Fortanix SDKMS Appliance

FIPS 140-2 Level 3 Non-Proprietary Security Policy

Date:	9/29/2021
Version:	3.6.20
Document Number:	1.2

This document can be freely distributed in its entirety without modification

Table of Contents

Table of Figures

Table of Figures

This document can be freely distributed in its entirety without modification

Revision History

Author(s)	Version	Date	Updates
Fortanix, Inc.	1.0	August 6, 2020	Initial Release
Fortanix, Inc.	1.1	April 22, 2021	Updates to include new hardware versions
Fortanix, Inc.	1.2	September 29, 2021	Addressed review comments

1. Module Overview

Fortanix SDKMS appliance is the building block for running Fortanix Self-Defending Key Management Service™ (SDKMS), a unified HSM and Key Management solution. With SDKMS, you can securely generate, store, and use cryptographic keys and certificates, as well as secrets, such as passwords, API keys, tokens, or any blob of data. SDKMS ensures that you remain in complete control over your keys and secrets. Your business-critical applications and containers can integrate with SDKMS using legacy cryptographic interfaces or using its native RESTful interface. SDKMS provides control of and visibility into your key management operations using a centralized web-based UI with enterprise level access controls and comprehensive auditing. SDKMS is built to scale horizontally and geographically as your demand for managing your keys and secrets increase, while providing automated load-balancing and high availability.

FIPS 140-2 conformance testing was performed at Security Level 3. The following configuration was tested by the lab.

Module Name and Version	Hardware Version	Firmware Version
Fortanix SDKMS Appliance (FX2200)	FX2200-II-T-F FX2200-II-SX-F FX2200-II-TN-F FX2200-II-SXN-F	3.6.20

Table 1 - Configurations tested

FIPS Security Area	Security Level
Cryptographic Module Specification	3
Cryptographic Module Ports and Interfaces	3
Roles, Services and Authentication	3
Finite State Model	3
Physical Security	3

FIPS Security Area	Security Level
Operational Environment	N/A
Cryptographic Key Management	3
EMI/EMC	3
Self-tests	3
Design Assurance	3
Mitigation of Other Attacks	N/A

Table 2- Security Level Specification Table

1.1 Cryptographic Boundary

The module is a Hardware module, satisfying a multi-chip standalone embodiment. The module supports a limited operational environment with a Firmware Integrity Test (HMAC-SHA256 and a 16-bit EDC Checksum) and Firmware Download Test (ECDSA P-256 SHA-256 Signature Verification). Only trusted code signed by Fortanix may be loaded into the module.

The cryptographic boundary of the module is the enclosure that contains components of the module. The removable power supply units (PSU) are not part of the boundary. The strong enclosure of the cryptographic module is opaque within the visible spectrum. The module uses tamper evident labels to provide the evidence of tampering. The module contains tamper response and zeroization circuitry.

Figure 1 FX2200 Front view (FX2200-II-T-F)

Figure 2 FX2200 Rear view (FX2200-II-T-F)

Figure 3 FX2200 Rear View (FX2200-II-SX-F)

1.1.1 Hardware Block Diagram

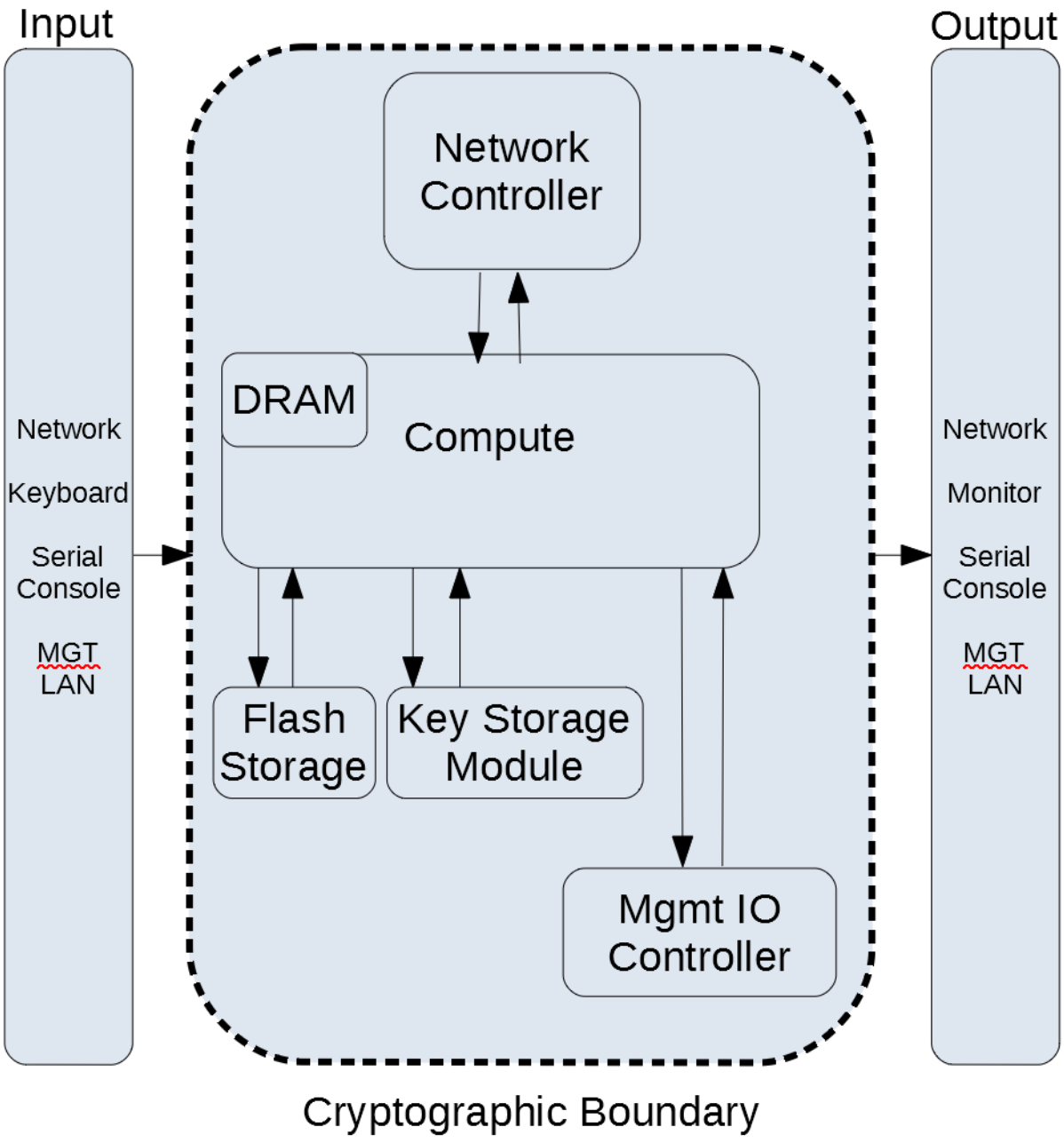

Figure 4 Hardware Block Diagram

1.1.2 Software Block Diagram

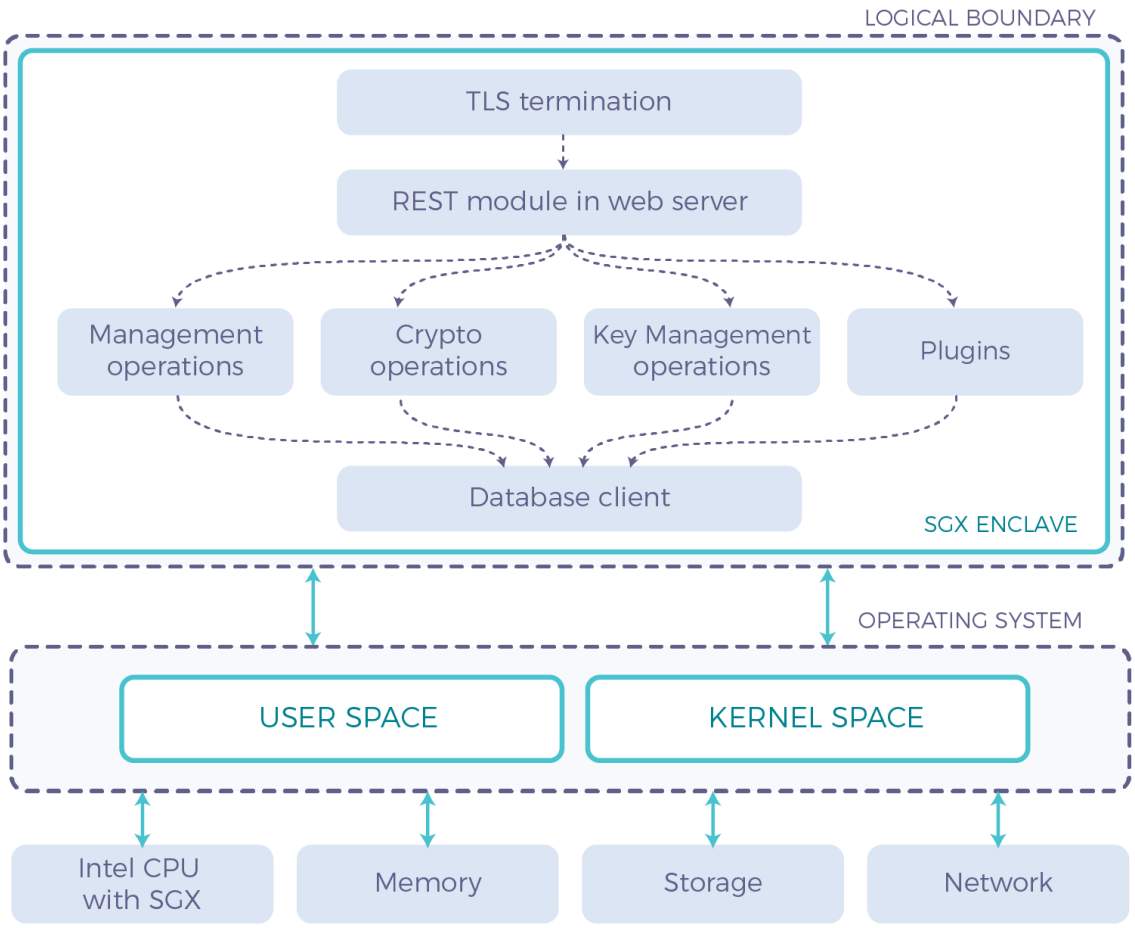

Figure 5 Software Block Diagram

2. Modes of Operations

The module always operates in the FIPS approved mode. The Crypto Officer shall follow these steps to verify the module is running in the FIPS Approved Mode:

1. Invoke the version API provided by the "Get status" service
2. Verify that the output is correct, with the following format and value of "fips_level" attribute is 3:

```
{
   "version":"3.6.20",
   "api_version":"v1-20170718",
   "server_mode":"Sgx",
   "fips_level":3
}
```

2.1.1 Approved Cryptographic Functions

CAVP Cert #	Algorithm	Standard	Mode/ Method	Key Lengths, Curves or Moduli	Use
C1628	AES	FIPS 197, SP 800-38D, SP 800-38C SP 800-38F	ECB, CBC, OFB, CTR, CFB 128, GCM, CCM, KW, KWP	128, 192, 256	Data Encryption / Decryption
Vendor affirmed	AES	SP 800-38G	FF1[1]	128, 192, 256	Data Encryption / Decryption
C1628	CMAC	SP 800-38B	AES	128, 192, 256	Message Authentication
C1628	CVL	SP 800 56-B	RSADP	2048	Data encapsulation
Vendor affirmed	CKG	SP 800-133			Cryptographic key generation
C1628	CVL	SP 800-135	KDF[2]		Key Establishment
C1628	DRBG	SP 800-90Ar1	CTR_DRBG With Derivation Function		Deterministic Random Bit Generation
C1628	ECDSA	FIPS 186-4		P-192[3], P-224, P-256 P-384, P-521	Key Pair Generation,

[1] Supports radix values of 2 to 36, min length is based on radix such that $radix^{minlen} >= 100$, max length is 2^{16} and Max tweak length (maxTlen) is same as maxlen

[2] As per FIPS 140-2 IG D.11 no parts of this protocol, other than the KDF, have been tested by the CAVP and CMVP.

[3] Module does not allow P-192 and/or SHA-1 for ECDSA signature generation. The minimum hash sizes allowed by the module are SHA-256 for P-224, SHA-256 for P-256, SHA-384 for P-384, and SHA-512 for P-521. Module does not allow key pair generation with P-192.

CAVP Cert #	Algorithm	Standard	Mode/ Method	Key Lengths, Curves or Moduli	Use
					Digital Signature Generation and Verification
C1628	HMAC	FIPS 198-1	HMAC-SHA-1 HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512	112, 128, 192, 256	Message Authentication
C1628	KBKDF	SP 800-108	KBKDF		Key Derivation
C1628	KTS	SP 800-38F	GCM	256	Key establishment methodology provides 256 bits of encryption strength; this is claimed for TLS encryption key.
C1628	KTS	SP 800-38F	KW, KWP	128, 192, 256	Key establishment methodology provides between 128 and 256 bits of encryption strength; this is claimed for the symmetric key.
C1628[4]	CVL Partial DH	SP 800-56A	ECC SHA-512	P-224, P-256, P-384, P-521	Shared Secret Computation

[4] Please note this is a latent functionality not used by the module and disabled by firmware. Module does not support ECC CDH.

CAVP Cert #	Algorithm	Standard	Mode/ Method	Key Lengths, Curves or Moduli	Use
C1628	RSA	FIPS 186-4,	PKCS1 v1.5; GenKey9.31; PSS SHA-1, SHA-256, SHA-384, SHA-512	1024[5], 2048, 3072, 4096[6]	Digital Signature Generation and Verification
C1628	SHS	FIPS 180-4	SHA-1, SHA-256, SHA-384 SHA-512		Message Digest
C1627	SHS	FIPS 180-4	SHA-512		Message Digest (pam_module)

Table 3 - Table of Approved Algorithms

For additional information on transitions associated with the use of cryptography refer to NIST Special Publication SP 800-131Ar1. This document can be located on the CMVP website at: (http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-131Ar1.pdf).
The data in the tables will inform Users of the risks associated with using a particular algorithm and a given key length.

[5] Module does not allow 1024-bit keys and SHA-1 for RSA Signature Generation.

[6] As per FIPS 140-2 Implementation Guidance A.14 CAVP validation has been performed on key sizes testable via CAVS, while the cryptographic module supports any RSA modulus size between 2048 and 8192.

2.1.2 Not Approved but Allowed Algorithms

Algorithm	Caveat	Use
NDRNG	Only used to seed the CTR_DRBG with derivation function. This provides 256 bits of security strength.	Seeding for the Approved DRBG
PBKDF	No Security Claimed	Used for obfuscation of passwords, considered as plaintext
RSA Key Wrapping	RSA (CVL Cert. #C1628, key wrapping);	Key Wrapping

Table 4 - Table of Non-Approved but Allowed Algorithms

3. Ports and Interfaces

The following table describes physical ports and logical interfaces of the module.

Port Name	Count	Interface(s)
Ethernet Ports (RJ45 or SFP28)	2	Data Input, Data Output, Control Input, Status Output
IPMI Port	1	Data Input, Data Output, Control Input, Status Output
VGA Port	1	Data Output, Status Output
Serial Port	1	Data Input, Data Output, Control Input, Status Output
USB Port	2	Data Input, Control Input
Power Receptacle	2	Power Input
Hard Disk Activity LED	1	Status Output
Power button with LED	1	Control Input, Status Output
ID button with LED	2	Control Input, Status Output

Table 5- Specification of Cryptographic Module Logical Interfaces

The module does not include a maintenance interface.

4. Roles, Services and Authentication

The module supports identity-based authentication for all roles. The module supports a Crypto Officer and User Role.

- The Crypto Officer installs and administers the module.
- The User uses the cryptographic services provided by the module. This role is assumed both by an actual user of the system and an external system that requires cryptographic services.

The module supports a variety of roles that are mapped to the two FIPS roles. Following table enumerates the mapping between module roles and FIPS roles:

Module Role	FIPS Role
System Administrator	Crypto Officer
System Operator	Crypto Officer
Account Administrator	Crypto Officer, User
Account Member	Crypto Officer, User
Account Auditor	Crypto Officer
Group Administrator	Crypto Officer, User
Group Auditor	Crypto Officer
Application	User
Console User	Crypto Officer

Table 6 – Mapping of Module Roles to FIPS roles

4.1 Authenticated Services

The module provides the following services:

Service	Module Roles	Cryptographic Keys, CSPs and Public Keys	Types of Access to Cryptographic Keys and CSPs R – Read or Execute W – Write or Create Z – Zeroize
Authentication	**System Administrator** **System Operator** **Account Administrator** **Account Member** **Account Auditor** **Group Administrator** **Group Auditor** **Application** **Console User**	User password and Console user password	R
		API key	R
		RSA Public key of external Application	R
		Public key of an outside entity/server (CA)	R
		User 2FA device public key	R
		Bearer token	R
Create/Generate key	**Account Administrator** **Group Administrator** **Application** **Console User**	Database Wrapping key	W
		DRBG Entropy Input String	W
		DRBG Seed	R, W
		DRBG internal state	W
		Symmetric key	W
		HMAC key	W
		RSA private key for Digital Signatures	W
		RSA private key for Key Encapsulation Operations	W
		ECDSA private key	W
		ECDSA public key	W
		RSA public key for Key Encapsulation Operations	W
		RSA public key for Digital Signatures	W
Encrypt/Decrypt	**Application**	Symmetric key	R
		Cipher State Wrapping key	R, W
		SP 800-108 KDF internal state	R
		Account key	R
		Database Wrapping key	R
Sign/Verify	**Application**	Database Wrapping key	R
		RSA private key for Digital Signatures	R
		RSA public key for Digital Signatures	R

Service	Module Roles	Cryptographic Keys, CSPs and Public Keys	Types of Access to Cryptographic Keys and CSPs R – Read or Execute W – Write or Create Z – Zeroize
		ECDSA private key	R
		ECDSA public key	R
		ECDSA random number “k"	R, W
Wrap/Unwrap	**Application**	Database Wrapping key	R
		Symmetric key	R
		RSA private key for Key Encapsulation Operations	R
		RSA public key for Key Encapsulation Operations	R
HMAC	**Application**	Database Wrapping key	R
		HMAC key	R
Digest	**Application**	N/A	N/A
Import Key	**Account Administrator Group Administrator Application**	Database Wrapping key	R
		Symmetric key	W
		HMAC key	W
		RSA private key for Digital Signatures	W
		RSA private key for Key Encapsulation Operations	W
		RSA public key for Key Encapsulation Operations	W
		ECDSA private key	W
		ECDSA public key	W
		RSA public key for Digital Signatures	W
Export Key	**Account Administrator Group Administrator Application**	Database Wrapping key (This key is not exported)	R
		Symmetric key – if it was created or imported with export permission	R
		HMAC key – if it was created or imported with export permission	R
		RSA private key for Digital Signatures – if it was created or imported with export permission	R
		RSA public key for Digital Signatures – if it was created or imported with export permission	R

Service	Module Roles	Cryptographic Keys, CSPs and Public Keys	Types of Access to Cryptographic Keys and CSPs R – Read or Execute W – Write or Create Z – Zeroize
		RSA private key for Key Encapsulation Operations – if it was created or imported with export permission	R
		RSA public key for Key Encapsulation Operations – if it was created or imported with export permission	R
		ECDSA private key – if it was created or imported with export permission	R
		ECDSA public key – if it was created or imported with export permission	R
System configuration and management	**System Administrator System Operator (Read only) Account Administrator Account Member Account Auditor (Read only) Group Administrator Group Auditor (Read only) Application**	Cluster Master key	R
		System key	R, W
		Account Wrapping key	R, W
		Account key	R, W
		Database Wrapping key	R, W
		SP 800-108 KDF internal state	R, W
		User 2FA device public key	R, W
		RSA Public key of external Application	R, W
		Public key of an outside entity/server (CA)	R, W
Zeroization	**Console User**	Personalization key	Z
TLS[7]	**System Administrator System Operator Account Administrator Account Member Account Auditor**	Cluster RSA private key for TLS	R
		Cluster RSA public key for TLS	R
		SP 800-135 TLS KDF internal state	R, W
		TLS integrity key (AES)	R,W
		TLS encryption key (AES)	R,W
		TLS pre-master secret	R,W
		TLS master secret	R,W

[7] All API calls into the module are done over TLS V1.2. No parts of these protocols, other than the KDFs, have been tested by the CAVP and CMVP.

Service	Module Roles	Cryptographic Keys, CSPs and Public Keys	Types of Access to Cryptographic Keys and CSPs R – Read or Execute W – Write or Create Z – Zeroize
	Group Administrator **Group Auditor** **Application**	Public key of an outside entity/server (CA)	R
		RSA Public key of external Application	R

Table 7 - Services Authorized for Roles, Access Rights within Services

4.2 Unauthenticated Services

Services
Get status
Run self-tests
Signup

Table 8 - Unauthenticated Services

4.3 Authentication

The module supports the following authentication mechanisms.

Module Role	Authentication Type	Authentication Data
System Administrator System Operator Account Administrator Account Member Account Auditor Group Administrator Group Auditor Console User	Identity Based	User password and console user password
Application	Identity Based	API key
Application	Identity Based	RSA Public key of external Application
Application	Identity Based	Public key of an outside entity/server (CA)
System Administrator System Operator Account Administrator Account Member Account Auditor Group Administrator Group Auditor	Identity Based	User 2FA device public key
System Administrator System Operator Account Administrator Account Member Account Auditor Group Administrator Group Auditor Application	Identity Based	Bearer token

Table 9- Roles and required Identification and Authentication

Our password authentication policy is as described for the Memorized Secret Authenticators in NIST SP 800-63B (8 characters or longer). The module supports concurrent operators and the module levies a restriction on session expiry time where if inactive, the Application's role session will expire in 10 minutes by default. Similarly, for all other Module roles there is a session expiry time of 24 hours. Session expiry time can be customized.

Authentication Mechanism	Strength of Mechanism
User password and console user password	Minimum password length is 8 characters. For a user who just meets the minimum password length, each of the eight characters will have at least 95 possible characters if we consider just the printable characters, although module supports UTF-8 characters for password and the number of possible characters with UTF-8 is much higher. Total number of password permutations with eight characters is 95^8 = 6,634,204,312,890,625. Therefore, the probability of guessing a password is significantly less than one in 1,000,000. Module only allows at the most 10 authentication attempts in a second. Therefore, a user could try at most 600 passwords in a minute. Given the total number of possible permutations (as shown above), the probability a random attempt in one-minute period to be correct will be 600/6,634,204,312,890,625. Therefore, the probability of guessing a password in a one-minute period is significantly less than one in 100,000.
API key	An application authenticates using an API key which contains app secret. App secret is a 64 bytes random data. Total number of permutations for app secret will be 2^512. Therefore, the probability of guessing an application's secret is significantly less than one in 1,000,000. Module only allows at the most 10 authentication attempts in a second. Therefore, a user could try at most 600 attempts in a minute. Given the total number of possible permutations (as shown above), the probability a random attempt in one-minute period to be correct will be

Authentication Mechanism	Strength of Mechanism
	600/(2^512). Therefore, the probability of guessing an app secret in a one-minute period is significantly less than one in 100,000.
User 2FA device public key	The module allows users to use a second factor authentication mechanism in addition to username and password. The strength of this combination mechanism relies upon the strength of the User password mechanism (described earlier) combined with the strength of two factor authentication. This mechanism adds more strength to the password mechanism which already far exceeds the FIPS requirements. U2F signature verification uses U2F device's public key which is an EC P-256 key. Security strength of this key is 128 bits. So, the probability of a random success will be 1 in 2^128. Probability of this combined scheme = (Probability of guessing username and password) * (Probability from signature verification scheme), which is 1/(95^8) * 1/(2^128). Therefore, the probability of guessing a password is significantly less than one in 1,000,000. Module only allows at the most 10 authentication attempts in a second. Therefore, a user could try at most 600 attempts in a minute. Given the total number of possible permutations (as shown above), the probability a random attempt in one-minute period to be correct will be 600/(95^8 * 2^128). Therefore, the probability of guessing a password in a one-minute period is significantly less than one in 100,000. Therefore, this mechanism of additional 2FA also far exceeds the FIPS requirements.
RSA Public key of external Application	The strength of this mechanism is based on the size of the private key space. The module relies upon minimum RSA 2048-bit keys. This provides an encryption strength of 112

Authentication Mechanism	Strength of Mechanism
	bits, so the probability of a random success will be 1 in 2^112, which is significantly less than one in 1,000,000. Using this mechanism, one can make very few attempts in one-minute period. Each attempt will require the module to check the signature on the certificate using FIPS approved signature algorithm and establishing TLS session with this certificate. On an average only one attempt can be made in a second. Therefore, at the most 60 attempts can be made in a one minute period. Therefore, the probability of guessing a 2048-bit private key and succeeding in a one minute period is 60/(2^112) which is significantly less than one in 100,000.
Bearer token	The bearer token is a base64 encoded random 64 bytes data which is generated using approved DRBG in SDKMS. Total number of permutations is 2^512. Therefore, the probability of guessing the token is 1/(2^512), which is significantly less than one in 1,000,000. Each authentication attempt takes approximately 12ms or more. Therefore, a user could try at most 5,000 attempts in a minute. Given the total number of possible permutations (as shown above), the probability a random attempt in one-minute period to be correct will be 5000/(2^512). Therefore, the probability of guessing a password in a one-minute period is significantly less than one in 100,000.
Public key of an outside entity/server (CA)	The strength of this mechanism is based on the size of the private key space. The module relies upon RSA 2048-bit node keys. This provides an encryption strength of 112 bits, so the probability of a random success will be 1 in 2^112, which is significantly less than one in 1,000,000.

Authentication Mechanism	Strength of Mechanism
	Each attempt will require the module to check the signature on the certificate using FIPS approved signature algorithm and establishing TLS session with this certificate. Each attempt takes 100ms or more. Therefore, at the most 600 attempts can be made in a one minute period. Therefore, the probability of guessing a 2048-bit private key and succeeding in a one minute period is 600/(2^112) which is significantly less than one in 100,000.

Table 10 - Strength of Authentication Mechanisms

5. Secure Operation Rules

5.1 Module Initialization and Setup

The Crypto Officer is required to follow the vendor procedural control guidelines to setup and install the module after it is received. Here is a brief summary of the procedure.

1. Module unpacking must be done in a secure location where only authorized personnel have access.
2. The installation must be carried out by authorized personnel who has crypto officer role in the organization. The installation must be carried out in a secure location which is accessible only by authorized personnel.
3. Login using default credentials and change password
4. Setup networking interfaces
5. Setup NTP
6. Perform setup using set of setup commands
7. Once setup is complete, run version command to check firmware version and verify FIPS mode.

6. Self-tests

The module performs the following power-up and conditional self-tests. Upon successful execution of **all** power-up self-test, module provides the following status message that can be viewed by console user:

"*Software Integrity test succeeded*"
"*Power-up self-tests succeeded*"

Upon failure of a power-up or conditional self-test, the module halts its operation and enters the error state. The error messages are provided via console (VGA port). The following tables describe self-tests implemented by the module along with status messages.

6.1 Power-Up Self Tests

Algorithm	Test	Status
AES 128-bit key size in ECB, CBC, CFB128, and CTR Modes 192-bit key size ECB, CBC, and CFB128 Modes 256-bit key size ECB, CBC, and CFB128 Modes	KAT (encryption)	Success: "*Power-up self-test succeeded*" Error: "*AES self test failed*"
AES 128-bit key size in ECB, CBC, CFB128, and CTR Modes 192-bit key size ECB, CBC, and CFB128 Modes 256-bit key size ECB, CBC, and CFB128 Modes	KAT (decryption)	Success: "*Power-up self-test succeeded*" Error: "*AES self test failed*"
AES GCM 128-bit, 192-bit, and 256-bit key size	KAT (encryption)	Success: "*Power-up self-test succeeded*" Error: "*GCM self test failed*"

Algorithm	Test	Status
AES GCM 128-bit, 192-bit, and 256-bit key size	KAT (decryption)	Success: "*Power-up self-test succeeded*" Error: "*GCM self test failed*"
AES CCM 128-bit key size	KAT (encryption)	Success: "*Power-up self-test succeeded*" Error: "*CCM self test failed*"
AES CCM 128-bit key size	KAT (decryption)	Success: "*Power-up self-test succeeded*" Error: "*CCM self test failed*"
AES KW 128-bit, 192-bit, and 256-bit key size	KAT	Success: "*Power-up self-test succeeded*" Error: "*KW/KWP self test failed*"
AES KWP 128-bit, 192-bit, and 256-bit key size	KAT	Success: "*Power-up self-test succeeded*" Error: "*KW/KWP self test failed*"
ECC CDH Primitive "Z" P-224 Curve	KAT	Success: "*Power-up self-test succeeded*" Error: "*KAS ECC Primitive Z test failed*"
SHA-1	KAT	Success: "*Power-up self-test succeeded*" Error: "*SHA1 self test failed*"
SHA-256	KAT	Success: "*Power-up self-test succeeded*" Error: "*SHA256 self test failed*"
SHA-512	KAT	Success: "*Power-up self-test succeeded*" Error: "*SHA512 self test failed*"
HMAC-SHA-1 128-bit key size	KAT	Success: "*Power-up self-test succeeded*" Error: "*HMAC SHA1 self test failed*"

Algorithm	Test	Status
HMAC-SHA-256 128-bit key size	KAT	Success: "*Power-up self-test succeeded*" Error: *"HMAC SHA256 self test failed"*
HMAC-SHA-512 2048-bit key size	KAT	Success: "*Power-up self-test succeeded*" Error: *"HMAC SHA512 self test failed"*
SP 800-90A DRBG SO800-90A section 11.3 health tests	KAT	Success: "*Power-up self-test succeeded*" Error: *"CTR DRBG self test failed"*
RSA 2048-bit key size, SHA-256 (PKCS1 v1.5)	Signature generation/verification KAT	Success: "*Power-up self-test succeeded*" Error: *"RSA self test failed"*
RSA DP 2048 bit key size	Encryption / Decryption	Success: "*Power-up self-test succeeded*" Error: *"RSA self test failed"*
ECDSA P-224 curve	Signature generation/verification pairwise consistency test	Success: "*Power-up self-test succeeded*" Error: *"ECDSA self test failed"*
SP 800-135 TLS V1.2 KDF	KAT	Success: "*Power-up self-test succeeded*" Error: *"TLS 1.2 KDF self test failed"*
SP 800-108 KDF 256-bit key size	KAT	Success: "*Power-up self-test succeeded*" Error: *"KDF108 self test failed"*
HMAC-SHA-256 256-bit key size	Software integrity test	Success: *"Software Integrity test succeeded"* Error: *"Software integrity check failed"*
Checksum	Firmware integrity test	Success: *"Software integrity test of personalization key store module succeeded"* Error: *"Integrity check failed"*
FF1	KAT	Success: "*Power-up self-test succeeded*"

Algorithm	Test	Status
		Error: *"FF1 self test failed"*
CMAC 128-bit, 192-bit, and 256-bit key size	KAT	Success: "*Power-up self-test succeeded"* Error: "*CMAC self test failed"*
SHA-512 pam_module	KAT	Success: "*Power-up self-test succeeded"* Error: *"panic"*
Critical Functions Tests	N/A	N/A

Table 11 – Power-Up Self-tests

6.2 Conditional Self Tests

Algorithm	Test	Status
Continuous RNG test performed on output of NDRNG (RDSEED)	Continuous Random Number Generator (RNG) Test	Error: *"FIPS conditional test failure: Error in cryptographic operation – RNG failed"*
Continuous RNG test performed on output of software-based Approved SP 800-90A CTR_DRBG	Continuous Random Number Generator (RNG) Test	Error: *"FIPS conditional test failure: Error in cryptographic operation – RNG failed"*
RSA 2048-bit to 8192-bit key size SHA-256, SHA-384, SHA-512	Pairwise Consistency Test (Sign and Verify)	Error: *"FIPS conditional test failure: Pairwise consistency test failed. Sign / Verify test failed."*
RSA 2048-bit to 8192-bit key size	Pairwise Consistency Test (Encrypt and Decrypt)	Error: *"FIPS conditional test failure: Pairwise consistency test failed. Encryption / Decryption test failed."*

Algorithm	Test	Status
ECDSA P-224, P-256, P-384, P-521 SHA-256, SHA-384, SHA-512	Pairwise Consistency Test (Sign and Verify)	Error: *"FIPS conditional test failure: Pairwise consistency test failed. Sign / Verify test failed."*
Bypass Test	N/A	N/A
Firmware Load Test ECDSA P-256 SHA-256	Signature verification test	Error: *"Firmware verification failed."*
Manual Key Entry Test	N/A	N/A

Table 12- Conditional Self-tests

7. Physical Security

The cryptographic module consists of production-grade components. The strong enclosure of the cryptographic module is opaque within the visible spectrum. The removable covers are protected with tamper-evident seals. The tamper-evident seals must be checked periodically by the Crypto Officer. If the tamper-evident seals are broken or missing, the Crypto Officer must halt the operation of the module and ship the module to Fortanix for replacement.

The module contains tamper response and zeroization circuitry. The tamper response and zeroization circuitry immediately zeroizes all plaintext secret and private keys and CSPs when a cover is removed. The tamper response and zeroization circuitry remains operational when plaintext secret and private cryptographic keys or CSPs are contained within the cryptographic module. Ventilation holes are constructed in a manner that prevents undetected physical probing inside the enclosure.

7.1 Inspection/Testing of Physical Security Mechanisms

The following guidelines should be considered when producing an Operational Policy for the environment for which the module is deployed.

The SDKMS appliance enclosure should be periodically checked by the Crypto Officer for evidence of tampering damage to the two tamper-evident labels and any physical damage to the enclosure material.

The frequency of a physical inspection depends upon the information being protected and the environment in which the unit is located. At a minimum, it would be expected that a physical inspection would be made by the Crypto Officer at least monthly.

The tamper evident labels are applied at the Fortanix manufacturing facility, are serialized, and are not available for order or replacement from Fortanix. The labels are designed and intended to stay intact for the entire life of the module. The labels are applied in the two positions shown in the figure below.

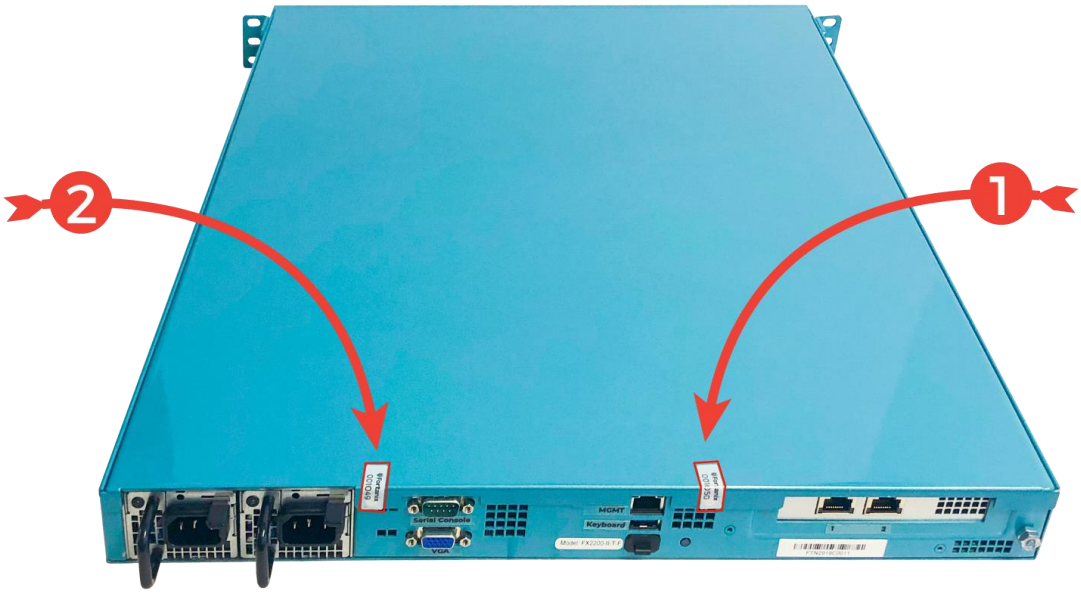

Figure 6 Tamper Evident Label Positions on FX2200-II-T-F and FX2200-II-SX-F

Figure 7 Tamper Evident Label Positions on FX2200-II-TN-F and FX2200-II-SXN-F

Following figure shows the tamper label. It leaves “VOID” markings in place of tamper label and the tamper label cannot be reapplied.

Figure 8 Tamper Evident Label used on on FX2200-II-T-F and FX2200-II-SX-F

Figure 9 Tamper Evident Label used on on FX2200-II-TN-F and FX2200-II-SXN-F

The two tamper seals sit over a screw on the lid and extend over the lid seam to the module chassis, as shown in the figure below. The only way to remove the cover is to break or damage the tamper seals.

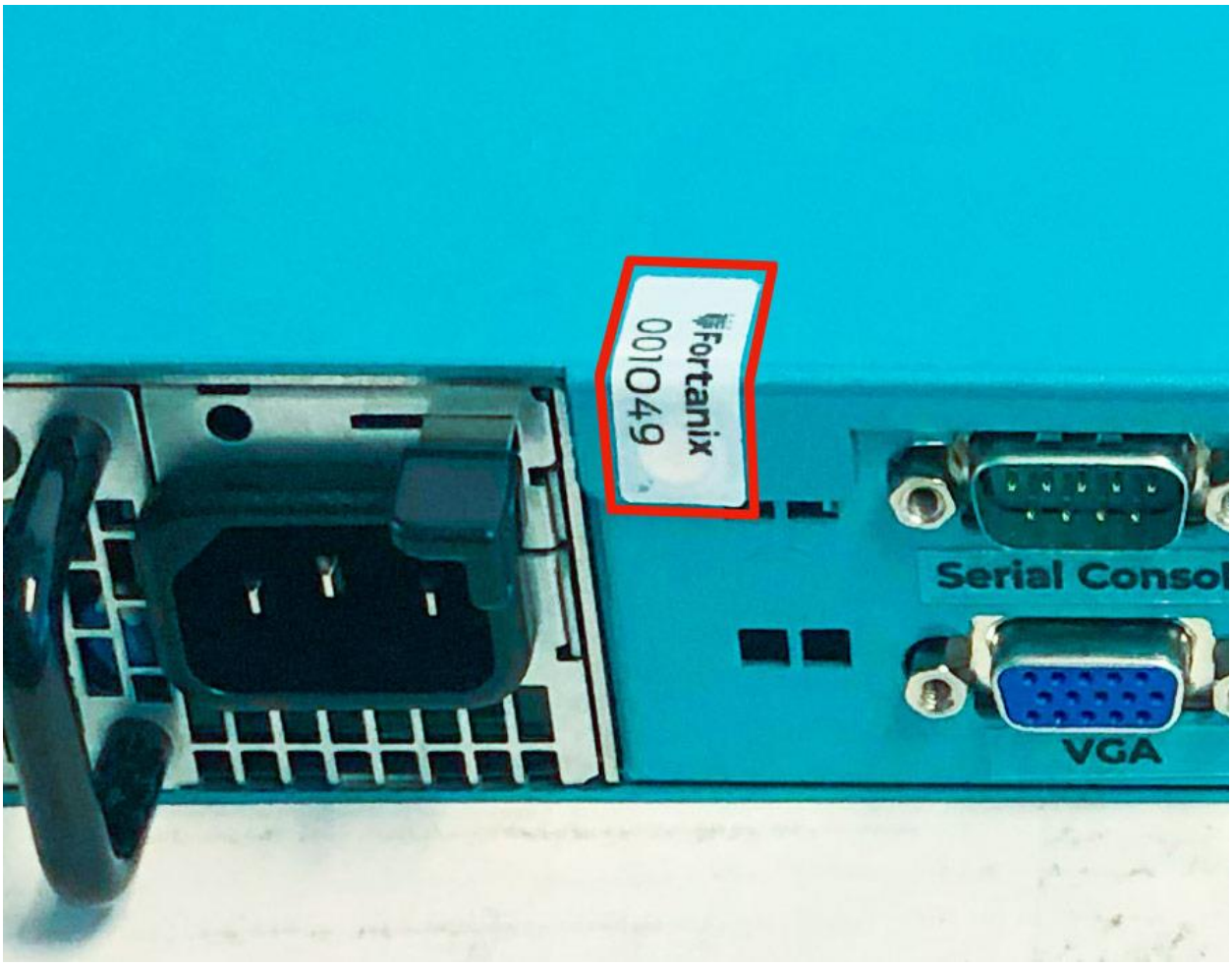

Figure 10 Tamper Evident Label Closeup - FX2200-II-T-F and FX2200-II-SX-F

Figure 11 Tamper Evident Label Closeup - FX2200-II-TN-F and FX2200-II-SXN-F

NOTE: The module hardness testing was performed at an ambient room temperature of 80.4°F and no assurance is provided for Level 3 hardness conformance at any other temperature. Following table summarized inspection / testing of physical security mechanisms

Physical Security Mechanisms	Recommended Frequency of Inspection / Test	Inspection / Test Guidance Details
Tamper Evident Labels	Monthly	The SDKMS appliance enclosure should be periodically checked by the Crypto Officer for evidence of tampering damage to the two tamper-evident labels and any physical damage to the enclosure material.

Table 13 Inspection / Testing of Physical Security Mechanisms

The following non-security relevant components has been excluded from FIPS 140-2 requirements:

- 8-pin power connector cable (non-security relevant)
- Power distributor (R1T2-5300G2H) board / circuitry for power supply receptacle (non-security relevant)
- Wiring connected to Front bezel LEDs and buttons (non-security relevant)
- Components U2 (NOR Gate) and C6 (Capacitor) on network card (PE3251G21/1-SR) (non-security relevant)
- Hardware component H2 (chassis body assembly screw) and EC19 (Capacitor) on main board (MX32-4L0) (non-security relevant)
- Component U42 (32 bit CMOS Flash) on main board (MX32-4L0) (non-security relevant)

This document can be freely distributed in its entirety without modification

8. Mitigation of Other Attacks Policy

The cryptographic module is not designed to mitigate any other attacks beyond the specific scope of FIPS 140-2.

Other Attacks	Mitigation Mechanism	Specific Limitations
N/A	N/A	N/A

Table 14- Table of Mitigation of Other Attacks

9. Security Rules

1. The module enforces logical separation between all data inputs, data outputs, control inputs, and status outputs via the cryptographic module API.
2. The cryptographic module inhibits all data output during self-tests and error states. The data output interface is logically disconnected from the processes performing self-tests and zeroization.
3. The cryptographic module conforms to the EMI/EMC requirements specified by 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, Digital Devices, Class B (i.e. for Home use) which vacuously satisfies Class A.
4. Power-up self-tests do not require any operator intervention.
5. Power-up self-tests may be initiated on demand by power-cycling the module.
6. The cryptographic module does not support a maintenance interface or maintenance role.
7. The cryptographic module does not support manual key entry.
8. The cryptographic module does not support a bypass capability.
9. Results of previous authentications are cleared when the module is powered off. The operator is required to re-authenticate into the module.
10. The operator can Power cycle the module in order to exit the error states and resume normal operation.
11. The module protects public keys and CSPs from unauthorized disclosure, unauthorized modification, and unauthorized substitution.
12. The module does not output intermediate key values.
13. The module complies with FIPS 140-2 IG A.5 requirements for AES-GCM:
    a. For TLS V1.2 Protocol, the module constructs the IV (internally) as allowed per Technique #1 in FIPS 140-2 IG A.5 for Industry Protocols. The IV total length is 96-bits, where the fixed IV length is 32-bits and nonce_explicit part of the IV is 64-bits. The GCM key and IV are session specific; if the module loses power the implementation is required to re-initialize a TLS V1.2 session, creating a new IV altogether.
    b. For the Encrypt/Decrypt service, a 96-bit IV is constructed from the output of the CTR_DRBG, allowed as per Technique #2 in FIPS 140-2 IG A.5 for IVs generated "internally at its entirety randomly". In case the module's power is lost and then restored, a new key for use with the AES GCM encryption/decryption will be generated from the output of the CTR_DRBG.
14. In accordance with FIPS 140-2 IG D.12, the cryptographic module performs Cryptographic Key Generation (CKG) as per SP 800-133 (Vendor Affirmed). The resulting generated symmetric key and/or generated seed for asymmetric key generation, are from the unmodified output of the SP 800-90A DRBG.

15. The module complies with FIPS 140-2 IG A.10 requirements for SP 800-38G. The module supports radix values of 2 to 36, min length is based on radix such that $radix^{minlen} >= 100$, max length is 2^{16} and Max tweak length (maxTlen) is same as maxlen
16. The module complies with FIPS 140-2 IG D.1-rev2 for SP 800-56Ar2. The module supports approved DLC primitives and uses approved hash algorithms.

10. Appendix A: CSPs

1. Personalization key
   - Description: 256-bit Key Derivation key (SP 800-108 KDF) used to derive the Sealing key
   - Generation: SP 800-90A CTR_DRBG; As per SP 800-133 Section 7.1, key generation is performed as per the “Direct Generation” of Symmetric Keys which is an Approved key generation method
   - Establishment: N/A
   - Entry: N/A
   - Output: N/A
   - Storage: Plaintext in RAM, plaintext in persistent storage
   - Key-to-Entity: This key belongs to the node
   - Zeroization: On tamper and zeroization service

2. Sealing key
   - Description: 256-bit Key Derivation key (SP 800-108 KDF) used to derive the System key and Account Wrapping key
   - Generation: Derived from Personalization key using NIST SP 800-108 KDF in Feedback Mode (§5.2); As per SP 800-133 Section 7.4, key derivation is performed by an Approved KDF which is an Approved key derivation method
   - Establishment: N/A
   - Entry: N/A
   - Output: N/A
   - Storage: Plaintext in RAM
   - Key-to-Entity: This key belongs to the node
   - Zeroization: On tamper and zeroization service

3. Cluster Master key
   - Description: 256-bit Key Derivation key (SP 800-108 KDF) used to derive the System key and Account Wrapping key
   - Generation: SP 800-90A CTR_DRBG; As per SP 800-133 Section 7.1, key generation is performed as per the “Direct Generation” of Symmetric Keys which is an Approved key generation method
   - Establishment: N/A
   - Entry: N/A
   - Output: N/A
   - Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Sealing key

- Key-to-Entity: This key belongs to the single node cluster
- Zeroization: On tamper and zeroization service

4. System key
   - Description: 256-bit AES GCM key used to wrap all user and session information that is stored in persistent storage
   - Generation: Derived from Cluster Master key using NIST SP 800-108 KDF in Feedback Mode (§5.2); As per SP 800-133 Section 7.4, key derivation is performed by an Approved KDF which is an Approved key derivation method
   - Establishment: N/A
   - Entry: N/A
   - Output: N/A
   - Storage: Plaintext in RAM
   - Key-to-Entity: This key belongs to the single node cluster
   - Zeroization: On tamper and zeroization service

5. Account Wrapping key
   - Description: 256-bit AES GCM key used to wrap Account key when it is stored in persistent storage
   - Generation: Derived from Cluster Master key using NIST SP 800-108 KDF in Feedback Mode (§5.2); As per SP 800-133 Section 7.4, key derivation is performed by an Approved KDF which is an Approved key derivation method
   - Establishment: N/A
   - Entry: N/A
   - Output: N/A
   - Storage: Plaintext in RAM
   - Key-to-Entity: This key belongs to the single node cluster
   - Zeroization: On tamper and zeroization service

6. Account key
   - Description: 256-bit Key Derivation key (SP 800-108 KDF) used to derive the Database Wrapping key and Cipher State Wrapping key
   - Generation: SP 800-90A CTR_DRBG; As per SP 800-133 Section 7.1, key generation is performed as per the “Direct Generation” of Symmetric Keys which is an Approved key generation method
   - Establishment: N/A
   - Entry: N/A
   - Output: N/A

- Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Account Wrapping key
- Key-to-Entity: This key belongs to a specific account / tenant and is unique to every account
- Zeroization: On tamper and zeroization service

7. Database Wrapping key
   - Description: 256-bit AES GCM key used to wrap all account / tenant data and keys that belong to a specific account / tenant when it is stored in persistent storage
   - Generation: Derived from Account key using NIST SP 800-108 KDF in Feedback Mode (§5.2); As per SP 800-133 Section 7.4, key derivation is performed by an Approved KDF which is an Approved key derivation method
   - Establishment: N/A
   - Entry: N/A
   - Output: N/A
   - Storage: Plaintext in RAM
   - Key-to-Entity: This key belongs to a specific account / tenant and is unique to every account
   - Zeroization: On tamper and zeroization service

8. Cipher State Wrapping key
   - Description: 128-bit AES GCM key used to wrap all cipher state data that belongs to a specific account / tenant
   - Generation: Derived from Account key using NIST SP 800-108 KDF in Feedback Mode (§5.2); As per SP 800-133 Section 7.4, key derivation is performed by an Approved KDF which is an Approved key derivation method
   - Establishment: N/A
   - Entry: N/A
   - Output: N/A
   - Storage: Plaintext in RAM
   - Key-to-Entity: This key belongs to a specific account / tenant and is unique to every account
   - Zeroization: On tamper and zeroization service

9. Symmetric key
   - Description: 128-bit, 192-bit, or 256-bit AES keys in the following modes:
     - ECB
     - CBC

    - CTR
    - CFB 128
    - OFB
    - GCM
    - CCM Mode
    - KW
    - KWP
    - CMAC
    - FF1
- Generation: SP 800-90A CTR_DRBG; As per SP 800-133 Section 7.1, key generation is performed as per the “Direct Generation” of Symmetric Keys which is an Approved key generation method
- Establishment: N/A
- Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Import Key" service
- Output: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Export Key" service if the key was created or imported with export permission
- Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Database Wrapping key
- Key-to-Entity: Only authenticated clients can request the use of the key and authorization to access the key is checked. Client making the request must have authorization to use the key
- Zeroization: On tamper and zeroization service

10. HMAC key

- Description: HMAC key with the following key sizes:
    - For HMAC-SHA-1, the minimum key size is 112-bits.
    - For HMAC-SHA-256, the minimum key size is 128-bits.
    - For HMAC-SHA-384, the minimum key size is 192-bits.
    - For HMAC-SHA-512, the minimum key size is 256-bits.
- Generation: SP 800-90A CTR_DRBG; As per SP 800-133 Section 7.1, key generation is performed as per the “Direct Generation” of Symmetric Keys which is an Approved key generation method
- Establishment: N/A
- Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Import Key" service
- Output: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Export Key" service if the key was created or imported with export permission

- Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Database Wrapping key
- Key-to-Entity: Only authenticated clients can request the use of the key and authorization to access the key is checked. Client making the request must have authorization to use the key
  Zeroization: On tamper and zeroization service

11. RSA private key for Digital Signatures
- Description: 2048-bit to 8192-bit RSA key
- Generation: SP 800-90A CTR_DRBG; this key is used for Digital Signature Generation. As per SP 800-133 Section 6.1, key generation is performed as per FIPS 186-4 which is an Approved key generation method
- Establishment: N/A
- Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Import Key" service
- Output: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Export Key" service if the key was created or imported with export permission
- Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Database Wrapping key
- Key-to-Entity: Only authenticated clients can request the use of the key and authorization to access the key is checked. Client making the request must have authorization to use the key
- Zeroization: On tamper and zeroization service

12. RSA private key for Key Encapsulation Operations
- Description: 2048 to 8192-bit RSA key with PKCSv1_5 and OAEP padding
- Generation: SP 800-90A CTR_DRBG; this key is used for Key Un-encapsulation (decryption) operations. This is an allowed method for key transport as per FIPS 140-2 IG D.9
- Establishment: N/A
- Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Import Key" service
- Output: Automatic, Encrypted over TLS session (with TLS encryption key (AES)) during "Export Key" service if the key was created or imported with export permission
- Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Database Wrapping key
- Key-to-Entity: Only authenticated clients can request the use of the key and authorization to access the key is checked. Client making the request must have authorization to use the key

- Zeroization: On tamper and zeroization service

13. ECDSA private key
- Description: EC Key (P-224, P-256, P-384, P-521)
- Generation: SP 800-90A CTR_DRBG; As per SP 800-133 Section 6.1, key generation is performed as per FIPS 186-4 which is an Approved key generation method
- Establishment: N/A
- Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Import Key" service
- Output: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Export Key" service if the key was created or imported with export permission
- Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Database Wrapping key
- Key-to-Entity: Only authenticated clients can request the use of the key and authorization to access the key is checked. Client making the request must have authorization to use the key
- Zeroization: On tamper and zeroization service

14. ECDSA random number “k”
- Description: A secret random number generated via SP 800-90A CTR_DRBG for use during the ECDSA signature generation process. The sizes are as follows:
    - For P-224, k is 224 bits.
    - For P-256, k is 256 bits.
    - For P-384, k is 384 bits.
    - For P-521, k is 521 bits.
- Generation: SP 800-90A CTR_DRBG; As per SP 800-133 Section 6.1, key generation is performed as per FIPS 186-4 which is an Approved key generation method
- Establishment: N/A
- Entry: N/A
- Output: N/A
- Storage: Plaintext in RAM
- Key-to-Entity: Process - “Sign/Verify” service with ECDSA
- Zeroization: On tamper and zeroization service

15. Cluster RSA private key for TLS
- Description: 2048-bit RSA key; when the module behaves as a TLS Server this key is used for RSA Key Un-encapsulation of the TLS pre-master secret
- Generation: SP 800-90A CTR_DRBG; As per SP 800-133 Section 6.2, key generation is performed as per FIPS 186-4; this is an allowed method as per FIPS 140-2 IG D.9

- Establishment: N/A
- Entry: N/A
- Output: N/A
- Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 System key
- Key-to-Entity: This key belongs to the single node cluster
- Zeroization: On tamper and zeroization service

16. SP 800-135 TLS KDF internal state
- Description: 128-byte internal state for SP 800-135 TLS V1.2 KDF (HMAC-SHA-256 PRF or HMAC-SHA-384 PRF)
- Generation: N/A
- Establishment: SP 800-135 Section 4.2.1 or 4.2.2; allowed method as per FIPS 140-2 IG D.8 Scenario 4
- Entry: N/A
- Output: N/A
- Storage: Plaintext in RAM
- Key-to-Entity: Process – TLS KDF internal state
- Zeroization: On tamper and zeroization service

17. TLS integrity key (AES)
- Description: AES-256-GCM key used for both encryption and integrity
- Generation: Derived from TLS master secret using SP 800-135 KDF Section 4.2.1 or 4.2.2; allowed method as per FIPS 140-2 IG D.8 Scenario 4
- Establishment: N/A
- Entry: N/A
- Output: N/A
- Storage: Plaintext in RAM
- Key-to-Entity: Process – TLS
- Zeroization: On tamper and zeroization service

18. TLS encryption key (AES)
- Description: AES with the following modes and key sizes:
    - AES-256-GCM
- Generation: Derived from TLS master secret using SP 800-135 KDF Section 4.2.1 or 4.2.2; allowed method as per FIPS 140-2 IG D.8 Scenario 4
- Establishment: N/A
- Entry: N/A

- Output: N/A
- Storage: Plaintext in RAM
- Key-to-Entity: Process – TLS
- Zeroization: On tamper and zeroization service

19. TLS pre-master secret
    - Description: 48-byte pre-master secret
    - Generation: SP 800-90A CTR_DRBG; generated only when the module behaves as a TLS Client. As per SP 800-133 Section 7.1, key generation is performed as per the "Direct Generation" of Symmetric Keys which is an Approved key generation method
    - Establishment: N/A
    - Entry: When the module behaves as a TLS Server, the module may receive this secret RSA Key Encapsulated with "Cluster RSA public key for TLS". This is allowed as per FIPS 140-2 IG D.9
    - Output: When the module behaves as a TLS Client, the module may output this value RSA Key Encapsulated with "Public key of an outside entity / server. This is allowed as per FIPS 140-2 IG D.9
    - Storage: Plaintext in RAM
    - Key-to-Entity: Process – TLS
    - Zeroization: On tamper and zeroization service

20. TLS master secret
    - Description: 48-byte master secret
    - Generation: Derived from TLS pre-master secret using SP 800-135 KDF Section 4.2.1 or 4.2.2; allowed method as per FIPS 140-2 IG D.8 Scenario 4
    - Establishment: N/A
    - Entry: N/A
    - Output: N/A
    - Storage: Plaintext in RAM
    - Key-to-Entity: Process – TLS
    - Zeroization: On tamper and zeroization service

21. DRBG Entropy Input String

- Description: 384-bit Entropy Input String output from NDRNG (RDSEED)[8]
- Generation: Internally generated by the NDRNG (RDSEED)
- Establishment: N/A
- Entry: N/A
- Output: N/A
- Storage: Plaintext in RAM
- Key-to-Entity: Process - DRBG
- Zeroization: On tamper and zeroization service

22. DRBG Seed
    - Description: 384-bit DRBG Entropy Input String XOR with personalization string and processed by derivation function
    - Generation: SP 800-90A CTR_DRBG (AES-256) with Derivation Function
    - Establishment: N/A
    - Entry: N/A
    - Output: N/A
    - Storage: Plaintext in RAM
    - Key-to-Entity: Process - DRBG
    - Zeroization: On tamper and zeroization service

23. DRBG internal state
    - Description: Value of V (128-bits) and Key (256-bits) for SP 800-90A CTR_DRBG (AES-256) with Derivation Function
    - Generation: SP 800-90A CTR_DRBG (AES-256) with Derivation Function
    - Establishment: N/A
    - Entry: N/A
    - Output: N/A
    - Storage: Plaintext in RAM
    - Key-to-Entity: Process - DRBG
    - Zeroization: On tamper and zeroization service

24. SP 800-108 KDF internal state

[8] The software module contains an approved CTR_DRBG that is seeded exclusively from one known entropy source (RDSEED) located within the operational environment inside the module's physical boundary but outside the logical boundary.

- Description: 256-bit internal state for SP 800-108 KDF in Feedback Mode (§5.2) with HMAC-SHA-256
- Generation: SP 800-108 KDF in Feedback Mode (§5.2); As per SP 800-133 Section 7.4, key derivation is performed by an Approved KDF which is an Approved key derivation method
- Establishment: N/A
- Entry: N/A
- Output: N/A
- Storage: Plaintext in RAM
- Key-to-Entity: Internal state
- Zeroization: On tamper and zeroization service

25. User password
- Description: String of ASCII characters with a minimum of 8 bytes
- Generation: N/A - Entered by user
- Establishment: N/A
- Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Authentication" service
- Output: N/A
- Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 System key
- Key-to-Entity: This CSP belongs to a specific user; PBKDF2 resulting key is stored along with user object for future authentication
- Zeroization: On tamper and zeroization service

26. Console User password
- Description: String of ASCII characters with a minimum of 8 bytes
- Generation: N/A - Entered by console user
- Establishment: N/A
- Entry: Automatic over trusted path
- Output: N/A
- Storage: Plaintext in RAM, hash value in persistent storage with SHA-512
- Key-to-Entity: This CSP belongs to a console user; SHA-512 resulting hash is stored for future authentication
- Zeroization: On tamper and zeroization service

27. API key
- Description: 64-byte application authentication data

- Generation: SP 800-90A CTR_DRBG
- Establishment: N/A
- Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Authentication" service
- Output: Automatic, encrypted over TLS session (with TLS encryption key (AES)) for "System configuration and management" service
- Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Database Wrapping key
- Key-to-Entity: This belongs to a specific application
- Zeroization: On tamper and zeroization service

28. Bearer token

- Description: 64-byte authentication data
- Generation: SP 800-90A CTR_DRBG
- Establishment: N/A
- Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Authentication" service and invocation of all subsequent authenticated services thereof
- Output: Automatic, encrypted over TLS session (with TLS encryption key (AES)) for "Authentication" service
- Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 System key
- Key-to-Entity: This belongs to a specific authenticated session
- Zeroization: On tamper and zeroization service

11. Appendix B: Public Keys

1. RSA public key for Digital Signatures
   - Description: 2048-bit to 8192-bit RSA key
   - Generation: SP 800-90A CTR_DRBG; this key is used for Digital Signature Verification. As per SP 800-133 Section 6.1, key generation is performed as per FIPS 186-4 which is an Approved key generation method
   - Establishment: N/A
   - Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Import Key" service
   - Output: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Export Key" service if the key was created or imported with export permission
   - Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Database Wrapping key
   - Key-to-Entity: Only authenticated clients can request the use of the key and authorization to access the key is checked. Client making the request must have authorization to use the key
   - Zeroization: N/A

2. RSA public key for Key Encapsulation Operations
   - Description: 2048-bit to 8192-bit RSA key
   - Generation: SP 800-90A CTR_DRBG; this key is used for Key Encapsulation operations. This is an allowed method for key transport as per FIPS 140-2 IG D.9
   - Establishment: N/A
   - Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Import Key" service
   - Output: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Export Key" service if the key was created or imported with export permission
   - Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Database Wrapping key
   - Key-to-Entity: Only authenticated clients can request the use of the key and authorization to access the key is checked. Client making the request must have authorization to use the key
   - Zeroization: N/A

3. ECDSA public key
   - Description: EC key (P-224, P-256, P-384, P-521)
   - Generation: SP 800-90A CTR_DRBG; As per SP 800-133 Section 6.1, key generation is performed as per FIPS 186-4 which is an Approved key generation method
   - Establishment: N/A
   - Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Import Key" service
   - Output: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "Export Key" service if the key was created or imported with export permission
   - Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 Database Wrapping key
   - Key-to-Entity: Only authenticated clients can request the use of the key and authorization to access the key is checked. Client making the request must have authorization to use the key
   - Zeroization: N/A

4. Cluster RSA public key for TLS
   - Description: 2048-bit RSA key
   - Generation: SP 800-90A CTR_DRBG; when the module is a TLS Server, this key is used for RSA Key Encapsulation of the TLS pre-master secret. As per SP 800-133 Section 6.2, key generation is performed as per FIPS 186-4; this is an allowed method as per FIPS 140-2 IG D.9
   - Establishment: N/A
   - Entry: N/A
   - Output: Plaintext during TLS handshake
   - Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 System key
   - Key-to-Entity: This key belongs to the single node cluster
   - Zeroization: N/A

5. Public key of an outside entity/server (CA)
   - Description: 2048-bit to 8192-bit RSA Key used for authentication using digital certificate
   - Generation: N/A - Generated outside of the module
   - Establishment: N/A
   - Entry: Plaintext during "Authentication" service
   - Output: N/A
   - Storage: Plaintext in RAM
   - Key-to-Entity: This key belongs to an outside entity/server (CA)

- Zeroization: N/A

6. RSA Public key of external Application
   - Description: 2048-bit to 8192-bit RSA Key used for authentication using digital certificate
   - Generation: N/A - Generated outside of the module
   - Establishment: N/A
   - Entry: Plaintext during "Authentication" service
   - Output: N/A
   - Storage: Plaintext in RAM
   - Key-to-Entity: This key belongs to an outside entity, external Application
   - Zeroization: N/A

7. User 2FA device public key
   - Description: ECDSA P-256 key with SHA-256
   - Generation: N/A - Generated outside of the module
   - Establishment: N/A
   - Entry: Automatic, encrypted over TLS session (with TLS encryption key (AES)) during "System Configuration and management" service
   - Output: N/A
   - Storage: Plaintext in RAM, encrypted in persistent storage with AES-GCM-256 System key
   - Key-to-Entity: This key belongs to a specific user’s two factor device
   - Zeroization: N/A

12. Appendix C: Acronyms

TERM	DESCRIPTION
AES	Advanced Encryption Standard (FIPS-197)
API	Application Programming Interface
CBC	Cipher Block Chaining
CTR	Counter
CO	Crypto Officer
DRBG	Deterministic Random Bit Generator (SP 800-90Ar1)
EMI/EMC	Electromagnetic Interference/Electromagnetic Compatibility
FIPS	Federal Information Processing Standards
FIPS 140-2 IG	Federal Information Processing Standards 140-2 Implementation Guidance
GCM	Galois/Counter Mode
HMAC	Keyed-hash Message Authentication Code (FIPS 198-1)
IV	Initialization Vector
KAT	Known Answer Test
N/A	Not Applicable
NDRNG	Non-deterministic random number generator
RAM	Random-access Memory
RBG	Random Bit Generator
RNG	Random Number Generator
SDKMS	Self-Defending Key Management Service™
SHA-1	Secure Hash Algorithm 1 (FIPS 180-4)
USB	Universal Serial Bus
VGA	Video Graphics Array

Table 15 Specification of acronyms and their descriptions

13. Appendix D: References

[FIPS 140-2] National Institute of Standards and Technology, Security Requirements for Cryptographic Modules, Federal Information Processing Standards Publication 140-2, May 25, 2001

[FIPS 186-4] National Institute of Standards and Technology, Digital Signature Standard (DSS), Federal Information Processing Standards Publication 186-4, July 2013

[FIPS 197] National Institute of Standards and Technology, Advanced Encryption Standard (AES), Federal Information Processing Standards Publication 197, November 26, 2001

[FIPS 198-1] National Institute of Standards and Technology, The Keyed-Hash Message Authentication Code (HMAC), Federal Information Processing Standards Publication 198-1, July 2008

[SP 800-38A] Dworkin, Morris; Recommendation for Block Cipher Modes of Operation: Methods and Techniques, NIST Special Publication 800-38A, December 2001

[SP 800-38F] Dworkin, Morris; Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, NIST Special Publication 800-38D

[SP 800-56A] Barker, Elaine; Chen, Lily; et al.; Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, NIST Special Publication 800-56A Revision 2, May 2013

[SP 800-90A] Barker, Elaine and Kelsey, John; Recommendation for Random Number Generation Using Deterministic Random Bit Generators, NIST Special Publication 800-90A Revision 1, June 2015

[SP 800-108] Chen, Lily; Recommendation for Key Derivation Using Pseudorandom Functions (Revised), NIST Special Publication 800-108, October 2009

[SP 800-135] Dang, Quynh; Recommendation for Existing Application-Specific Key Derivation Functions, NIST Special Publication 800-135 Revision 1, December 2001