domain Parameter Generation Methods - P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDF IKEv2 (CVL)	A6720	Diffie-Hellman Shared Secret Length - Diffie-Hellman Shared Secret Length: 2048-3072 Increment 8 Derived Keying Material Length - Derived	SP 800-135 Rev. 1

Algorithm	CAVP Cert	Properties	Reference
		Keying Material Length: 1056-3072 Increment 8 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	
KDF SNMP (CVL)	A6720	Password Length - Password Length: 64-128 Increment 8	SP 800-135 Rev. 1
KDF SSH (CVL)	A6720	Cipher - AES-128, AES-256 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
RSA KeyGen (FIPS186-5)	A6720	Key Generation Mode - probable Modulo - 3072 Primality Tests - 2pow100, 2powSecStr	FIPS 186-5
RSA SigGen (FIPS186-5)	A6720	Modulo - 3072 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-5)	A6720	Modulo - 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
SHA-1	A6720	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A6720	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A6720	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A6720	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
TLS v1.2 KDF RFC7627 (CVL)	A6720	Hash Algorithm - SHA2-256, SHA2-384	SP 800-135 Rev. 1
TLS v1.3 KDF (CVL)	A6720	HMAC Algorithm - SHA2-256, SHA2-384 KDF Running Modes - DHE	SP 800-135 Rev. 1

Table 6: Approved Algorithms - Ruckus OpenSSL Crypto Implementation

As the Module can only be operated in the Approved mode of operation, and any algorithms not listed in the tables above will be rejected by the Module while in the Approved mode, the options defined in SP 800-140B for the following categories are missing from this document.

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Key Type:Asymmetric	N/A	The cryptographic module performs Cryptographic Key Generation (CKG) for asymmetric keys as per section 4 example 1 in SP800-133rev2 (vendor affirmed) and FIPS 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 (A6720)

Table 7: 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 (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 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: (A6720) Curves: P-256, P-384, P-521 KDF SSH: (A6720) Counter DRBG: (A6720) 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 192 bits of security 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: (A6720) Curve: P-384 TLS v1.2 KDF RFC7627: (A6720) Counter DRBG: (A6720) CKG: () Key Type: Asymmetric
KAS-ECC (TLSv1.3)	CKG KAS-Full	Full KAS-ECC Key Agreement used for TLSv1.3 service	Caveat:Key establishment methodology provides 192 bits of security	KAS-ECC-SSC Sp800-56Ar3: (A6720) Curve: P-384 TLS v1.3 KDF:

Name	Type	Description	Properties	Algorithms
			IG:IG D.F Scenario 2, Path 2 (Split) Key Confirmation:No Key Derivation:IG 2.4.B SP 800-135rev1 CVL	(A6720) Counter DRBG: (A6720) CKG: () Key Type: Asymmetric
KAS-ECC (IKEv2)	CKG KAS-Full	Full KAS-ECC Key Agreement used for IKEv2 service	Caveat:Key establishment methodology provides 192 bits of security 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: (A6720) Curve: P-384 KDF IKEv2: (A6720) Counter DRBG: (A6720) CKG: () Key Type: Asymmetric
KTS (SSHv2 with AES and HMAC)	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:IG D.G use of any approved symmetric encryption mode and message authentication code	AES-CBC: (A6720) Key Length: 128, 256 AES-CTR: (A6720) Key Length: 128, 256 HMAC-SHA2-256: (A6720) HMAC-SHA2-512: (A6720) SHA2-256: (A6720) SHA2-512: (A6720)
KTS (SSHv2 with AES-GCM)	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:IG D.G use of any approved	AES-GCM: (A6720) Key Length: 128, 256

Name	Type	Description	Properties	Algorithms
			authenticated symmetric encryption mode	
KTS (TLSv1.2 with AES-GCM)	KTS-Unwrap	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:IG D.G use of any approved authenticated symmetric encryption mode	AES-GCM: (A6720) Key Length: 128, 256
KTS (TLSv1.3 with AES-GCM)	KTS-Unwrap	KTS via TLSv1.3 service by using AES-GCM	Caveat:Key establishment methodology provides 128 or 256 bits of security strength Standard:SP 800-38F IG:IG D.G use of any approved authenticated symmetric encryption mode	AES-GCM: (A6720) Key Length: 128, 256
RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)	AsymKeyPair-KeyGen CKG	RSA KeyGen for SSHv2, TLSv1.2, TLSv1.3, and IKEv2 services		RSA KeyGen (FIPS186-5): (A6720) Modulus: 3072 bits Counter DRBG: (A6720) CKG: () Key Type: Asymmetric
ECDSA KeyGen (SSHv2, TLSv1.2, TLSv1.3)	AsymKeyPair-KeyGen CKG	ECDSA KeyGen for SSHv2, TLSv1.2, and TLSv1.3 services		ECDSA KeyGen (FIPS186-5): (A6720) Curves: P-256, P-384 Counter DRBG: (A6720) CKG: () Key Type: Asymmetric

Name	Type	Description	Properties	Algorithms
ECDSA KeyGen (IKEv2)	AsymKeyPair-KeyGen CKG	ECDSA KeyGen for IKEv2 service		ECDSA KeyGen (FIPS186-5): (A6720) Curve: P-384 Counter DRBG: (A6720) CKG: () Key Type: Asymmetric
RSA SigGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)	DigSig-SigGen	RSA SigGen for SSHv2, TLSv1.2, TLSv1.3, and IKEv2 services		RSA SigGen (FIPS186-5): (A6720) Modulus: 3072 bits
ECDSA SigGen (SSHv2, TLSv1.2, TLSv1.3)	DigSig-SigGen	ECDSA SigGen for SSHv2, TLSv1.2, and TLSv1.3 services		ECDSA SigGen (FIPS186-5): (A6720) Curves: P-256, P-384
ECDSA SigGen (IKEv2)	DigSig-SigGen	ECDSA SigGen for IKEv2 service		ECDSA SigGen (FIPS186-5): (A6720) Curve: P-384
RSA SigVer (SSHv2, TLSv1.2, TLSv1.3, IKEv2)	DigSig-SigVer	RSA SigVer for SSHv2, TLSv1.2, TLSv1.3, and IKEv2 services		RSA SigVer (FIPS186-5): (A6720) Modulus: 3072 bits
ECDSA SigVer (SSHv2, TLSv1.2, TLSv1.3)	DigSig-SigVer	ECDSA SigVer for SSHv2, TLSv1.2, and TLSv1.3 services		ECDSA SigVer (FIPS186-5): (A6720) Curves: P-256, P-384
ECDSA SigVer (IKEv2)	DigSig-SigVer	ECDSA SigVer for IKEv2 services		ECDSA SigVer (FIPS186-5): (A6720) Curve: P-384
SSHv2 Session Encrypt/Decrypt	BC-Auth BC-UnAuth	SSHv2 session protection.		AES-CTR: (A6720) Key Length: 128, 256 AES-GCM: (A6720) Key Length: 128, 256
SSHv2 Session Authentication	MAC	SSHv2 Session Authentication.		HMAC-SHA2-256: (A6720) HMAC-SHA2-512: (A6720)

Name	Type	Description	Properties	Algorithms
				SHA2-256: (A6720) SHA2-512: (A6720)
SSHv2 Keying Materials Development	KAS-135KDF	SSHv2 session keying materials, used to derive SSHv2 session keys.		KDF SSH: (A6720)
TLSv1.2 Session Encrypt/Decrypt	BC-Auth	TLSv1.2 session protection & authentication		AES-GCM: (A6720) Key Length: 128, 256
TLSv1.2 Keying Materials Development	KAS-135KDF	TLSv1.2 session keying materials, used to derive TLS session keys		TLS v1.2 KDF RFC7627: (A6720)
TLSv1.3 Session Encrypt/Decrypt	BC-Auth	TLSv1.3 session protection & authentication		AES-GCM: (A6720) Key Length: 128, 256
TLSv1.3 Keying Materials Development	KAS-135KDF	TLSv1.3 session keying materials, used to derive TLS session keys		TLS v1.3 KDF: (A6720)
IPsec/IKEv2 Session Encrypt/Decrypt	BC-Auth BC-UnAuth	IPsec/IKEv2 session protection		AES-CBC: (A6719, A6720) Key Length: 128, 192, 256 AES-GCM: (A6720) Key Length: 128, 256 bits
IPsec/IKEv2 Session Authentication	MAC	IPsec/IKEv2 session authentication		HMAC-SHA-1: (A6719, A6720) HMAC-SHA2-256: (A6719, A6720) HMAC-SHA2-384: (A6719, A6720) HMAC-SHA2-512: (A6719, A6720) SHA-1: (A6719, A6720) SHA2-256:

Name	Type	Description	Properties	Algorithms
				(A6719, A6720) SHA2-384: (A6719, A6720) SHA2-512: (A6719, A6720)
IPsec/IKEv2 Keying Materials Development	KAS-135KDF	IPsec/IKEv2 session keying materials, used to derive IPsec/IKEv2 session keys		KDF IKEv2: (A6720)
SNMPv3 Session Encrypt/Decrypt	BC-UnAuth	SNMPv3 session authentication		AES-CFB128: (A6720) Key Length: 128
SNMPv3 Session Authentication	MAC	SNMPv3 session authentication		HMAC-SHA-1: (A6720) SHA-1: (A6720)
SNMPv3 Keying Materials Development	KAS-135KDF	SNMPv3 session keying materials, used to derive SNMPv3 session keys		KDF SNMP: (A6720)
Software Load Test	DigSig-SigVer	Digital signature verification for software load test		RSA SigVer (FIPS186-5): (A6720) Modulus: 4096 bits SHA2-384: (A6720)
DRBG Function	DRBG	Used for DRBG generation		Counter DRBG: (A6720)

Table 8: Security Function Implementations

2.7 Algorithm Specific Information

- Algorithm Cert. #A6719 & #A6720 was tested for the OE with PAA and non-PAA.
- 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 section 2.5 Algorithms.
- For SSHv2, the Module's AES-GCM implementation conforms to Implementation Guidance C.H scenario #1 of FIPS 140-3 IG C.H.
- For TLSv1.2, 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.

- For TLSv1.3, the Module offers the AES-GCM implementation and uses the context of Scenario #1 of FIPS 140-3 IG C.H. The protocol that provides this compliance is TLS 1.3, defined in RFC8446 of August 2018, using the cipher suites that explicitly select AES-GCM as the encryption/decryption cipher (Appendix B.4 of RFC8446). The Module supports acceptable AES-GCM cipher suites from Section 3.3.1 of SP800-52 Rev2. The Module implements, within its boundary, an IV generation unit for TLS 1.3 that keeps control of the 64-bit counter value within the AES-GCM IV. If the exhaustion condition is observed, the Module will return an error indication to the calling application, who will then need to either trigger a re-key of the session (i.e., a new key for AES-GCM), or terminate the connection

- 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 SP800-133 Rev2. The resulting generated seed used in the asymmetric key generation is the unmodified output from SP800-90A Rev1 DRBG.

2.8 RBG and Entropy

Cert Number	Vendor Name
E233	Fortanix, Inc.

Table 9: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Fortanix DRNG RDSEED Entropy	Physical	Intel Xeon Silver 4410Y	128 bits	Full Entropy	AES-CBC-MAC (A5245)

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Source (Sapphire Rapids-MCC)					

Table 10: Entropy Sources

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:

- RSA key pair generation: compliant with SP 800-133rev2, Section 5.1, which maps to FIPS 186-5. The method described in Appendix A.1.3 of FIPS 186-5 (“Probable Primes”) is used.
- ECC (ECDSA/ECDH) key pair generation: compliant with SP800-133r2, Section 5.1/5.2, which maps to FIPS 186-5. The method described in Appendix A.2.2 of FIPS 186-5 (“Testing Candidates”) is used.

Additionally, the module implements the following key derivation methods:

- SSHv2 KDF, TLS 1.2 KDF (RFC 7627), TLSv1.3 KDF, IKEv2 KDF and SNMPv3 KDF: compliant with SP 800-135r1. These implementations shall only be used to generate secret keys in the context of the SSHv2, TLSv1.2, TLSv1.3, IKEv2 and SNMPv3 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-ECC Shared Secret Computation:

- The Module provides SP800-56A Rev3 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 (RFC 7627), TLS v1.3, IPsec/IKEv2 and SNMPv3 industrial protocols. No parts of IPSec/IKEv2, SNMPv3, SSH and TLS 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	Arguments for an API call used to control and configure a connected Ruckus Access Point and Data Plane.
N/A	Status Output	Return values, and/or log messages.

Table 11: 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) as follows.

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 fifteen (15) characters (94 possible characters).	Password Based	The probability that a random attempt will succeed or a false acceptance will occur is 1/(94^15) which is less than 1/1,000,000.	The probability of successfully authenticating to the module within one minute is 10/(94^15), which is less than 1/100,000.
RSA-Based Certificate	The Module supports RSA public-key based authentication mechanism using a minimum of RSA 3072 bits, which provides 128 bits of security strength. The probability that a random attempt will	RSA SigVer (FIPS186-5) (A6720)	The probability that a random attempt will succeed is 1/(2^128). 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

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
	succeed is 1/(2^128) 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/(2^128), which is less than 1/100,000.		table for more details	this table for more details
ECDSA-Based Certificate	The Module supports 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 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.	ECDSA SigVer (FIPS186-5) (A6720)	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 table for more details.	The probability of successfully authenticating to the Module within a one minute period is 17,000 * 60 = 1,020,000/(2^128). Please refer to Description section in this table for more details.

Table 12: Authentication Methods

4.2 Roles

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

Table 13: Roles

The Module supports Crypto Officer (CO) role, User role, and Access Point (AP) role.

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	N/A	API command to show status.	Module's current status.	None	Crypto Officer User Access Point
Show Version	Provide Module's name/ID and versioning information	N/A	Commands `show cpuinfo` and `show fips-version`	Module's name/ID and versioning information	None	Crypto Officer User Access Point
Reboot/Perform Self-Tests	Perform Self-Tests (Pre-operational self-tests and Conditional Self-Tests)	N/A	API commands to conduct on-demand Self-Tests.	Status of the self-tests results.	None	Crypto Officer User

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	after reboot.					
Perform Zeroization	Perform Zeroization.	N/A	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 - Crypto Officer Password: Z - User Password: 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

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Private Key: Z - SSH ECDSA Public Key: Z - SSH Session Encryption Key: Z - SSH Session Authentication Key: Z - TLSv1.2 ECDH Private Key: Z - TLSv1.2 ECDH Public Key: Z - TLSv1.2 Peer ECDH Public Key: Z - TLSv1.2 ECDH Shared Secret: Z - TLSv1.2 RSA Private Key: Z - TLSv1.2 RSA Public Key: Z - TLSv1.2 ECDSA Private Key: Z - TLSv1.2 ECDSA Public Key: Z - TLSv1.2 Master

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Secret: Z - TLSv1.2 Session Encryption Key: Z - TLSv1.3 ECDH Private Key: Z - TLSv1.3 ECDH Public Key: Z - TLSv1.3 Peer ECDH Public Key: Z - TLSv1.3 ECDH Shared Secret: Z - TLSv1.3 RSA Private Key: Z - TLSv1.3 RSA Public Key: Z - TLSv1.3 ECDSA Private Key: Z - TLSv1.3 ECDSA Public Key: Z - TLSv1.3 Master Secret: Z - TLSv1.3 Session Encryption Key: Z - IPsec/IKEv2 ECDH Private

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Key: Z - IPsec/IKEv2 ECDH Public Key: Z - 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 Secret: Z - SKEYSEED: Z - IPsec/IKEv2 Session Encryption

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Key: Z - IPsec/IKEv2 Authentication Key: Z - SNMPv3 Shared Secret: Z - SNMPv3 Encryption Key: Z - SNMPv3 Authentication Key: Z
Crypto Officer Authentication	Crypto Officer Role Authentication	N/A	Crypto Officer Authentication Request	Status of the Crypto Officer Authentication	None	Crypto Officer - Crypto Officer Password: W
User Authentication	User Role Authentcation	N/A	User Authentication Requset	Status of the User Authentication	None	User - User Password: W
Access Point Authentication	Access Point Role Authentcation	N/A	Access Point Authentication Request	Status of the Access Point Authentication	None	Access Point - AP Authentication Certificate: W
Configure Network	Sets configuration of the systems.	N/A	API commands to configure the Module.	Status of the completion of network related configuration.	None	Crypto Officer
Configure SSHv2 Function	Configure SSHv2 Function	Status Mode indicator "FIPS compliance is Enable" and SSHv2	API commands to configure SNMPv3.	API commands to configure SSHv2.	KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) KTS (TLSv1.2	Crypto Officer - SSH RSA Private Key: G,W,E - SSH RSA

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		configuration success status message			with AES-GCM) KTS (TLSv1.3 with AES-GCM) RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2) ECDSA KeyGen (SSHv2, TLSv1.2, TLSv1.3) DRBG Function	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 State V value: G,W,E - DRBG Key: G,W,E
Configure TLSv1.2 Function	Configure TLSv1.2 Function.	Status Mode indicator "FIPS compliance is Enable" and TLSv1.2 configuration success status message.	API commands to configure TLSv1.2	Status of the completion of TLSv1.2 configuration.	KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) KTS (TLSv1.2 with AES-GCM) KTS (TLSv1.3 with AES-GCM) RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)	Crypto Officer - TLSv1.2 RSA Private Key: G,W,E - TLSv1.2 RSA Public Key: G,R,W - TLSv1.2 ECDSA Private Key: G,W,E - TLSv1.2 ECDSA Public Key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					ECDSA KeyGen (SSHv2, TLSv1.2, TLSv1.3) DRBG Function	G,R,W - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - DRBG Internal State V value: G,W,Z - DRBG Key: G,W,E
Configure TLSv1.3 Function	Configure TLSv1.3 Function.	Status Mode indicator "FIPS compliance is Enable" and TLSv1.3 configuration success status message.	API commands to configure TLSv1.3	Status of the completion of TLSv1.3 configuration.	KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) KTS (TLSv1.2 with AES-GCM) KTS (TLSv1.3 with AES-GCM) RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2) ECDSA KeyGen (SSHv2, TLSv1.2, TLSv1.3) DRBG Function	Crypto Officer - TLSv1.3 RSA Private Key: G,W,E - TLSv1.3 RSA Public Key: G,R,W - TLSv1.3 ECDSA Private Key: G,E - TLSv1.3 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

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- DRBG Key: G,W,E
Configure IPsec/IKEv2 Functions	Configure IPsec/IKEv2 Functions	Status Mode indicator "FIPS compliance is Enable" with IPsec/IKEv2 configuration success status message.	API commands to configure IPsec/IKEv2.	Status of the completion of IPsec/IKEv2 secure tunnel configuration.	KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) KTS (TLSv1.2 with AES-GCM) KTS (TLSv1.3 with AES-GCM) RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2) ECDSA KeyGen (IKEv2) DRBG Function	Crypto Officer - IPsec/IKEv2 RSA Private Key: G,W,E - IPsec/IKEv2 RSA Public Key: G,R,W - IPsec/IKEv2 ECDSA Private Key: G,W,E - IPsec/IKEv2 ECDSA Public Key: G,R,W - IPsec/IKEv2 Pre-Shared Secret: W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - DRBG Internal State V value: G,W,E - DRBG

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Key: G,W,E
Configure SNMPv3 Function	Configure SNMPv3 Function	Status Mode indicator "FIPS compliance is Enable" and SNMPv3 configuration success status message.	API commands to configure SNMPv3.	Status of the completion of SNMPv3 configuration.	KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) KTS (TLSv1.2 with AES-GCM) KTS (TLSv1.3 with AES-GCM) SNMPv3 Keying Materials Development	Crypto Officer - SNMPv3 Shared Secret: W,E - SNMPv3 Encryption Key: G,W,E - SNMPv3 Authentication Key: G,W,E
Run SSHv2 Function	Execute SSHv2 Function	Status Mode indicator "FIPS compliance is Enable" and Successful SSHv2 log message.	API commands to execute SSHv2 service.	Status of SSHv2 secure tunnel establishment.	KAS-ECC (SSHv2) KTS (SSHv2 with AES and HMAC) KTS (SSHv2 with AES-GCM) RSA SigGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2, TLSv1.3) RSA SigVer (SSHv2, TLSv1.2, TLSv1.3, IKEv2) ECDSA SigVer	Crypto Officer - 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: W,E - SSH

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					(SSHv2, TLSv1.2, TLSv1.3) SSHv2 Session Encrypt/Decrypt SSHv2 Session Authentication SSHv2 Keying Materials Development DRBG Function	RSA Public Key: R,W - SSH ECDSA Private Key: W,E - SSH ECDSA Public Key: R,W - SSH Session Encryption Key: G,W,E - SSH 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 User - SSH ECDH Private Key: G,W,E - SSH ECDH Public Key: G,R,W - SSH Peer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						ECDH Public Key: W,E - SSH ECDH Shared Secret: G,W,E - SSH RSA Private Key: W,E - SSH RSA Public Key: R,W - SSH ECDSA Private Key: W,E - SSH ECDSA Public Key: R,W - SSH Session Encryption Key: G,W,E - SSH 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

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Access Point - 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: W,E - SSH RSA Public Key: R,W - SSH ECDSA Private Key: W,E - SSH ECDSA Public Key: R,W - SSH Session Encryption Key: G,W,E - SSH Session Authentication Key: G,W,E - DRBG

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Entropy Input: G,W,E - DRBG Seed: G,W,E - DRBG Internal State V value: G,W,E - DRBG Key: G,W,E
Run TLSv1.2 Function	Execute TLSv1.2 Function.	Status Mode indicator "FIPS compliance is Enable" and Successful TLSv1.2 log message.	API command to execute TLSv1.2 service.	Status of TLSv1.2 establishment.	KAS-ECC (TLSv1.2) KTS (TLSv1.2 with AES-GCM) RSA SigGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2, TLSv1.3) RSA SigVer (SSHv2, TLSv1.2, TLSv1.3, IKEv2) ECDSA SigVer (SSHv2, TLSv1.2, TLSv1.3) TLSv1.2 Session Encrypt/Decrypt TLSv1.2 Keying Materials Developme	Crypto Officer - TLSv1.2 ECDH Private Key: G,W,E - TLSv1.2 ECDH Public Key: G,R,W - TLSv1.2 Peer ECDH Public Key: W,E - TLSv1.2 ECDH Shared Secret: G,E - TLSv1.2 RSA Private Key: W,E - TLSv1.2 RSA Public Key: R,W - TLSv1.2 ECDSA Private Key: W,E - TLSv1.2

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					nt DRBG Function	ECDSA Public Key: R,W - TLSv1.2 Master Secret: G,W,E - TLSv1.2 Session Encryption 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 User - TLSv1.2 ECDH Private Key: G,W,E - TLSv1.2 ECDH Public Key: G,R,W - TLSv1.2 Peer ECDH Public Key: W,E - TLSv1.2 ECDH Shared Secret: G,E - TLSv1.2

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						RSA Private Key: W,E - TLSv1.2 RSA Public Key: R,W - TLSv1.2 ECDSA Private Key: W,E - TLSv1.2 ECDSA Public Key: R,W - TLSv1.2 Master Secret: G,W,E - TLSv1.2 Session Encryption 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 TLSv1.3 Function	Execute TLSv1.3 Function.	Status Mode indicator "FIPS compliance is Enable" and Successful	API command to execute TLSv1.3 service.	Status of TLSv1.3 establishment.	KAS-ECC (TLSv1.3) KTS (TLSv1.3 with AES-GCM) RSA SigGen (SSHv2, TLSv1.2,	Crypto Officer - TLSv1.3 ECDH Private Key: G,W,E - TLSv1.3 ECDH Public

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		TLSv1.3 log message.			TLSv1.3, IKEv2) ECDSA SigGen (SSHv2, TLSv1.2, TLSv1.3) RSA SigVer (SSHv2, TLSv1.2, TLSv1.3, IKEv2) ECDSA SigVer (SSHv2, TLSv1.2, TLSv1.3) TLSv1.3 Session Encrypt/Decrypt TLSv1.3 Keying Materials Development DRBG Function	Key: G,R,W - TLSv1.3 Peer ECDH Public Key: W,E - TLSv1.3 ECDH Shared Secret: G,W,E - TLSv1.3 RSA Private Key: W,E - TLSv1.3 RSA Public Key: R,W - TLSv1.3 ECDSA Private Key: W,E - TLSv1.3 ECDSA Public Key: R,W - TLSv1.3 Master Secret: G,W,E - TLSv1.3 Session Encryption Key: G,W,E - DRBG Entropy Input: G,W,E - DRBG Seed: G,W,E - DRBG Internal State V value: G,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- DRBG Key: G,W,E User - TLSv1.3 ECDH Private Key: G,W,E - TLSv1.3 ECDH Public Key: G,R,W - TLSv1.3 Peer ECDH Public Key: W,E - TLSv1.3 ECDH Shared Secret: G,W,E - TLSv1.3 RSA Private Key: W,E - TLSv1.3 RSA Public Key: R,W - TLSv1.3 ECDSA Private Key: W,E - TLSv1.3 ECDSA Public Key: R,W - TLSv1.3 Master Secret: G,W,E - TLSv1.3 Session Encryption Key: G,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- 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	Status Mode indicator "FIPS compliance is Enable" and Successful IPsec/IKEv2 log message.	API command to execute IPsec/IKEv2	Status of IPsec/IKEv2 secure tunnel establishment	KAS-ECC (IKEv2) RSA SigGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2) ECDSA SigGen (IKEv2) RSA SigVer (SSHv2, TLSv1.2, TLSv1.3, IKEv2) ECDSA SigVer (IKEv2) IPsec/IKEv2 Session Encrypt/Decrypt IPsec/IKEv2 Session Authentication IPsec/IKEv2 Keying Materials Development DRBG Function	Crypto Officer - 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: W,E - IPsec/IKEv2 RSA

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Public Key: R,W - IPsec/IKEv2 ECDSA Private Key: W,E - IPsec/IKEv2 ECDSA Public Key: R,W - IPsec/IKEv2 Pre-Shared Secret: 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 User

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- 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: W,E - IPsec/IKEv2 RSA Public Key: R,W - IPsec/IKEv2 ECDSA Private Key: W,E - IPsec/IKEv2 ECDSA Public Key: R,W - IPsec/IKEv2 Pre-Shared Secret:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						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.	Status Mode indicator "FIPS compliance is Enable" 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 Shared Secret: W,E - SNMPv3 Encryption Key: G,W,E - SNMPv3 Authentication Key: G,W,E User - SNMPv3 Shared Secret:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						W,E - SNMPv3 Encryption Key: G,W,E - SNMPv3 Authentication Key: G,W,E
Software Load Test	Execute the Software Load Test	Status Mode indicator "FIPS compliance is Enable" and successful Software Loading status message	Commands to load new software image	Outcome of the Software Load Test	Software Load Test	Crypto Officer - Software Load Test Key: R

Table 14: Approved Services

4.4 Non-Approved Services

N/A for this module.

4.5 External Software/Firmware Loaded

The Module supports the software load test by using RSA 4096 SigVer with SHA2-384 (Cert. #A6720) for the new validated software to be uploaded into the Module. A Software Load Test Key was preloaded to the Module's binary at the factory and used for software load test. In order to load new software, the Crypto Officer must authenticate to the Module before loading the software. 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.

4.6 Bypass Actions and Status

N/A for this module.

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, where the unauthenticated users can run 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, all firmware components within the physical boundary are protected by RSA 4096 SigVer with SHA2-384 (Cert. #A6720) signature calculated at build time. At 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 module exits with an 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: Modifiable

The Module is a software module, which is operated in a modifiable operational environment per FIPS 140-3 level 1 specifications. The Module's software version running on each tested platform is 7.1.1.3.

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 SSPs 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

N/A for this module.

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 (within TOEPP) provided by the ESXi host for the Module temporary.	Dynamic
Flash (NVRAM)	Non-Volatile memory (within TOEPP) provided by the ESXi host for the Module to retain memory across power-cycles.	Static

Table 15: 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 AES and HMAC	External (Outside of the Module's Boundary)	Module	Encrypted	Automated	Electronic	KTS (SSHv2 with AES and HMAC)
Secret Input via SSHv2 encrypted by GCM	External (Outside of the Module's Boundary)	Module	Encrypted	Automated	Electronic	KTS (SSHv2 with AES-GCM)
Secret Input via TLSv1.2 encrypted by GCM	External (Outside of the Module's Boundary)	Module	Encrypted	Automated	Electronic	KTS (TLSv1.2 with AES-GCM)
Secret Input via TLSv1.3	External (Outside of the	Module	Encrypted	Automated	Electronic	KTS (TLSv1.3

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
encrypted by GCM	Module's Boundary)					with AES-GCM)

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/NVRAM of the Module.	CO issus command `fips disable` from CLI
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 17: 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
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
Crypto Officer Password	Used to authenticate the Crypto Officer	15-30 Characters - 15-30 Characters	Authentication Data - CSP			
User Password	Used to authenticate the User	15-30 Characters - 15-30	Authentication Data - CSP			

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		Characters				
AP Authentication Certificate	Used for initial AP authentication to Module	Modulus 3072 bits - 128 bits	Certificate - CSP			
Software Load Test Key	Used for Software Load Test	4096 bits - 152 bits	Public Key - CSP			Software Load Test
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 ECDH 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 3072 bits - 128 bits	Private Key - CSP	RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)		RSA SigGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)
SSH RSA Public Key	Used for SSH session authentication	Modulus 3072 bits - 128 bits	Public Key - PSP		RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)	

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
SSH ECDSA Private Key	Used for SSH session authentication	Curves: P-256, P-384 - 128 to 192 bits	Private Key - CSP	ECDSA KeyGen (SSHv2, TLSv1.2, TLSv1.3)		ECDSA SigGen (SSHv2, TLSv1.2, TLSv1.3)
SSH ECDSA Public Key	Used for SSH session authentication	Curves: P-256, P-384 - 128 to 192 bits	Public Key - PSP		ECDSA KeyGen (SSHv2, TLSv1.2, TLSv1.3)	
SSH Session Encryption Key	Used for SSH session confidentiality protection	128, 256 bits - 128, 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 - At least 160 bits	Session Key - CSP		SSHv2 Keying Materials Development	SSHv2 Session Authentication
TLSv1.2 ECDH Private Key	Used to Derive TLSv1.2 ECDH Shared Secret	Curve: P-384 - 192 bits	Private Key - CSP	KAS-ECC (TLSv1.2)		KAS-ECC (TLSv1.2)
TLSv1.2 ECDH Public Key	Used to Derive TLSv1.2 ECDH Shared Secret	Curve: P-384 - 192 bits	Public Key - PSP		KAS-ECC (TLSv1.2)	
TLSv1.2 Peer ECDH Public Key	Used to derive TLSv1.2 ECDH Shared Secret	Curve: P-384 - 192 bits	Public Key - PSP			KAS-ECC (TLSv1.2)
TLSv1.2 ECDH Shared Secret	Used to Derive TLSv1.2 Session Encryption Key and TLS Session	Curve: P-384 - 192 bits	Shared Secret - CSP		KAS-ECC (TLSv1.2)	TLSv1.2 Keying Materials Development

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	Authentication Key					
TLSv1.2 RSA Private Key	Generated by the module used for TLSv1.2 authentication	Modulus 3072 bits - 128 bits	Private Key - CSP	RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)		RSA SigGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)
TLSv1.2 RSA Public Key	Generated by the module used for TLSv1.2 authentication	Modulus 3072 bits - 128 bits	Public Key - PSP		RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)	
TLSv1.2 ECDSA Private Key	Generated by the module used for TLSv1.2 authentication	Curves: P-256, P-384 - 128 to 192 bits	Private Key - CSP	ECDSA KeyGen (SSHv2, TLSv1.2, TLSv1.3)		ECDSA SigGen (SSHv2, TLSv1.2, TLSv1.3)
TLSv1.2 ECDSA Public Key	Generated by the module used for TLSv1.2 authentication	Curves: P-256, P-384 - 128 to 192 bits	Public Key - PSP		ECDSA KeyGen (SSHv2, TLSv1.2, TLSv1.3)	
TLSv1.2 Master Secret	A shared secret created during the handshake that forms the basis for the encryption key.	384 bits - 384 bits	Master Secret - CSP		TLSv1.2 Keying Materials Development	TLSv1.2 Session Encrypt/Decrypt
TLSv1.2 Session Encryption Key	A key generated from the master secret and used to encrypt and decrypt the	128, 256 bits - 128, 256 bits	Symmetric Key - CSP		TLSv1.2 Keying Materials Development	TLSv1.2 Session Encrypt/Decrypt

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	actual data sent in the TLS connection.					
TLSv1.3 ECDH Private Key	Used to Derive TLSv1.3 ECDH Shared Secret	Curve: P-384 - 192 bits	Private Key - CSP	KAS-ECC (TLSv1.3)		KAS-ECC (TLSv1.3)
TLSv1.3 ECDH Public Key	Used to Derive TLSv1.3 ECDH Shared Secret	Curve: P-384 - 192 bits	Public Key - PSP		KAS-ECC (TLSv1.3)	
TLSv1.3 Peer ECDH Public Key	Used to derive TLSv1.3 ECDH Shared Secret	Curve: P-384 - 192 bits	Public Key - PSP			KAS-ECC (TLSv1.3)
TLSv1.3 ECDH Shared Secret	Used to Derive TLSv1.3 Session Encryption Key and TLS Session Authentication Key	Curve: P-384 - 192 bits	Shared Secret - CSP		KAS-ECC (TLSv1.3)	TLSv1.3 Keying Materials Development
TLSv1.3 RSA Private Key	Generated by the module used for TLSv1.3 authentication	Modulus 3072 bits - 128 bits	Private Key - CSP	RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)		RSA SigGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)
TLSv1.3 RSA Public Key	Generated by the module used for TLSv1.3 authentication	Modulus 3072 bits - 128 bits	Public Key - PSP		RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)	

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
TLSv1.3 ECDSA Private Key	Generated by the module used for TLSv1.3 authentication	Curves: P-256, P-384 - 128 to 192 bits	Private Key - CSP	ECDSA KeyGen (SSHv2, TLSv1.2, TLSv1.3)		ECDSA SigGen (SSHv2, TLSv1.2, TLSv1.3)
TLSv1.3 ECDSA Public Key	Generated by the module used for TLSv1.3 authentication	Curves: P-256, P-384 - 128 to 192 bits	Public Key - PSP		ECDSA KeyGen (SSHv2, TLSv1.2, TLSv1.3)	
TLSv1.3 Master Secret	A shared secret created during the handshake that forms the basis for the encryption key.	384 bits - 384 bits	Master Secret - CSP		TLSv1.3 Keying Materials Development	TLSv1.3 Session Encrypt/Decrypt
TLSv1.3 Session Encryption Key	A key generated from the master secret and used to encrypt and decrypt the actual data sent in the TLS connection.	128, 256 bits - 128, 256 bits	Symmetric Key - CSP		TLSv1.3 Keying Materials Development	TLSv1.3 Session Encrypt/Decrypt
IPsec/IKEv2 ECDH Private Key	Used to derive IPsec/IKEv2 ECDH Shared Secret	Curve: P-384 - 192 bits	Private key - CSP	KAS-ECC (IKEv2)		KAS-ECC (IKEv2)
IPsec/IKEv2 ECDH Public Key	Used to derive IPsec/IKEv2 ECDH Shared Secret	Curve: P-384 - 192 bits	Public Key - PSP		KAS-ECC (IKEv2)	

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
IPsec/IKEv2 Peer ECDH Public Key	Used to derive IPsec/IKEv2 ECDH Shared Secret	Curve: P-384 - 192 bits	Public Key - PSP			KAS-ECC (IKEv2)
IPsec/IKEv2 ECDH Shared Secret	Used to derive IPsec/IKEv2 ECDH Shared Secret	Curve: P-384 - 192 bits	Shared Secret - CSP		KAS-ECC (IKEv2)	IPsec/IKEv2 Keying Materials Development
IPsec/IKEv2 RSA Private Key	Used for IPsec/IKEv2 authentication	Modulus 3072 bits - 128 bits	Private Key - CSP	RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)		RSA SigGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)
IPsec/IKEv2 RSA Public Key	Used for IPsec/IKEv2 authentication	Modulus 3072 bits - 128 bits	Public Key - PSP		RSA KeyGen (SSHv2, TLSv1.2, TLSv1.3, IKEv2)	
IPsec/IKEv2 ECDSA Private Key	Used for IPsec/IKEv2 authentication	Curve: P-384 - 192 bits	Private Key - CSP	ECDSA KeyGen (IKEv2)		ECDSA SigGen (IKEv2)
IPsec/IKEv2 ECDSA Public Key	Used for IPsec/IKEv2 authentication	Curve: P-384 - 192 bits	Public Key - PSP		ECDSA KeyGen (IKEv2)	
IPsec/IKEv2 Pre-Shared Secret	Used for IPsec/IKEv2 authentication	1-128 characters - 1-128 characters	Shared Secret - CSP			
SKEYSEED	Keying material used to derive the IPSec/IKE Session Encryption Key and	160 bits - 160 bits	Keying Material - CSP		IPsec/IKEv2 Keying Materials Development	IPsec/IKEv2 Session Encrypt/Decrypt IPsec/IKEv2 Session Authentication

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	IPSec/IKE Authentication Key					
IPsec/IKEv2 Session Encryption Key	Used to secure IPsec/IKEv2 session confidentiality	128, 192, 256 bits - 128, 192, 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 - at least 160 bits	Message Authentication Key - CSP		IPsec/IKEv2 Keying Materials Development	IPsec/IKEv2 Session Authentication
SNMPv3 Shared Secret	Used for SNMPv3 user authentication	8-32 characters - N/A	Authentication on Secret - CSP			
SNMPv3 Encryption Key	Used for SNMPv3 confidentiality	128 bits - 128 bits	Symmetric Key - CSP		SNMPv3 Keying Materials Development	SNMPv3 Session Encrypt/Decrypt
SNMPv3 Authentication Key	Used for SNMPv3 authentication	At least 160 bits - At least 160 bits	Authentication key - CSP		SNMPv3 Keying Materials Development	SNMPv3 Session Authentication

Table 18: 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

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
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
Crypto Officer Password	Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via SSHv2 encrypted by GCM Secret Input via TLSv1.2 encrypted by GCM Secret Input via TLSv1.3 encrypted by GCM	Flash (NVRAM):Encrypted		Zeroization Command	
User Password	Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via SSHv2	Flash (NVRAM):Encrypted		Zeroization Command	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	encrypted by GCM Secret Input via TLSv1.2 encrypted by GCM Secret Input via TLSv1.3 encrypted by GCM				
AP Authentication Certificate	Secret Input via TLSv1.2 encrypted by GCM Secret Input via TLSv1.3 encrypted by GCM	DRAM:Plaintext	While AP is connected to Module	Zeroization Command Session Termination Reboot	
Software Load Test Key		Flash (NVRAM):Plaintext		N/A	
SSH ECDH Private Key		DRAM:Plaintext	While SSHv2 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 SSHv2 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 SSHv2 session is active	Zeroization Command Session	SSH ECDH Private Key:Used With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				Termination Reboot	
SSH ECDH Shared Secret		DRAM:Plaintext	While SSHv2 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 (NVRAM):Plaintext		Zeroization Command	SSH RSA Public Key:Paired With
SSH RSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via SSHv2 encrypted by GCM Secret Input via TLSv1.2 encrypted by GCM Secret Input via TLSv1.3 encrypted by GCM	Flash (NVRAM):Plaintext		Zeroization Command	SSH RSA Private Key:Paired With
SSH ECDSA Private Key		Flash (NVRAM):Plaintext		Zeroization Command	SSH ECDSA Public Key:Paired With
SSH ECDSA Public Key	Module Public Key	Flash (NVRAM):Plaintext		Zeroization Command	SSH ECDSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroizatio n	Related SSPs
	Output Secret Input via SSHv2 encrypte d by AES and HMAC Secret Input via SSHv2 encrypte d by GCM Secret Input via TLSv1.2 encrypte d by GCM Secret Input via TLSv1.3 encrypte d by GCM				
SSH Session Encryption Key		DRAM:Plaintext	While SSHv2 session is active	Zeroizatio n Command Session Terminatio n Reboot	SSH Session Authentication Key:Used With
SSH Session Authenticatio n Key		DRAM:Plaintext	While SSHv2 session is active	Zeroizatio n Command Session Terminatio n Reboot	SSH Session Encryption Key:Used With
TLSv1.2 ECDH Private Key		DRAM:Plaintext	While TLSv1.2 session is active	Zeroizatio n Command Session Terminatio n Reboot	TLSv1.2 ECDH Public Key:Paired With TLSv1.2 Peer ECDH Public Key:Used With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
TLSv1.2 ECDH Public Key	Module Public Key Output	DRAM:Plaintext	While TLSv1.2 session is active	Zeroization Command Session Termination Reboot	TLSv1.2 ECDH Private Key:Paired With
TLSv1.2 Peer ECDH Public Key	Peer Public Key Input	DRAM:Plaintext	While TLSv1.2 session is active	Zeroization Command Session Termination Reboot	TLSv1.2 ECDH Private Key:Used With
TLSv1.2 ECDH Shared Secret		DRAM:Plaintext	While TLSv1.2 session is active	Zeroization Command Session Termination Reboot	TLSv1.2 ECDH Private Key:Derived From TLSv1.2 Peer ECDH Public Key:Derived From
TLSv1.2 RSA Private Key		Flash (NVRAM):Plaintext		Zeroization Command	TLSv1.2 RSA Public Key:Paired With
TLSv1.2 RSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via SSHv2 encrypted by GCM Secret Input via TLSv1.2 encrypted by GCM Secret	Flash (NVRAM):Plaintext		Zeroization Command	TLSv1.2 RSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroizatio n	Related SSPs
	Input via TLSv1.3 encrypte d by GCM				
TLSv1.2 ECDSA Private Key		Flash (NVRAM):Plaintex t		Zeroizatio n Command	TLSv1.2 ECDSA Public Key:Paired With
TLSv1.2 ECDSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypte d by AES and HMAC Secret Input via SSHv2 encrypte d by GCM Secret Input via TLSv1.2 encrypte d by GCM Secret Input via TLSv1.3 encrypte d by GCM	Flash (NVRAM):Plaintex t		Zeroizatio n Command	TLSv1.2 ECDSA Private Key:Paired With
TLSv1.2 Master Secret		DRAM:Plaintext	While TLSv1.2 session is active	Zeroizatio n Command Session Terminatio n Reboot	TLSv1.2 ECDH Shared Secret:Derived From
TLSv1.2 Session Encryption Key		DRAM:Plaintext	While TLSv1.2 session is active	Zeroizatio n Command Session	TLSv1.2 Master Secret:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				Termination Reboot	
TLSv1.3 ECDH Private Key		DRAM:Plaintext	While TLSv1.3 session is active	Zeroization Command Session Termination Reboot	TLSv1.3 ECDH Public Key:Paired With TLSv1.3 Peer ECDH Public Key:Used With
TLSv1.3 ECDH Public Key	Module Public Key Output	DRAM:Plaintext	While TLSv1.3 session is active	Zeroization Command Session Termination Reboot	TLSv1.3 ECDH Private Key:Paired With
TLSv1.3 Peer ECDH Public Key	Peer Public Key Input	DRAM:Plaintext	While TLSv1.3 session is active	Zeroization Command Session Termination Reboot	TLSv1.3 ECDH Private Key:Used With
TLSv1.3 ECDH Shared Secret		DRAM:Plaintext	While TLSv1.3 session is active	Zeroization Command Session Termination Reboot	TLSv1.3 ECDH Private Key:Derived From TLSv1.3 Peer ECDH Public Key:Derived From
TLSv1.3 RSA Private Key		Flash (NVRAM):Plaintext		Zeroization Command	TLSv1.3 RSA Public Key:Paired With
TLSv1.3 RSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via SSHv2	Flash (NVRAM):Plaintext		Zeroization Command	TLSv1.3 RSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	encrypted by GCM Secret Input via TLSv1.2 encrypted by GCM Secret Input via TLSv1.3 encrypted by GCM				
TLSv1.3 ECDSA Private Key		Flash (NVRAM):Plaintext		Zeroization Command	TLSv1.3 ECDSA Public Key:Paired With
TLSv1.3 ECDSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via SSHv2 encrypted by GCM Secret Input via TLSv1.2 encrypted by GCM Secret Input via TLSv1.3 encrypted by GCM	Flash (NVRAM):Plaintext		Zeroization Command	TLSv1.3 ECDSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
TLSv1.3 Master Secret		DRAM:Plaintext	While TLSv1.3 session is active	Zeroization Command Session Termination Reboot	TLSv1.3 ECDH Shared Secret:Derived From
TLSv1.3 Session Encryption Key		DRAM:Plaintext	While TLSv1.3 session is active	Zeroization Command Session Termination Reboot	TLSv1.3 Master Secret:Derived From
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 (NVRAM):Plaintext		Zeroization Command	IPsec/IKEv2 RSA Public Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
IPsec/IKEv2 RSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via SSHv2 encrypted by GCM Secret Input via TLSv1.2 encrypted by GCM Secret Input via TLSv1.3 encrypted by GCM	Flash (NVRAM):Plaintext		Zeroization Command	IPsec/IKEv2 RSA Private Key:Paired With
IPsec/IKEv2 ECDSA Private Key		Flash (NVRAM):Plaintext		Zeroization Command	IPsec/IKEv2 ECDSA Public Key:Paired With
IPsec/IKEv2 ECDSA Public Key	Module Public Key Output Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via SSHv2 encrypted by GCM	Flash (NVRAM):Plaintext		Zeroization Command	IPsec/IKEv2 ECDSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Secret Input via TLSv1.2 encrypted by GCM Secret Input via TLSv1.3 encrypted by GCM				
IPsec/IKEv2 Pre-Shared Secret	Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via SSHv2 encrypted by GCM Secret Input via TLSv1.2 encrypted by GCM Secret Input via TLSv1.3 encrypted by GCM	Flash (NVRAM):Plaintext		Zeroization Command	SKEYSEED:Derived to
SKEYSEED		DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	IPsec/IKEv2 ECDH Shared Secret:Derived From IPsec/IKEv2 Pre-Shared Secret:Derived From
IPsec/IKEv2 Session		DRAM:Plaintext	While IPsec/IKEv	Zeroization Command	SKEYSEED:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Encryption Key			2 tunnel is active	Session Termination Reboot	
IPsec/IKEv2 Authentication Key		DRAM:Plaintext	While IPsec/IKEv2 tunnel is active	Zeroization Command Session Termination Reboot	SKEYSEED:Derived From
SNMPv3 Shared Secret	Secret Input via SSHv2 encrypted by AES and HMAC Secret Input via SSHv2 encrypted by GCM Secret Input via TLSv1.2 encrypted by GCM Secret Input via TLSv1.3 encrypted by GCM	Flash (NVRAM):Plaintext	While SNMPv3 session is active	Zeroization Command Session Termination Reboot	SNMPv3 Encryption Key:Derived to SNMPv3 Authentication Key:Derived to
SNMPv3 Encryption Key		DRAM:Plaintext	While SNMPv3 session is active	Zeroization Command Session Termination Reboot	SNMPv3 Shared Secret:Derived From SNMPv3 Authentication Key:Used With
SNMPv3 Authentication Key		DRAM:Plaintext	While SNMPv3 session is active	Zeroization Command Session Terminatio	SNMPv3 Shared Secret:Derived From SNMPv3

Name	Input - Output	Storage	Storage Duration	Zeroizatio n	Related SSPs
				n Reboot	Encryption Key:Used With

Table 19: SSP Table 2

9.5 Transitions

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

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
RSA SigVer (FIPS186-5) KAT (A6720)	RSA 4096 SigVer with SHA2-384	RSA signature verification	SW/FW Integrity	Module is in normal state	RSA SigVer

Table 20: Pre-Operational Self-Tests

The Module performs the following self-tests, which include the 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 (A6719)	128 bits	KAT	CAST	Module is in normal state	Encrypt	Power up
AES-CBC Decrypt KAT (A6719)	128 bits	KAT	CAST	Module is in normal state	Decrypt	Power up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-ECB Encrypt KAT (A6720)	128 bits	KAT	CAST	Module is in normal state	Encrypt	Power up
AES-ECB Decrypt KAT (A6720)	128 bits	KAT	CAST	Module is in normal state	Decrypt	Power up
AES-GCM Authenticated Encrypt KAT (A6720)	256 bits	KAT	CAST	Module is in normal state	Encrypt	Power up
AES-GCM Authenticated Decrypt KAT (A6720)	256 bits	KAT	CAST	Module is in normal state	Decrypt	Power up
Counter DRBG Instantiate/Generate/Reseed (A6720)	AES-128	KAT	CAST	Module is in normal state	Instantiate, Generate, and Reseed KATs	Power up
ECDSA SigGen (FIPS186-5) KAT (A6720)	Curve P-256 with SHA2-256	KAT	CAST	Module is in normal state	ECDSA	Power up
ECDSA SigVer (FIPS186-5) KAT (A6720)	Curve P-256 with SHA2-256	KAT	CAST	Module is in normal state	ECDSA	Power up
Entropy Source Start-up Health Tests	Runs the CHTs for a probationary period of 65535 bits from the entropy source.	Startup Entropy Source Health Tests	CAST	Module is in normal state	N/A	Power up
Entropy Source Startup Logic Integrity BIST	Performs a logic integrity of the Entropy Source	Startup Logic Integrity BIST	CAST	Module is in normal state	N/A	Power up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Entropy Source Continuous Health Tests (CHT)	Composed of a short-term test yielding a pass/fail over individual 256-bit blocks of noise source data and a long-term evaluation of the pass/fail history of the past 256 block (totaling 65536 bits) to infer an entropy source failure	CHT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA-1 KAT (A6719)	SHA-1	KAT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA-1 KAT (A6720)	SHA-1	KAT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA2-256 KAT (A6719)	SHA2-256	KAT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA2-256 KAT (A6720)	SHA2-256	KAT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA2-384 KAT (A6719)	SHA2-384	KAT	CAST	Module is in normal state	N/A	Power up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-384 KAT (A6720)	SHA2-384	KAT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA2-512 KAT (A6719)	SHA2-512	KAT	CAST	Module is in normal state	N/A	Power up
HMAC-SHA2-512 KAT (A6720)	SHA2-512	KAT	CAST	Module is in normal state	N/A	Power up
KAS-ECC-SSC Sp800-56Ar3 KAT (A6720)	Curve P-256	KAT	CAST	Module is in normal state	Primitive Z KAT	Power up
KDF IKEv2 KAT (A6720)	N/A	KAT	CAST	Module is in normal state	N/A	Power up
KDF SNMP KAT (A6720)	N/A	KAT	CAST	Module is in normal state	N/A	Power up
KDF SSH KAT (A6720)	N/A	KAT	CAST	Module is in normal state	N/A	Power up
RSA SigGen (FIPS186-5) KAT (A6720)	2048 bit modulus with SHA2-256	KAT	CAST	Module is in normal state	RSA	Power up
RSA SigVer (FIPS186-5) KAT (A6720)	2048 bit modulus with SHA2-256	KAT	CAST	Module is in normal state	RSA	Power up
SHA2-384 KAT (A6720)	Output size: 384 bits	KAT	CAST	Module is in normal state	N/A	Power up
TLS v1.2 KDF RFC7627 KAT (A6720)	N/A	KAT	CAST	Module is in normal state	N/A	Power up
TLS v1.3 KDF KAT (A6720)	N/A	KAT	CAST	Module is in	N/A	Power up

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
				normal state		
ECDSA KeyGen (FIPS186-5) PCT (A6720)	N/A	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 (A6720)	N/A	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 (FIPS186-5) PCT (A6720)	N/A	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.
Software Load Test	RSA 4096 SigVer	Signature Verification	SW/FW Load	Module is in	RSA 4096 SigVer	When software has been

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
	with SHA2-384			normal state	with SHA2-384	uploaded to the Module

Table 21: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigVer (FIPS186-5) KAT (A6720)	RSA signature verification	SW/FW Integrity	Recommend 60 Days	Reboot

Table 22: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CBC Encrypt KAT (A6719)	KAT	CAST	Recommended 60 Days	Reboot
AES-CBC Decrypt KAT (A6719)	KAT	CAST	Recommended 60 Days	Reboot
AES-ECB Encrypt KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
AES-ECB Decrypt KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
AES-GCM Authenticated Encrypt KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
AES-GCM Authenticated Decrypt KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
Counter DRBG Instantiate/Generate/Reseed (A6720)	KAT	CAST	Recommended 60 Days	Reboot
ECDSA SigGen (FIPS186-5) KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
ECDSA SigVer (FIPS186-5) KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
Entropy Source Start-up Health Tests	Startup Entropy Source Health Tests	CAST	Recommended 60 Days	Reboot
Entropy Source Startup Logic Integrity BIST	Startup Logic Integrity BIST	CAST	Recommended 60 Days	Reboot
Entropy Source Continuous Health Tests (CHT)	CHT	CAST	Recommended 60 Days	Reboot
HMAC-SHA-1 KAT (A6719)	KAT	CAST	Recommended 60 Days	Reboot

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA-1 KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
HMAC-SHA2-256 KAT (A6719)	KAT	CAST	Recommended 60 Days	Reboot
HMAC-SHA2-256 KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
HMAC-SHA2-384 KAT (A6719)	KAT	CAST	Recommended 60 Days	Reboot
HMAC-SHA2-384 KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
HMAC-SHA2-512 KAT (A6719)	KAT	CAST	Recommended 60 Days	Reboot
HMAC-SHA2-512 KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
KAS-ECC-SSC Sp800-56Ar3 KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
KDF IKEv2 KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
KDF SNMP KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
KDF SSH KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
RSA SigGen (FIPS186-5) KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
RSA SigVer (FIPS186-5) KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
SHA2-384 KAT (A6720)	KAT	CAST	Recommend 60 Days	Reboot
TLS v1.2 KDF RFC7627 KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
TLS v1.3 KDF KAT (A6720)	KAT	CAST	Recommended 60 Days	Reboot
ECDSA KeyGen (FIPS186-5) PCT (A6720)	PCT	PCT	Recommended 60 Days	Reboot
KAS-ECC-SSC Sp800-56Ar3 PCT (A6720)	PCT	PCT	Recommended 60 Days	Reboot
RSA KeyGen (FIPS186-5) PCT (A6720)	PCT	PCT	Recommended 60 Days	Reboot
Software Load Test	Signature Verification	SW/FW Load	N/A	N/A

Table 23: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Hard Error State	If self-tests fail, the Module is put into an error state	Pre-Operational Firmware Integrity Test AES-CBC Encrypt KAT (A6719) AES-CBC Decrypt KAT (A6719) AES-ECB Encrypt KAT (A6720) AES-ECB Decrypt KAT (A6720) AES-GCM Authenticated Encrypt KAT (A6720) AES-GCM Authenticated Decrypt KAT (A6720) Counter DRBG Instantiate/Generate/Reseed (A6720) ECDSA SigGen (FIPS186-5) KAT (A6720) ECDSA SigVer (FIPS186-5) KAT (A6720) HMAC-SHA-1 KAT (A6719) HMAC-SHA-1 KAT (A6720) HMAC-SHA2-256 KAT (A6719) HMAC-SHA2-256 KAT (A6720) HMAC-SHA2-384 KAT (A6719) HMAC-SHA2-384 KAT (A6720) HMAC-SHA2-512 KAT (A6719) HMAC-SHA2-512 KAT (A6720) KAS-ECC-SSC Sp800-56Ar3 KAT (A6720) KDF IKEv2 KAT (A6720) KDF SNMP KAT (A6720) KDF SSH KAT (A6720) RSA SigGen (FIPS186-5) KAT (A6720) RSA SigVer (FIPS186-5) KAT (A6720) SHA2-384 KAT (A6720) TLS v1.2 KDF RFC7627 KAT (A6720) TLS v1.3 KDF KAT (A6720)	Reboot the Module	System enters quarantine state and reboots

Name	Description	Conditions	Recovery Method	Indicator
Soft Error State	If any of the less severe self-tests fail, the module will be put into the soft error state where the failed self-tests are retried until passed and an error indicator provided.	Entropy Source Start-up Health Tests Entropy Source Startup Logic Integrity BIST Entropy Source Continuous Health Tests (CHT) ECDSA KeyGen (FIPS186-5) PCT (A6720) KAS-ECC-SSC Sp800-56Ar3 PCT (A6720) RSA KeyGen (FIPS186-5) PCT (A6720) Software Load Test	The self-test will be retried.	Log message

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 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.

10.5 Operator Initiation of 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.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The Module meets all the Level 1 requirements for FIPS 140-3. The Crypto Officer must configure and enforce the following initialization steps:

- **Deployment**: Deploy the Virtual SmartZone (vSZ) image on a hypervisor by following up the steps specified in section “vSZ Installation with Approved mode Software Image” of “*RUCKUS FIPS and Common Criteria Configuration Guide for SmartZone and AP, 7.1.1.3”*, Published on June 18th, 2025 with the documentation Part Number 800-72735-001 RevA.

- **Controller Configuration with Approved mode Software Image**:
    1. Power on the Module and access the CLI via its virtual console port from hypervisor.

    2. At the login prompt, login with the default administrator username and password (admin/admin). And then, issue ‘enable’ (en) command with the privileged mode password to promote the authorization.

    3. If the system is first time boot up, issue ‘setup’ command and follow the virtual console’s instructions to configure the system fundamental parameters, such as network settings

and the Approved mode of operation. The system will reboot when the mode of operation is being changed. Once the system is reloaded, the CO needs to login to CLI and issue the ‘setup’ command again to configure cluster information and to change the default login and privileged mode passwords. The CO needs to make sure the passwords and all other shared secrets used by the Module must each be at least eight (8) characters long, including at least one alphabet, one numeric character, one special character (note: The special character ` cannot be used in the password and the special characters combination '$(' cannot be used in the password).

4. At the command prompt enter ‘fips?’ to display the list of available commands.

5. Enter ‘fips status’ to verify whether the Approved mode is enabled or disabled. If the Approved mode is enabled, user should be able to observe “FIPS compliance is Enable” from the console. On the other hand, if the Approved mode is disabled, “FIPS compliance is Disable” will be shown on the console.

6. User must issue ‘fips enable’ to enable the Approved mode, then enter “yes” to confirm.

   Note: Although not recommended, the operator can disable the Approved mode by issuing the command `fips disable` and enter “yes” to confirm.

7. Configure RadSec Service (RADIUS over TLS). Please refer to Configuring RadSec section in “*RUCKUS FIPS and Common Criteria Configuration Guide for SmartZone and AP, 7.1.1.3*”, Published on June 18th, 2025 with the documentation Part Number 800-72735-001 RevA.

8. Enter ‘fips showlog’ to display the results of self-tests and verify all are passing.

11.2 Administrator Guidance

RUCKUS SmartZone (LT-GA) Network Management Guide, 7.1.0

- This guide is written for service operators and system administrators who are responsible for managing, configuring, and troubleshooting Ruckus devices. Consequently, it assumes a basic working knowledge of local area networks, wireless networking, and wireless devices. This guide supports all platforms of the controller. It is recommended to view the New In This Document section for a high-level overview.

RUCKUS SmartZone (LT-GA) Controller Administration Guide, 7.1.0

- The *Controller Administration Guide, 7.1.0* provides comprehensive instructions for configuring, managing, and maintaining the controller platform running version 7.1.0. Intended for network administrators and IT professionals, the guide covers key administrative tasks such as initial setup, system configuration, network and security settings, user management, firmware upgrades, backup and restore procedures, monitoring, logging, and troubleshooting. It includes best practices, tips for navigating both the CLI and GUI, and guidance on integrating with external services and management tools. Supporting both new and existing deployments, the guide enables administrators to effectively manage controller performance, security, and scalability in dynamic network environments.

11.3 Non-Administrator Guidance

No specific Non-Administrator guidance.

12 Mitigation of Other Attacks

N/A for this module.