Nuvoton Technology Corporation

Nuvoton NPCT7xx TPM 2.0 Cryptographic Engine

Hardware Version 0x00FC

Firmware Version 7.2.4.1

FIPS 140-3 Non-Proprietary Security Policy

Document Version: 1.4

Last update: 2025-07-29

Prepared by:

atsec information security corporation

4516 Seton Center Parkway, Suite 250

Austin, TX 78759

www.atsec.com

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This document is the non-proprietary FIPS 140-3 Security Policy for the Nuvoton NPCT7xx TPM 2.0 Cryptographic Engine (hereafter referred to as “the module”). It has a one-to-one mapping to the [SP 800-140Br1] starting with section B.2.1 named “General” that maps to section 1 in this document and ending with section B.2.12 named “Mitigation of other attacks” that maps to section 12 in this document.

1.2 Security Levels

The module meets the requirements of FIPS Pub 140-3 overall Security Level 1 with Physical Security section meeting Security Level 3.

Section	Title	Security Level
1	General	1
2	Cryptographic module specification	1
3	Cryptographic module interfaces	1
4	Roles, services, and authentication	1
5	Software/Firmware security	1
6	Operational environment	N/A
7	Physical security	3
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:

The Nuvoton NPCT7xx TPM 2.0 Cryptographic Engine is a hardware cryptographic module (hereafter simply referred to as “the module”) that implements advanced cryptographic algorithms, including symmetric and asymmetric cryptography, as well as key generation and random number generation. The module is a contained within a single-chip embodiment that provides cryptographic services utilized by external applications. The module meets commercial-grade specifications for power, temperature, reliability, shock, and vibrations, and includes chip packaging to meet the physical security requirements at security level 3.

Module Type: Hardware
Module Embodiment: SingleChip

Cryptographic Boundary:

The block diagram below shows the cryptographic boundary of the module, and its interfaces with the operational environment. The components within TOEPP include an RNG block to provide entropy input for the module’s DRBG, Cryptographic Accelerators (SHA, AES, and PKA), ROM containing non-modifiable runtime execution code (CrypLib and Booter), Flash containing modifiable runtime execution code (Bootloader and TPM Library), CPU and RAM for runtime processing. Lastly the GPIO, Power Management and Host Interfaces provide the interfaces to/from the module.

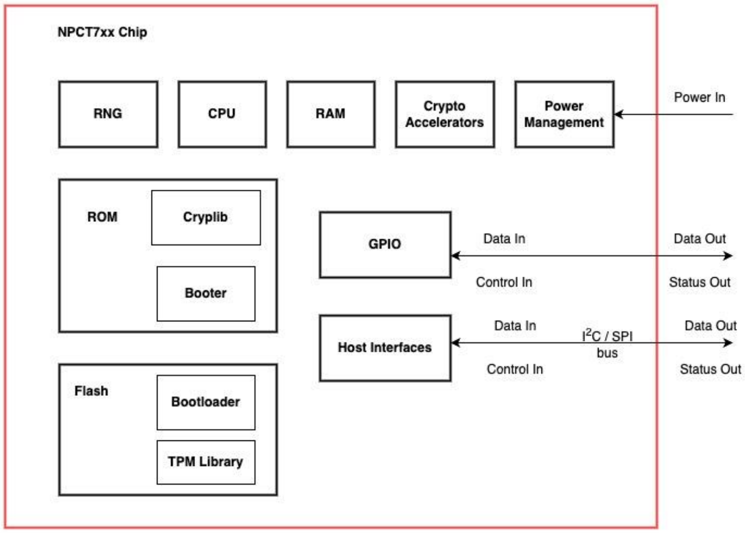


Figure 1: Block Diagram

Tested Operational Environment’s Physical Perimeter (TOEPP)

The TOEPP and cryptographic boundary are one and the same, encompassing the entire physical chip (outlined in red).

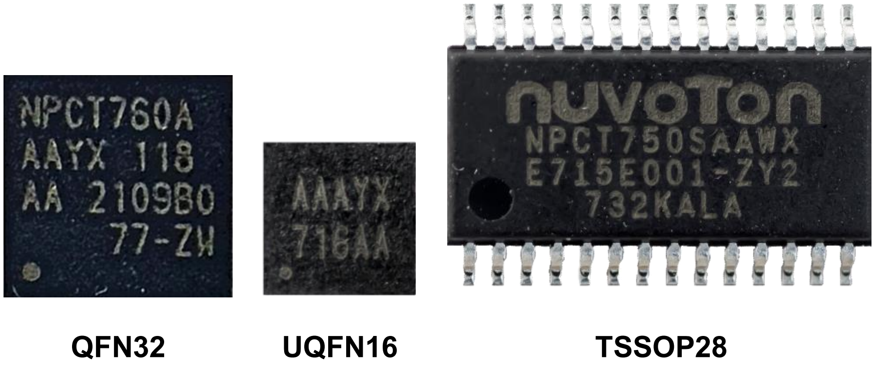
2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

This module is available in three hardware configurations.

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
NPCT7xx embedded in UQFN16 package	0x00FC	7.2.4.1	NPCT7xx CPU	N/A
NPCT7xx embedded in QFN32 package	0x00FC	7.2.4.1	NPCT7xx CPU	N/A
NPCT7xx embedded in TSSOP28 package	0x00FC	7.2.4.1	NPCT7xx CPU	N/A

Table 2: Tested Module Identification – Hardware



QFN32 **UQFN16** **TSSOP28**

Figure 2 – Hardware Module Photographs

2.3 Excluded Components

There are no components within the cryptographic boundary excluded from the FIPS 140-3 requirements.

2.4 Modes of Operation

Modes List and Description:

For some TPM host platforms, it might take too much time to execute all self-tests during power up. Therefore, the TPM supports the following two Approved modes.

Mode Name	Description	Type	Status Indicator
Transient mode	Default mode entered when the TPM powers up and has completed self tests for SHS (SHA-1, SHA2-256, SHA2-384), HMAC, AES, DRBG, KBKDF and KDF algorithms which are used for basic TPM commands.	Approved	Same as section 4.3
Full approved mode of operation	This mode can be entered by either forcing to run the self tests for all algorithms using 'TPM2_SelfTest' command or by explicitly calling service that will require use of algorithms not tested in transient mode. This corresponds to all commands except the ones listed in section 6.5.1.6, of platform TPM profile specification and the command 'TPM2_IncrementalSelfTest'.	Approved	Same as section 4.3

Mode Name	Description	Type	Status Indicator
Non-Approved mode of operation	Automatically entered whenever a non-approved service is invoked.	Non-Approved	Same as section 4.3

Table 3: Modes List and Description

2.5 Algorithms

Approved Algorithms:

The table below lists all approved algorithms used by the module, including specific key strengths employed for approved services, and implemented modes of operation.

Algorithm	CAVP Cert	Properties	Reference
AES-CFB128	A4792	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38A
AES-CTR	A4792	Direction - Encrypt Key Length - 128, 256	SP 800-38A
AES-OFB	A4792	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38A
Conditioning Component Block Cipher Derivation Function SP800-90B	A4792	Key Length - 256	SP 800-90B
Counter DRBG	A4792	Prediction Resistance - No Mode - AES-256 Derivation Function Enabled - No	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-4)	A4792	Curve - P-256, P-384	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A4792	Curve - P-256, P-384	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A4792	Component - No, Yes Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A4792	Component - No Curve - P-256, P-384	FIPS 186-4
HMAC-SHA-1	A4792	Key Length - Key Length: 160-240 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4792	Key Length - Key Length: 160-1024 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4792	Key Length - Key Length: 160-2048 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
KAS-ECC Sp800-56Ar3	A4792	Domain Parameter Generation Methods - P-256, P-384 Function - Key Pair Generation, Partial Validation Scheme - fullUnified - KAS Role - Initiator, Responder Key Length - 1024	SP 800-56A Rev. 3
KAS-ECC-SSC Sp800-56Ar3	A4792	Domain Parameter Generation Methods - P-256, P-384 Scheme - fullUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA OneStep Sp800-56Cr1	A4792	Derived Key Length - 1024 Shared Secret Length - Shared Secret Length: 384-768 Increment 8	SP 800-56C Rev. 2
KDF SP800-108	A4792	KDF Mode - Counter	SP 800-108 Rev. 1
KTS-IFC	A4792	Modulo - 2048, 3072, 4096 Key Generation Methods - rsakpg1-crt Scheme - KTS-OAEP-basic - KAS Role - initiator, responder Key Length - 384	SP 800-56B Rev. 2
RSA KeyGen (FIPS186-4)	A4792	Key Generation Mode - B.3.3 Modulo - 2048, 3072, 4096 Primality Tests - Table C.2 Private Key Format - Chinese Remainder Theorem	FIPS 186-4
RSA SigGen (FIPS186-4)	A4792	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
RSA Signature Primitive (CVL)	A4792	Private Key Format - crt	FIPS 186-4
RSA SigVer (FIPS186-4)	A4792	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
SHA-1	A4792	-	FIPS 180-4
SHA2-256	A4792	-	FIPS 180-4
SHA2-384	A4792	-	FIPS 180-4

Table 4: Approved Algorithms

Vendor-Affirmed Algorithms:

The following table lists all vendor affirmed approved algorithms implemented by the module.

Name	Properties	Implementation	Reference
CKG	Key Type:Symmetric and Asymmetric	N/A	SP800-133, Rev2 Section 4, example 1

Table 5: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

Name	Caveat	Use and Function
ECDSA SigVer Component	Allowed as per IG 2.4.A	Non-Security Related Input Verification (No authentication claimed)

Table 6: Non-Approved, Allowed Algorithms with No Security Claimed

Non-Approved, Not Allowed Algorithms:

The following table list all non-approved algorithms not allowed in the approved mode of operation.

Name	Use and Function
RSA signature generation using SHA-1	Digital signature generation
ECDSA signature generation using SHA-1	Digital signature generation
RSA Key Transport	RSA Key Transport with Non-Approved Padding schemes RSAES-PKCS-v1.5/NULL
CKG	HMAC key generation with Key Size < 112 bits
HMAC	Message Authentication Code using HMAC with Key Size < 112 bits
KAS-ECC-SSC	ECC Shared Secret Calculation with Derived Asymmetric ECC Key

Table 7: Non-Approved, Not Allowed Algorithms

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
AES-CFB128	BC-UnAuth	AES encryption/decryption	Key Size:128, 256 bits Key Strength:128, 256 bits	AES-CFB128: (A4792)
AES-CTR	BC-UnAuth	AES encryption/decryption	Key Size:128, 256 bits Key Strength:128, 256 bits	AES-CTR: (A4792)

Name	Type	Description	Properties	Algorithms
AES-OFB	BC-UnAuth	AES encryption/decryption	Key Size:128, 256 bits Key Strength:128, 256 bits	AES-OFB: (A4792)
CTR_DRBG	DRBG	Deterministic random bit generation	Mode:AES-256 Key Size:256 bits Key Strength:256 bits Prediction Resistance:No Supports Reseed:Yes Derivation Function Enabled:No	Counter DRBG: (A4792)
ECDSA KeyGen	AsymKeyPair-KeyGen CKG	ECC key generation	Curves and Key Strength:P-256, P-384 with 128 and 192 bits of key strength	ECDSA KeyGen (FIPS186-4): (A4792)
ECDSA KeyVer	AsymKeyPair-KeyVer	ECC public key validation	Curves:P-256, P-384 Key Strength:128 and 192 bits	ECDSA KeyVer (FIPS186-4): (A4792)
ECDSA SigGen	DigSig-SigGen	ECC signature generation	Curves:P-256, P-384 Hash Algorithm:SHA2-256, SHA2-384	ECDSA SigGen (FIPS186-4): (A4792)
ECDSA SigVer	DigSig-SigVer	ECC signature verification	Curves:P-256, P-384 Hash Algorithm:SHA-1, SHA2-256, SHA2-384	ECDSA SigVer (FIPS186-4): (A4792)
ECDSA SigGen Component	DigSig-SigGen	ECC signature generation component	Curves:P-256, P-384	ECDSA SigGen (FIPS186-4): (A4792)
HMAC	MAC	Message Authentication Code using HMAC	HMAC-SHA-1:Key Sizes 160 to 240 with 128 bits of security strength HMAC-SHA-256:Key Sizes 160 to 1024 with 256 bits of security strength HMAC-SHA-384:Key Sizes 160 to 2048 with 256 bits of security strength	HMAC-SHA-1: (A4792) HMAC-SHA2-256: (A4792) HMAC-SHA2-384: (A4792)
KAS-ECC	KAS-Full	ECC key agreement	Key Agreement Schemes:Full Unified, One Pass DH KDF Methods:onestepkdf (SHA2-256/SHA2-	KAS-ECC Sp800-56Ar3: (A4792)

Name	Type	Description	Properties	Algorithms
			384) Caveat:Key establishment methodology provides 128 or 192 bits of security strength Compliance:IG D.F	
KAS-ECC-SSC	KAS-SSC	ECC shared secret calculation	Key Agreement Schemes:Full Unified, One Pass DH Caveat:Key establishment methodology provides 128 or 192 bits of security strength	KAS-ECC-SSC Sp800-56Ar3: (A4792)
KDA	KAS-56CKDF	Symmetric key derivation (KDA)	Auxiliary Function Methods:SHA2-256, SHA2-384 derived key size and strength:1024 bits with 256 bits of key strength	KDA OneStep Sp800-56Cr1: (A4792)
KBKDF	KBKDF	Symmetric key derivation (KBKDF)	KDF Mode:Counter HMAC-SHA-1 Key Size and Strength:160 and 384 bit key with 128 bits of security strength HMAC-SHA-256 Key Size and Strength Key Strength:160 and 384 bit key with 256 bits of security strength HMAC-SHA-384 Key Size and strength:160 and 384 bit key with 256 bits of security strength	KDF SP800-108: (A4792)
KTS RSA	KTS-Wrap	RSA key transport	Scheme:KTS-OAEP-basic Key Transport Method:SHA2-256, SHA2-384 Key Generation Methods:rsakpg1-crt Caveat:Key establishment methodology provides between 112 and 150 bits of security strength Compliance:IG D.G	KTS-IFC: (A4792)

Name	Type	Description	Properties	Algorithms
RSA KeyGen	AsymKeyPair-KeyGen	RSA key generation	Key Generation Mode:B.3.3 Key Size and Strength :2048-, 3072- or 4096-bit modulus with 112, 128, 150 bits of key strength	RSA KeyGen (FIPS186-4): (A4792)
RSA SigGen	DigSig-SigGen	RSA signature generation	Signature Type:PKCS 1.5, PKCSPSS Hash Pair:SHA2-256, SHA2-384 Key Size:2048, 3072, 4096 bits	RSA SigGen (FIPS186-4): (A4792)
RSA SigGen Primitive	DigSig-SigGen	RSA signature generation primitive	Key Size:2048, 3072, 4096 bits	RSA Signature Primitive: (A4792)
RSA SigVer	DigSig-SigVer	RSA signature verification	Signature Type:PKCS 1.5, PKCSPSS Hash Pair:SHA2-256, SHA2-384 Key Size:2048, 3072, 4096 bits	RSA SigVer (FIPS186-4): (A4792)
SHA	SHA	Message digest		SHA-1: (A4792) SHA2-256: (A4792) SHA2-384: (A4792)
AES key generation	CKG	AES key generation	Key Size and Strength:128, 256 bits with 128 or 256 bits of key strength	Counter DRBG: (A4792)
HMAC key generation	CKG	HMAC key generation	Key Size and Strength:160, 256, 384 bits with 128 or 256 bits of key strength	Counter DRBG: (A4792)
KTS (AES + HMAC) key wrapping	KTS-Wrap	Symmetric key wrapping	Caveat:Key establishment methodology provides 128 or 256 bits of security strength	AES-CFB128: (A4792) HMAC-SHA-1: (A4792) HMAC-SHA2-256: (A4792) HMAC-SHA2-384: (A4792)
KTS (AES + HMAC) key unwrapping	KTS-Wrap	Symmetric key unwrapping	Caveat:Key establishment methodology provides 128 or 256 bits of security strength	AES-CFB128: (A4792) HMAC-SHA-1: (A4792) HMAC-SHA2-256: (A4792)

Name	Type	Description	Properties	Algorithms
				HMAC-SHA2-384: (A4792)
Entropy Source	ENT-ESV	Physical entropy source	Conditioning Component:Block Cipher DF Sample Size:384 bits Entropy Per Sample:384 bits	Conditioning Component Block Cipher Derivation Function SP800-90B: (A4792)

Table 8: Security Function Implementations

2.7 Algorithm Specific Information

Compliance to SP 800-56ARev3 assurances

For KAS-ECC, the module satisfies IG D.F Scenario 2 path (2) (i.e., tested compliance with Full Unified and One Pass DH key agreement schemes followed by the derivation of the key as shown in Section 5.8 of SP 800-56Arev3). The key derivation function complies to SP 800-56C rev2 (i.e, One-Step KDF). Furthermore, the module obtained the appropriate assurances, as required in Sections 5.6.2 of SP 800-56A rev3.

5.6.2.1 Assurances Required by a Key Pair Owner:

The key generation implemented by the module is CAVP validated. The entity using the module, must use the module's key generation service to generate the ECDH keys. The module will perform a pairwise consistency check upon generating ECDH keys.

5.6.2.2 Assurances Required by a Public Key Recipient Assurance of public-key validity: The module makes used of approved EC curves listed in SP 800-140D and performs a successful public-key validation of the received public key.

5.6.2.2.3 Recipient’s assurance of owner’s possession of private key can be met via the use of a Trusted Third party that requires the key confirmation procedure. Both of which are handled by the entity outside of the module that requested the ECDH Key Agreement service from the module. That is, such checks are out of the module's scope.

5.6.2.3 Public Key Validation Routines: The module performs the required public key validation before initiating the handshake.

Compliance to SP 800-56BRev3 assurances

For KTS RSA, the tester verified the implementation satisfies IG D.G by employing an approved RSA-based key transport scheme as specified in SP 800-56Brev2.

The following summary of assurances, as defined in Sections 5 and 6 of SP 800-56Brev2:

Section 5.1 – The module uses an approved hash function (SHS, Cert. #A4792) for mask generation during RSA-OEAP encryption.

Section 5.2 and Section 5.6 – N/A, The module does not implement key confirmation.

Section 5.3 - The module uses an approved random bit generator (CTR_DRBG, Cert. #A4792) when generating random values.

Section 5.4 and Section 5.5 – N/A, The module does not implement a key agreement scheme (i.e., KAS1).

For additional assurances found in its Section 6 (specifically SP 800-56Brev2 Section 6.4 Required Assurances):

1) The entity requesting the RSA key unwrapping (decapsulation) service from the module, shall only use an RSA private key that was generated by an active FIPS validated module that implements FIPS 186-4 compliant RSA key generation service and performs the key pair validity and the pairwise consistency as stated in section 6.4.1.1 of the SP 800-56Brev2. Additionally, the entity shall renew these assurances over time by using any method described in section 6.4.1.5 of the SP 800-56Brev2.

2) For use of an RSA key wrapping (encapsulation) service in the context of key transport per IG D.G,

a) the entity using the module, shall verify the validity of the peer's public key using the public key validation service of the module.

b) the entity using the module, shall confirm the peer's possession of private key by using any method specified in section 6.4.2.3 of the SP 800-56Brev2.

Only after the above assurances are successfully met, shall the entity use the peer's public key to perform the RSA key wrapping (encapsulation) service of the module."

2.8 RBG and Entropy

Cert Number	Vendor Name
E18	Nuvoton

Table 9: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Nuvoton NPCT7xx TPM 2.0 Cryptographic Engine	Physical	Nuvoton NPCT7xx TPM 2.0 Cryptographic Engine on Nuvoton NPCT7xx (firmware version 7.2.4.1)	384 bits	384 bits	Vetted Conditioning Component: Block Cipher Derivation Function Cert# A4792

Table 10: Entropy Sources

Entropy Information: The SP 800-90B entropy source consists of a noise source which feeds 1024-bit samples to a block cipher vetted conditioning component. The final output from the conditioning component provides 384-bits of full entropy.

DRBG Information: The module implements an approved SP 800-90Ar1 Deterministic Random Bit Generator in the form of CTR_DRBG. The CTR_DRBG is provided with 384-bits of entropy input from the SP 800-90B compliant entropy source. The DRBG is used internally by the module to generate symmetric keys, seeds for asymmetric key pairs and random numbers for security functions.

2.9 Key Generation

This module implements vendor affirmed Cryptographic Key Generation (CKG) for AES, HMAC, RSA, ECDSA and EC Diffie-Hellman keys, compliant to SP 800-133rev2 and IG D.H.

When generating RSA and ECDSA and EC Diffie-Hellman asymmetric key pairs, the seed used for asymmetric key generation is obtained directly from the module's approved SP 800-90rev1 DRBG (CTR_DRBG) specified in SP 800-133 Rev2 Section 4, example 1. This is followed by an asymmetric key generation method compliant with FIPS 186-4 (and SP 800-56A Rev 3 for ECDH key pairs).

Symmetric keys (AES and HMAC) are also generated directly from the module's approved CTR_DRBG as following Section 4, example 1 of SP 800-133rev2.

2.10 Key Establishment

The module provides an approved [SP800-56Arev3] EC Diffie-Hellman Key Agreement Scheme. The key agreement scheme is compliant with IG D.F scenario 2 path (2). The CAVP testing was performed end-to-end, using the Full Unified and One Pass DH Models with approved domain parameters (i.e., P-256 and P-384). Per FIPS 140-3 IG D.B, the curves provide 128 or 192 bits of security strength.

The module provides key derivation services using SP 800-108 KBKDF and SP 800-56C One-Step KDF with key sizes 160, 256, 384 bits. Per FIPS 140-3 IG D.B, the key sizes provide 160-256 bits of security strength.

The module provides SP 800-56B rev2 Key Transport using KTS-OAEP-basic. The implementation supports 2048-, 3072-, or 4096-bits modulus size, with both key encapsulation and un-encapsulation supported. The module does not implement key confirmation. Per FIPS 140-3 IG D.B, the key sizes provide 112, 128, or 150 bits of security strength.

The module also implements AES-CFB128 combined with HMAC as approved key wrapping method compliant SP 800-38F. Per FIPS 140-3 IG D.B, the key sizes provide 128 or 256 bits of security strength.

2.11 Industry Protocols

The cryptographic module does not implement any relevant industry protocols.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
SPI Bus	Data Input Data Output Control Input Status Output	Data provided to the chip as part of the data processing commands; Data output by the chip a part of the data processing commands; Control Input commands issued to the chip; Status data output by the chip
I2C Bus	Data Input Data Output Control Input Control Output	Data provided to the chip as part of the data processing commands; Data output by the chip a part of the data processing commands; Control Input commands issued to the chip; Status data output by the chip
PP pin	Data Input Control Input	Data provided to the chip as part of the data processing commands; Control Input commands issued to the chip
Power	Power	Power interface of the chip

Table 11: Ports and Interfaces

The module does not implement any control output interface.

The logical interfaces are the API through which applications request services. The ports and interfaces are shown in the following table.

All data output via data output interface is inhibited when the module is performing pre-operational test or zeroization or when the module enters error state.

4 Roles, Services, and Authentication

4.1 Authentication Methods

N/A for this module.

The module does not support role authentication.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Object Administrator	Role	Crypto Officer	None
Object User	Role	User	None
Duplicate	Role	Crypto Officer	None

Table 12: Roles

4.3 Approved Services

The table below lists the approved services. For each service, the table lists the associated cryptographic algorithm(s), the role to perform the service, the cryptographic keys or SSPs involved, and their access type(s). The following convention is used to specify access rights for SSPs:

G = **Generate**: The module generates or derives the SSP.

R = **Read**: The SSP is read from the module (e.g., the SSP is output).

W = **Write**: The SSP is updated, imported, or written to the module.

E = **Execute**: The module uses the SSP in performing a cryptographic operation.

Z = **Zeroise**: The module zeroizes the SSP.

N/A: The calling application does not access any SSP or key during its operation.

Details on the approved cryptographic algorithms, can be found in the SFI table. The module implements a FIPS 140-3 service indicator function that outputs its value. This value corresponds to the three categories defined in IG 2.4.C. **Non-security relevant** services are set to '00', **approved services** are set to '01' and **non-approved services** are set to '10'.

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_Startup	Used to initiate a startup process, where the TPM state is either reset or loaded from a saved state.	'00'	Startup Type	N/A	Entropy Source	Unauthenticated - nullSeed: Z - nullProof: Z - platformAuth: Z - platformPolic

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						y: Z - endorsementPolicy: Z - ownerPolicy: Z - lockoutPolicy: Z - Asymmetric Signing Keys (authValue): Z - Asymmetric Signing Keys (seed value): Z - Asymmetric Signing Keys (sensitive data): Z - Asymmetric Signing Keys (authPolicy): Z - Asymmetric Signing Keys (public data): Z - Asymmetric Encryption Keys (authValue): Z - Asymmetric Encryption Keys (seedValue): Z - Asymmetric Encryption Keys (sensitive data): Z - Asymmetric Encryption Keys (authPolicy): Z - Asymmetric Encryption Keys (public data): Z - Symmetric

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Encryption Keys (authValue): Z - Symmetric Encryption Keys (seedValue): Z - Symmetric Encryption Keys (sensitive data): Z - Symmetric Signing Keys (authValue): Z - Symmetric Signing Keys (seedValue): Z - Symmetric Signing Keys (sensitive data): Z - Session (sessionKey): Z - DRBG state: Z - DRBG Entropy Input: G,Z - Transient DRBG state: Z
TPM2_Shutdown	Used to prepare the TPM for a power cycle.	'00'	Shutdown Type	N/A	None	Unauthenticated
TPM2_IncrementalSelfTest	Perform Self-Test of selected algorithms.	'01'	List of algorithms to be tested	To do list of the selected algorithms All algorithms mentioned in Table 22	AES-CFB128 AES-CTR AES-OFB CTR_DRBG ECDSA KeyGen ECDSA KeyVer ECDSA	Unauthenticated - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					SigGen ECDSA SigVer HMAC KAS-ECC KAS-ECC-SSC KDA KBKDF KTS RSA RSA KeyGen RSA SigGen RSA SigGen Primitive RSA SigVer SHA	data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Asymmetric Encryption Keys (authValue): E - Asymmetric Encryption Keys (seedValue): E - Asymmetric Encryption Keys (sensitive data): E - Asymmetric Encryption Keys (authPolicy): E - Asymmetric Encryption Keys (public data): E - Symmetric Encryption Keys (authValue): E - Symmetric Encryption Keys (seedValue): E - Symmetric Encryption Keys (sensitive data): E - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue):

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						E - Symmetric Signing Keys (sensitive data): E
TPM2_SelfTest	Perform Self-Test of all functions or only those that have not previously been tested.	'01'	Choose whether to perform the test everything (fullTest = YES) or only the untested functions (fullTest = NO)	N/A	AES-CFB128 AES-CTR AES-OFB CTR_DRBG ECDSA KeyGen ECDSA KeyVer ECDSA SigGen ECDSA SigVer HMAC KAS-ECC KAS-ECC-SSC KDA KBKDF KTS RSA RSA KeyGen RSA SigGen RSA SigGen Primitive RSA SigVer SHA	Unauthenticated - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Asymmetric Encryption Keys (authValue): E - Asymmetric Encryption Keys (seedValue): E - Asymmetric Encryption Keys (sensitive data): E - Asymmetric Encryption Keys (authPolicy): E - Asymmetric Encryption Keys (public data): E - Symmetric Encryption Keys

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						(authValue): E - Symmetric Encryption Keys (seedValue): E - Symmetric Encryption Keys (sensitive data): E - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_GetTestResult	Returns manufacturer-specific information regarding the results of a self-test and an indication of the test status.	'00'	N/A	test result data (manufacturer-specific information), test result	None	Unauthenticated
TPM2_StartAuthSession	Start authorization session.	'01'	Session Parameters: session Type, encryption algorithm, key size, hash algorithm	tpmKey, authValue, nonce size, encrypted salt, session Type, encryption algorithm, key size, hash algorithm	CTR_DRBG KAS-ECC KBKDF KTS RSA	Unauthenticated - platformAuth: E - Asymmetric Encryption Keys (authValue): E - Asymmetric Encryption Keys (seedValue): E - Asymmetric Encryption Keys (sensitive data): E - Asymmetric Encryption

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Keys (authPolicy): E - Asymmetric Encryption Keys (public data): E - Ephemeral Key Agreement Keys: G,E - Session (salt): G,E - Session (sessionKey): G,E - Session (symKey): G,E
TPM2_PolicyRestart	Allows a policy authorization session to be returned to its initial state.	'00'	session handle	N/A	None	Unauthenticated - Session (sessionKey): E - Session (symKey): E Object User - Session (sessionKey): E - Session (symKey): E
TPM2_Create	Creation of an ordinary object.	'01'	Parent handle, sensitive data, public template, outside info, creationPCR	private portion, public portion, creation data, hash value, creation ticket (see TPM2_CertifyCreation)	CTR_DRBG Entropy Source ECDSA KeyGen HMAC RSA KeyGen SHA KBKDF AES key generation HMAC key generation KTS (AES +	Object User - Asymmetric Signing Keys (authValue): G - Asymmetric Signing Keys (seed value): G - Asymmetric Signing Keys (sensitive data): G - Asymmetric Signing Keys (authPolicy): G - Asymmetric Signing Keys (public data): G

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					HMAC) key unwrapping KTS (AES + HMAC) key wrapping	- Asymmetric Encryption Keys (authValue): G - Asymmetric Encryption Keys (seedValue): G - Asymmetric Encryption Keys (sensitive data): G - Asymmetric Encryption Keys (authPolicy): G - Asymmetric Encryption Keys (public data): G - Symmetric Encryption Keys (authValue): G - Symmetric Encryption Keys (seedValue): G - Symmetric Encryption Keys (sensitive data): G - Symmetric Signing Keys (authValue): G - Symmetric Signing Keys (seedValue): G - Symmetric Signing Keys (sensitive data): G

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_Load	Loading an protected object.	'01'	Parent handle, private portion, public portion	Object handle, name of the loaded object	HMAC KAS-ECC-SSC KBKDF SHA KTS (AES + HMAC) key unwrapping	Object User - Asymmetric Signing Keys (authValue): W - Asymmetric Signing Keys (seed value): W - Asymmetric Signing Keys (sensitive data): W - Asymmetric Signing Keys (authPolicy): W - Asymmetric Signing Keys (public data): W - Asymmetric Encryption Keys (authValue): W - Asymmetric Encryption Keys (seedValue): W - Asymmetric Encryption Keys (sensitive data): W - Asymmetric Encryption Keys (authPolicy): W - Asymmetric Encryption Keys (public data): W - Symmetric Encryption Keys (authValue): W - Symmetric Encryption Keys (seedValue):

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						W - Symmetric Encryption Keys (sensitive data): W - Symmetric Signing Keys (authValue): W - Symmetric Signing Keys (seedValue): W - Symmetric Signing Keys (sensitive data): W - Object Ephemeral Keys (symKey): E - Object Ephemeral Keys (hmacKey): E
TPM2_LoadExternal	Loading an external object.	'01'	Private portion, public portion, associated hierarchy	Object handle, name of the loaded object	HMAC KAS-ECC-SSC SHA	Unauthenticated - Asymmetric Signing Keys (authValue): W - Asymmetric Signing Keys (seed value): W - Asymmetric Signing Keys (sensitive data): W - Asymmetric Signing Keys (authPolicy): W - Asymmetric Signing Keys (public data): W - Asymmetric Encryption Keys (authValue): W - Asymmetric

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Encryption Keys (seedValue): W - Asymmetric Encryption Keys (sensitive data): W - Asymmetric Encryption Keys (authPolicy): W - Asymmetric Encryption Keys (public data): W - Symmetric Encryption Keys (authValue): W - Symmetric Encryption Keys (seedValue): W - Symmetric Encryption Keys (sensitive data): W - Symmetric Signing Keys (authValue): W - Symmetric Signing Keys (seedValue): W - Symmetric Signing Keys (sensitive data): W
TPM2_ReadPublic	Allows access to the public area of a loaded object.	'00'	object handle	public area of object, name of object, qualified name of object	SHA	Unauthenticated - Asymmetric Signing Keys (authValue): R - Asymmetric Signing Keys (seed value):

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						R - Asymmetric Signing Keys (sensitive data): R - Asymmetric Signing Keys (authPolicy): R - Asymmetric Signing Keys (public data): R - Asymmetric Encryption Keys (authValue): R - Asymmetric Encryption Keys (seedValue): R - Asymmetric Encryption Keys (sensitive data): R - Asymmetric Encryption Keys (authPolicy): R - Asymmetric Encryption Keys (public data): R - Symmetric Encryption Keys (authValue): R - Symmetric Encryption Keys (seedValue): R - Symmetric Encryption Keys (sensitive data): R - Symmetric Signing Keys (authValue):

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						R - Symmetric Signing Keys (seedValue): R - Symmetric Signing Keys (sensitive data): R
TPM2_ActivateCredential	Decrypts an object credential.	'01'	active handle, key handle, credential blob, secret	decrypted certificate information	KTS RSA KAS-ECC KBKDF KTS (AES + HMAC) key unwrapping	Object Administrator - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Credential Ephemeral Keys (symKey): R,E - Credential Ephemeral Keys (hmacKey): R,E - Ephemeral Key Agreement Keys: E Object User - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Credential Ephemeral Keys (symKey): R,E - Credential Ephemeral Keys (hmacKey): R,E - Ephemeral Key Agreement Keys: E
TPM2_MakeCredential	Encrypts object credential.	'01'	Object handle, credential, object name	encrypted secret, credentialBlob	CTR_DRBG KTS RSA ECDSA KeyGen KAS-ECC KTS (AES + HMAC) key wrapping KBKDF	Unauthenticated - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Credential Ephemeral Keys (symKey): R,E - Credential

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Ephemeral Keys (hmacKey): R,E - Ephemeral Key Agreement Keys: E
TPM2_Unseal	Returns the data in a loaded Sealed Data Object.	'00'	item handle	unsealed data	None	Object User
TPM2_ObjectChangeAuth	Change the authorization secret of an object.	'01'	Object handle, parent handle, new authValue	private area containing new authValue	CTR_DRBG KBKDF SHA KTS (AES + HMAC) key unwrapping KTS (AES + HMAC) key wrapping	Object Administrator - Object Ephemeral Keys (symKey): R,W,E - Object Ephemeral Keys (hmacKey): R,W,E - Asymmetric Signing Keys (authValue): R,W - Asymmetric Signing Keys (seed value): R,W - Asymmetric Signing Keys (sensitive data): R,W - Asymmetric Signing Keys (authPolicy): R,W - Asymmetric Signing Keys (public data): R,W - Asymmetric Encryption Keys (authValue): R,W - Asymmetric Encryption Keys (seedValue): R,W

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- Asymmetric Encryption Keys (sensitive data): R,W - Asymmetric Encryption Keys (authPolicy): R,W - Asymmetric Encryption Keys (public data): R,W - Symmetric Encryption Keys (authValue): R,W - Symmetric Encryption Keys (seedValue): R,W - Symmetric Encryption Keys (sensitive data): R,W - Symmetric Signing Keys (authValue): R,W - Symmetric Signing Keys (seedValue): R,W - Symmetric Signing Keys (sensitive data): R,W
TPM2_CreateLoaded	Creation and loading of an ordinary or a derived object.	'01'	Parent handle, private portion, public key portion	Object handle, private portion, public portion, Name of the loaded object	CTR_DRBG Entropy Source ECDSA KeyGen HMAC KBKDF RSA KeyGen SHA AES key	Object User - ppSeed: E - epSeed: E - spSeed: E - nullSeed: E - platformAuth: E - endorsementAuth: E - ownerAuth: E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					generation HMAC key generation KTS (AES + HMAC) key wrapping	- Object Ephemeral Keys (symKey): G,W,E - Object Ephemeral Keys (hmacKey): G,W,E - Asymmetric Signing Keys (authValue): G,W - Asymmetric Signing Keys (seed value): G,W - Asymmetric Signing Keys (sensitive data): G,W - Asymmetric Signing Keys (authPolicy): G,W - Asymmetric Signing Keys (public data): G,W - Asymmetric Encryption Keys (authValue): G,W - Asymmetric Encryption Keys (seedValue): G,W - Asymmetric Encryption Keys (sensitive data): G,W - Asymmetric Encryption Keys (authPolicy): G,W - Asymmetric Encryption Keys (public data): G,W - Symmetric

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Encryption Keys (authValue): G,W - Symmetric Encryption Keys (seedValue): G,W - Symmetric Encryption Keys (sensitive data): G,W - Symmetric Signing Keys (authValue): G,W - Symmetric Signing Keys (seedValue): G,W - Symmetric Signing Keys (sensitive data): G,W
TPM2_Duplicate	Duplicates a loaded object to a new parent object.	'01'	Object handle, new parent handle, encryption key, symmetric algorithm for key wrapping	Encrypted key, duplicate object, seed value (asymetrically encrypted)	CTR_DRBG ECDSA KeyGen KAS-ECC KBKDF KTS RSA SHA KTS (AES + HMAC) key wrapping	Duplicate - Asymmetric Encryption Keys (authValue): R,E - Asymmetric Encryption Keys (seedValue): R,E - Asymmetric Encryption Keys (sensitive data): R,E - Asymmetric Encryption Keys (authPolicy): R,E - Asymmetric Encryption Keys (public data): R,E - Duplication Ephemeral Keys

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						(symKey): E - Duplication Ephemeral Keys (hmacKey): E - Ephemeral Key Agreement Keys: E - Asymmetric Signing Keys (authValue): R - Asymmetric Signing Keys (seed value): R - Asymmetric Signing Keys (sensitive data): R - Asymmetric Signing Keys (authPolicy): R - Asymmetric Signing Keys (public data): R - Symmetric Encryption Keys (authValue): R - Symmetric Encryption Keys (seedValue): R - Symmetric Encryption Keys (sensitive data): R - Symmetric Signing Keys (authValue): R - Symmetric Signing Keys (seedValue): R - Symmetric Signing Keys

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						(sensitive data): R
TPM2_Rewrap	Rewraps a duplicated object with a new parent key.	'01'	Old parent, new parent, duplicate object, name of object to be wrapped, seed value for the symmetric key and HMAC key	New duplicate object, seed for new object (encrypted with new parent's asymmetric key)	CTR_DRBG ECDSA KeyGen KAS-ECC KBKDF KTS RSA KTS (AES + HMAC) key unwrapping KTS (AES + HMAC) key wrapping	Object User - Duplication Ephemeral Keys (symKey): E - Duplication Ephemeral Keys (hmacKey): E - Ephemeral Key Agreement Keys: E - Asymmetric Signing Keys (authValue): R - Asymmetric Signing Keys (seed value): R - Asymmetric Signing Keys (sensitive data): R - Asymmetric Signing Keys (authPolicy): R - Asymmetric Signing Keys (public data): R - Asymmetric Encryption Keys (authValue): R - Asymmetric Encryption Keys (seedValue): R - Asymmetric Encryption Keys (sensitive data): R - Asymmetric Encryption Keys (authPolicy):

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						R - Asymmetric Encryption Keys (public data): R - Symmetric Encryption Keys (authValue): R - Symmetric Encryption Keys (seedValue): R - Symmetric Encryption Keys (sensitive data): R - Symmetric Signing Keys (authValue): R - Symmetric Signing Keys (seedValue): R - Symmetric Signing Keys (sensitive data): R
TPM2_Import	Import a duplicated object to be next loaded inside the TPM.	'01'	Parent handle, encryption key, public area of object to be imported, encrypted duplicate object, duplicate object seed, algorithm for key wrapping	Private portion encrypted with the symmetric key of parent handle	CTR_DRBG HMAC KAS-ECC KBKDF KTS RSA SHA KTS (AES + HMAC) key unwrapping	Object User - Asymmetric Encryption Keys (authValue): R,E - Asymmetric Encryption Keys (seedValue): R,E - Asymmetric Encryption Keys (sensitive data): R,E - Asymmetric Encryption Keys (authPolicy): R,E - Asymmetric

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Encryption Keys (public data): R,E - Duplication Ephemeral Keys (symKey): E - Duplication Ephemeral Keys (innerSymKey): E - Ephemeral Key Agreement Keys: E - Asymmetric Signing Keys (authValue): R - Asymmetric Signing Keys (seed value): R - Asymmetric Signing Keys (sensitive data): R - Asymmetric Signing Keys (authPolicy): R - Asymmetric Signing Keys (public data): R - Symmetric Encryption Keys (authValue): R - Symmetric Encryption Keys (seedValue): R - Symmetric Encryption Keys (sensitive data): R - Symmetric Signing Keys (authValue): R

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- Symmetric Signing Keys (seedValue): R - Symmetric Signing Keys (sensitive data): R
TPM2_RSA_Encrypt	RSA Encryption.	'01'	Key handle, message, padding scheme, label	Cipher text	KTS RSA	Unauthenticated - Asymmetric Encryption Keys (authValue): E - Asymmetric Encryption Keys (seedValue): E - Asymmetric Encryption Keys (sensitive data): E - Asymmetric Encryption Keys (authPolicy): E - Asymmetric Encryption Keys (public data): E
TPM2_RSA_Decrypt	RSA Decryption.	'01'	Key handle, cipher text, scheme, label	Plaintext	KTS RSA	Unauthenticated - Asymmetric Encryption Keys (authValue): E - Asymmetric Encryption Keys (seedValue): E - Asymmetric Encryption Keys (sensitive data): E - Asymmetric Encryption

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Keys (authPolicy): E - Asymmetric Encryption Keys (public data): E
TPM2_ECDH_KeyGen	Ephemeral key pair generation and Shared Secret Calculation.	'01'	Key handle	zPoint, public point	ECDSA KeyGen KAS-ECC-SSC	Unauthenticated - Ephemeral User ECC Keys: G
TPM2_ECDH_Zgen	Shared Secret Calculation.	'01'	Key handle, public point	Output point	KAS-ECC-SSC	Object User - Ephemeral User ECC Keys: G
TPM2_ECC_Parameters	Returns the parameters of an ECC curve identified by its TCG-assigned curveID.	'00'	Curve id	ECC parameters for selected curve	None	Unauthenticated
TPM2_EncryptDecrypt	Symmetric encryption or decryption of user data.	'01'	Key handle, encrypt/decrypt, mode, IV, ciphertext/plaintext	Plaintext/ciphertext, IV	AES-CFB128 AES-CTR AES-OFB	Object User - Symmetric Encryption Keys (authValue): E - Symmetric Encryption Keys (seedValue): E - Symmetric Encryption Keys (sensitive data): E
TPM2_EncryptDecrypt2	Symmetric encryption or decryption of user data.	'01'	Key handle, encrypt/decrypt, mode, IV, ciphertext/plaintext	Plaintext/ciphertext, IV	AES-CFB128 AES-CTR AES-OFB	Object User - Symmetric Encryption Keys (authValue): E - Symmetric Encryption Keys (seedValue): E - Symmetric

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Encryption Keys (sensitive data): E
TPM2_Hash	Performs a hash operation on user data.	'01'	Data, hash algorithm, hierarchy	Digest, validation ticket	HMAC SHA	Unauthenticated - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_HMAC	Performs a HMAC operation on user data.	'01'	Key handle, HMAC data, hash algorithm	Returned HMAC	HMAC	Object User - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_GetRandom	Random number generation.	'01'	Number of bytes requested	Random bytes	CTR_DRBG	Unauthenticated - DRBG state: G,E - DRBG Entropy Input: E - Transient DRBG state: G,E
TPM2_StirRandom	Reseed random number generator.	'01'	Key handle, auth value, hash algorithm	Sequence handle	CTR_DRBG Entropy Source	Unauthenticated - DRBG Entropy Input: G
TPM2_HMAC_Start	HMAC session start	'01'	Key handle, auth value,	Sequence handle	HMAC	Object User - Symmetric Signing Keys

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			algorithms to be used			(authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_HashSequenceStart	Hash session start	'01'	Auth value, hash algorithm	Sequence handle	SHA	Unauthenticated - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_SequenceUpdate	Sequence update	'01'	Sequence handle, data to add to hash	N/A	HMAC SHA	Unauthenticated - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_SequenceComplete	Sequence complete	'01'	Sequence handle, data, hierarchy	Returned HMAC or message digest, ticket	HMAC SHA	Unauthenticated - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						(sensitive data): E
TPM2_EventSequenceComplete	Event sequence complete	'01'	Data	List of digests	HMAC SHA	Object User - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_CertifyCreation	Proves the association between an object and its creation data	'01'	Sign handle, object handle, qualifying data, creation hash, scheme, creation ticket	Certify info, signature	ECDSA SigGen HMAC KBKDF RSA SigGen SHA	Object Administrator - shProof: E - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E Object User - shProof: E - Asymmetric

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_Quote	Quotes PCR values	'01'	sign handle, qualifying data, scheme, PCR selection	quoted information, signature	ECDSA SigGen HMAC KBKDF RSA SigGen SHA	Object User - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Symmetric Signing Keys

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						(authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_GetSessionAuditDigest	Returns a digital signature of the audit session digest	'01'	Privacy administrator handle, sign handle, session handle, qualifying data, scheme	Audit info, signature	ECDSA SigGen HMAC KBKDF RSA SigGen SHA	Object User - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E - Session (sessionKey): E - Session (symKey): E
TPM2_GetCommandAuditDigest	Returns the current value of the command audit digest	'01'	Privacy administrator handle, sign handle, qualifying	Audit info, signature	ECDSA SigGen HMAC KBKDF RSA	Object User - Asymmetric Signing Keys (authValue): E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			data, scheme		SigGen SHA	- Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_GetTime	Returns the current values of Time and Clock	'01'	Privacy administrator handle, sign handle, qualifying data, scheme	Time info, signature	ECDSA SigGen HMAC KBKDF RSA SigGen SHA	Object User - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Symmetric Signing Keys (authValue): E - Symmetric

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E - shProof: E - endorsementAuth: R
TPM2_VerifySignature	Uses loaded keys to validate a signature on a message with the message digest passed to the TPM.	'01'	Key handle, digest, signature	Validation	HMAC RSA SigVer	Unauthenticated - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_Sign	Causes the TPM to sign an externally provided hash with the specified symmetric or asymmetric signing key.	'01'	Key handle, digest, scheme, validation	Signature	HMAC RSA SigGen Primitive ECDSA SigGen	Object User - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value):

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					Component	E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys (public data): E - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_SetCommandCodeAuditStatus	Used by the Privacy Administrator or platform to change the audit status of a command or to set the hash algorithm used for the audit digest, but not both at the same time.	'00'	Auth handle, hash algorithm, list of commands to be audited, list of commands to no longer be audited	N/A	None	Object User
TPM2_PCR_Extend	Updates the indicated PCR	'01'	PCR handle, digests	N/A	SHA	Object User
TPM2_PCR_Event	Updates the indicated PCR and reports a list of digests	'01'	PCR handle, event data	Digests	SHA	Object User
TPM2_PCR_Read	Returns the values of all PCR specified in pcrSelectionIn.	'00'	PCT section to read	PCR update counter, returned PCR section, PCR values	None	Unauthenticated
TPM2_PCR_Allocate	Used to set the desired PCR	'00'	Auth handle, PCR	Allocation success, max	None	Object User -

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	allocation of PCR and algorithms. Requires Platform Authorization.		allocation selection	number of PCR, size needed, size available		platformAuth: E
TPM2_PCR_SetAuthPolicy	Used to associate a policy with a PCR or group of PCRs. The policy determines the conditions under which a PCR may be extended or reset.	'00'	Auth handle, auth policy, hash algorithm	N/A	None	Object User - platformAuth: E
TPM2_PCR_SetAuthValue	Changes the authValue of a PCR or group of PCRs.	'00'	PCR handle, auth value	N/A	None	Object User
TPM2_PCR_Reset	Used to set the PCR in all banks to zero.	'00'	PCR handle	N/A	None	Object User
TPM2_PolicySigned	Policy based on signing key	'01'	Signing key handle, policy session handle, TPM nonce, command parameter digest, policy reference, expiration, signed authorization	Timeout, policy ticket	ECDSA SigVer HMAC RSA SigVer SHA	Unauthenticated - phProof: E - ehProof: E - shProof: E - nullProof: E - Session (sessionKey): E
TPM2_PolicySecret	Policy based on an entity's authValue	'01'	Auth handle, policy session handle, TPM nonce, command parameter digest, policy reference, expiration, signed authorization	Timeout, policy ticket	HMAC SHA	Object User - phProof: E - ehProof: E - shProof: E - nullProof: E
TPM2_PolicyTicket	Policy based on ticket (produced by PolicySigned or PolicySecret)	'01'	Policy session handle, TPM nonce, command parameter digest, policy	N/A	HMAC SHA	Unauthenticated - phProof: E - ehProof: E - shProof: E - nullProof: E - Session

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			reference, auth name, ticket			(sessionKey): E
TPM2_PolicyOR	Policy enabling multiple authentication options	'01'	Policy session handle, list of hash values	N/A	SHA	Unauthenticated
TPM2_PolicyPCR	Policy based on PCR	'01'	Policy session handle, PCR digest, PCRs to include the digest	N/A	SHA	Unauthenticated
TPM2_PolicyLocality	Policy based on Locality	'01'	Policy session handle, allowed localities for the policy	N/A	SHA	Unauthenticated
TPM2_PolicyNV	Policy based on contents of an NV Index	'01'	Auth handle, nv index, policy session handle, operand B, offset of NV index for the start of operand A, operation	N/A	SHA	Object User
TPM2_PolicyCounterTimer	Policy based on time	'01'	Policy session handle, operand B, offset of TPMS_TIME_INFO for operand A, operation	N/A	SHA	Unauthenticated
TPM2_PolicyCommandCode	Policy based on command code	'01'	Policy session handle, command code	N/A	SHA	Unauthenticated

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_PolicyPhysicalPresence	Policy based on Physical Presence	'01'	Policy session handle	N/A	SHA	Unauthenticated
TPM2_PolicyCpHash	Policy bound to specific command with specific parameters and specific objects	'01'	Policy session handle, cpHash	N/A	SHA	Unauthenticated
TPM2_PolicyNameHash	Policy bound to specific objects	'01'	Policy session handle, digest to be added to the policy	N/A	SHA	Unauthenticated
TPM2_PolicyDuplicationSelect	Policy limiting duplication to only a selected parent	'01'	Policy session handle, object name, new parent name, included object name	N/A	SHA	Unauthenticated
TPM2_PolicyAuthorize	Policy enabling policy to change	'01'	Policy Session, digest of policy being approved, signing key, ticket	N/A	HMAC SHA	Unauthenticated - Session (sessionKey): E
TPM2_PolicyAuthValue	Policy bound to authValue of authorized entity (requiring HMAC session)	'01'	Policy session handle	N/A	SHA	Unauthenticated
TPM2_PolicyPassword	Policy bound to authValue of authorized entity (requiring password session)	'01'	Policy session handle	N/A	SHA	Unauthenticated
TPM2_PolicyGetDigest	Returns the current policyDigest of the session.	'00'	Policy session handle	Policy digest	None	Unauthenticated

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_PolicyNvWritten	Policy based on WRITTEN attribute of NV Index	'01'	Policy session handle	Policy digest	SHA	Unauthenticated
TPM2_PolicyTemplate	Policy bound to specific creation template	'01'	Policy session handle, indication whether NV index is required to be written	N/A	SHA	Unauthenticated
TPM2_PolicyAuthorizeNV	Policy bound to policy stored in an NV Index	'01'	Auth handle, nv index, policy session handle	N/A	SHA	Object User
TPM2_CreatePrimary	Creates a Primary Object	'01'	Primary handle, sensitive data, data to provide verifiable linkage between object and owner data, creation PCR	Object handle, public portion, creation data, creation hash, creation ticket, name	CTR_DRBG Entropy Source ECDSA KeyGen HMAC RSA KeyGen SHA AES key generation HMAC key generation	Object User - ppSeed: E - epSeed: E - spSeed: E - nullSeed: E - platformAuth: E - endorsementAuth: E - ownerAuth: E - Object Ephemeral Keys (symKey): G,W,E - Object Ephemeral Keys (hmacKey): G,W,E - Endorsement Keys (private values): R - Asymmetric Signing Keys (authValue): G,W - Asymmetric Signing Keys (seed value): G,W

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- Asymmetric Signing Keys (sensitive data): G,W - Asymmetric Signing Keys (authPolicy): G,W - Asymmetric Signing Keys (public data): G,W - Asymmetric Encryption Keys (authValue): G,W - Asymmetric Encryption Keys (seedValue): G,W - Asymmetric Encryption Keys (sensitive data): G,W - Asymmetric Encryption Keys (authPolicy): G,W - Asymmetric Encryption Keys (public data): G,W - Symmetric Encryption Keys (authValue): G,W - Symmetric Encryption Keys (seedValue): G,W - Symmetric Encryption Keys (sensitive data): G,W - Symmetric Signing Keys (authValue): G,W

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- Symmetric Signing Keys (seedValue): G,W - Symmetric Signing Keys (sensitive data): G,W
TPM2_HierarchyControl	Returns the current policyDigest of the session.	'00'	Auth handle, the enable being modified, state	N/A	None	Object User - platformAuth: E - endorsementAuth: E - ownerAuth: E - Asymmetric Signing Keys (authValue): Z - Asymmetric Signing Keys (seed value): Z - Asymmetric Signing Keys (sensitive data): Z - Asymmetric Signing Keys (authPolicy): Z - Asymmetric Signing Keys (public data): Z - Asymmetric Encryption Keys (authValue): Z - Asymmetric Encryption Keys (seedValue): Z - Asymmetric Encryption Keys (sensitive data): Z - Asymmetric Encryption

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Keys (authPolicy): Z - Asymmetric Encryption Keys (public data): Z - Symmetric Encryption Keys (authValue): Z - Symmetric Encryption Keys (seedValue): Z - Symmetric Encryption Keys (sensitive data): Z - Symmetric Signing Keys (authValue): Z - Symmetric Signing Keys (seedValue): Z - Symmetric Signing Keys (sensitive data): Z
TPM2_SetPrimaryPolicy	Allows setting of the authorization policy for the lockout (lockoutPolicy), the platform hierarchy (platformPolicy), the storage hierarchy (ownerPolicy), and the endorsement hierarchy (endorsementPolicy).	'00'	Auth handle, auth policy, hash algorithm	N/A	None	Object User - platformAuth: E - endorsementAuth: E - ownerAuth: E - lockoutAuth: E - platformPolicy: G
TPM2_ChangePPS	Changes the current platform primary seed (PPS)	'01'	Auth handle	N/A	CTR_DRBG	Object User - platformPolicy: E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- ppSeed: Z - phProof: Z - Asymmetric Signing Keys (authValue): Z - Asymmetric Signing Keys (seed value): Z - Asymmetric Signing Keys (sensitive data): Z - Asymmetric Signing Keys (authPolicy): Z - Asymmetric Signing Keys (public data): Z - Asymmetric Encryption Keys (authValue): Z - Asymmetric Encryption Keys (seedValue): Z - Asymmetric Encryption Keys (sensitive data): Z - Asymmetric Encryption Keys (authPolicy): Z - Asymmetric Encryption Keys (public data): Z - Symmetric Encryption Keys (authValue): Z - Symmetric Encryption Keys (seedValue):

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Z - Symmetric Encryption Keys (sensitive data): Z - Symmetric Signing Keys (authValue): Z - Symmetric Signing Keys (seedValue): Z - Symmetric Signing Keys (sensitive data): Z
TPM2_ChangeEPS	Changes the current endorsement primary seed (EPS)	'01'	Auth handle	N/A	CTR_DRBG	Object User - platformPolicy: E - epSeed: Z - ehProof: Z - endorsementAuth: Z - Asymmetric Signing Keys (authValue): Z - Asymmetric Signing Keys (seed value): Z - Asymmetric Signing Keys (sensitive data): Z - Asymmetric Signing Keys (authPolicy): Z - Asymmetric Signing Keys (public data): Z - Asymmetric Encryption Keys (authValue): Z - Asymmetric Encryption

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Keys (seedValue): Z - Asymmetric Encryption Keys (sensitive data): Z - Asymmetric Encryption Keys (authPolicy): Z - Asymmetric Encryption Keys (public data): Z - Symmetric Encryption Keys (authValue): Z - Symmetric Encryption Keys (seedValue): Z - Symmetric Encryption Keys (sensitive data): Z - Symmetric Signing Keys (authValue): Z - Symmetric Signing Keys (seedValue): Z - Symmetric Signing Keys (sensitive data): Z
TPM2_Clear	Zeroizes all TPM context associated with a specific Owner.	'01'	Auth handle	N/A	CTR_DRBG	Object User - platformAuth: E - lockoutAuth: E,Z - epSeed: Z - spSeed: Z - shProof: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- endorsementAuth: Z - ownerAuth: Z - endorsementPolicy: Z - ownerPolicy: Z - lockoutPolicy: Z - NV Index (authValue): Z - NV Index (authPolicy): Z
TPM2_ClearControl	Disables and enables the execution of TPM2_Clear().	'00'	Auth handle, disable flag	N/A	None	Object User - platformAuth: E - lockoutAuth: E
TPM2_HierarchyChangeAuth	Allows the authorization secret for a hierarchy or lockout to be changed using the current authorization value as the command authorization.	'00'	Auth handle, new auth value	N/A	None	Object User - platformAuth: E - endorsementAuth: E - ownerAuth: E - lockoutAuth: E
TPM2_DictionaryAttackLockReset	Cancels the effect of a TPM lockout due to a number of successive authorization failures.	'00'	Lock handle	N/A	None	Object User - lockoutAuth: E
TPM2_DictionaryAttackParameters	Changes the lockout parameters.	'00'	Lock handle, max tries, recovery time before failure count	N/A	None	Object User - lockoutAuth: E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			increases, lockout recovery time			
TPM2_PP_Commands	Used to determine which commands require assertion of Physical Presence (PP) in addition to platformAuth/platformPolicy.	'00'	Auth handle, list of commands to be asserted, list of commands to no longer be asserted	N/A	None	Object User
TPM2_ContextSave	Save a (object, object sequence or session) context	'01'	Save handle	Context	KBKDF KTS (AES + HMAC) key unwrapping	Unauthenticated - phProof: E - Context Ephemeral Keys (symKey): E - Context Ephemeral Keys (hmacKey): E - Session (salt): R
TPM2_ContextLoad	Reload a context	'01'	Context	Loaded handle	KBKDF KTS (AES + HMAC) key wrapping	Unauthenticated - phProof: E - Context Ephemeral Keys (symKey): E - Context Ephemeral Keys (hmacKey): E - Session (salt): R
TPM2_FlushContext	Causes all context associated with a loaded object, sequence object, or session to be removed from TPM memory.	'00'	Item to flush	N/A	None	Unauthenticated - Session (sessionKey): Z - Asymmetric Signing Keys (authValue): Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_EvictControl	Allows certain Transient Objects to be made persistent or a persistent object to be evicted.	'00'	Auth handle, object handle, persistent handle	N/A	None	Object User - platformAuth: E - ownerAuth: E - Asymmetric Signing Keys (authValue): W - Asymmetric Signing Keys (seed value): W - Asymmetric Signing Keys (sensitive data): W - Asymmetric Signing Keys (authPolicy): W - Asymmetric Signing Keys (public data): W - Asymmetric Encryption Keys (authValue): W - Asymmetric Encryption Keys (seedValue): W - Asymmetric Encryption Keys (sensitive data): W - Asymmetric Encryption Keys (authPolicy): W - Asymmetric Encryption Keys (public data): W - Symmetric Encryption Keys (authValue):

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						W - Symmetric Encryption Keys (seedValue): W - Symmetric Encryption Keys (sensitive data): W - Symmetric Signing Keys (authValue): W - Symmetric Signing Keys (seedValue): W - Symmetric Signing Keys (sensitive data): W
TPM2_ReadClock	Reads the current TPMS_TIME_INFO structure that contains the current setting of Time, Clock, resetCount, and restartCount.	'00'	N/A	Current time	None	Unauthenticated
TPM2_ClockSet	Used to advance the value of the TPM's Clock.	'00'	Auth handle, New time to set	N/A	None	Object User - platformAuth: E - ownerAuth: E
TPM2_ClockRateAdjust	Adjusts the rate of advance of Clock and Time to provide a better approximation to real time.	'00'	Auth handle, rate adjustment	N/A	None	Object User - platformAuth: E - ownerAuth: E
TPM2_GetCapability (Show status/version)	Shows various information regarding the TPM and its current state. This can also be used to return module's name and	'00'	Property to be read	Returned information.	None	Unauthenticated

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	versioning information.					
TPM2_TestParms	Used to check to see if specific combinations of algorithm parameters are supported.	'00'	Parameters	Success or error	None	Unauthenticated
TPM2_NV_DefineSpace	Defines the attributes of an NV Index and causes the TPM to reserve space to hold the data associated with the NV Index.	'00'	Auth handle, auth value, public parameters of the NV areaauth handle, auth value, public parameters of the NV area	N/A	None	Object User - platformAuth: E - ownerAuth: E - NV Index (authValue): G - NV Index (authPolicy): G - Endorsement Keys (public values): E
TPM2_NV_UndefineSpace	Removes an Index from the TPM.	'00'	Auth handle, NV index	N/A	None	Object User - platformAuth: E - ownerAuth: E - NV Index (authValue): Z - NV Index (authPolicy): Z
TPM2_NV_UndefineSpaceSpecial	Allows removal of a platform-created NV Index that has TPMA_NV_POLICY_DELETE SET .	'00'	NV index, platform	N/A	None	Object Administrator - platformAuth: E - ownerAuth: E - NV Index (authValue): Z - NV Index (authPolicy): Z Object User

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- platformAuth: E - ownerAuth: E - NV Index (authValue): Z - NV Index (authPolicy): Z
TPM2_NV_ReadPublic	Read public area and name of an NV Index	'01'	NV index	NV public area, NV name	SHA	Unauthenticated
TPM2_NV_Write	Writes a value to an area in NV memory that was previously defined by TPM2_NV_DefineSpace().	'00'	Auth handle, nv index, data to write, offset	N/A	None	Object User
TPM2_NV_Increment	Used to increment the value in an NV Index that has the TPM_NT_COUNTER attribute. The data value of the NV Index is incremented by one.	'00'	auth handle, NV index	N/A	None	Object User
TPM2_NV_Extend	Extend data to an NV Index	'01'	auth handle, nv index, data	N/A	SHA	Object User
TPM2_NV_SetBits	Used to SET bits in an NV Index that was created as a bit field.	'00'	auth handle, NV index, bits	N/A	None	Object User
TPM2_NV_WriteLock	If the TPMA_NV_WRITEDEFINE or TPMA_NV_WRITE_STCLEAR attributes of an NV location are SET, then this service may be used to inhibit further writes of the NV Index.	'00'	Auth handle, nv index	N/A	None	Object User

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_NV_GlobalWriteLock	Will SET TPMA_NV_WRITELOCKED for all indexes that have their TPMA_NV_GLOBALLOCK attribute SET.	'00'	Auth handle	N/A	None	Object User - platformAuth: E - ownerAuth: E
TPM2_NV_Read	Reads a value from an area in NV memory previously defined by TPM2_NV_DefineSpace().	'00'	Auth handle, nv index, number of octets to read, offset	Data	None	Object User
TPM2_NV_ReadLock	If TPMA_NV_READ_STCLEAR is SET in an Index, then this service may be used to prevent further reads of the NV Index until the next TPM2_Startup (TPM_SU_CLEAR).	'00'	Auth handle, nv index	N/A	None	Object User
TPM2_NV_ChangeAuth	Allows the authorization secret for an NV Index to be changed.	'00'	NV index, new auth value	N/A	None	Object Administrator - NV Index (authValue): W - NV Index (authPolicy): W
TPM2_NV_Certify	Certify contents of an NV Index.	'01'	Qualifying data, scheme, size, offset	Certify info, signature	ECDSA SigGen HMAC RSA SigGen SHA	Object User - Asymmetric Signing Keys (authValue): E - Asymmetric Signing Keys (seed value): E - Asymmetric Signing Keys (sensitive data): E - Asymmetric Signing Keys (authPolicy): E - Asymmetric Signing Keys

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						(public data): E - Symmetric Signing Keys (authValue): E - Symmetric Signing Keys (seedValue): E - Symmetric Signing Keys (sensitive data): E
TPM2_ACT_SetTimeout	Used to set the time remaining before an Authenticated Countdown Timer (ACT) expires.	'00'	Act handle, start timeout value	N/A	None	Object User
NTC_FIELD_UPGRADE	Used to verify arguments and protect the input firmware payload	'01'	Firmware payload	N/A	AES-CTR ECDSA SigVer	Object Administrator - Firmware Update Keys (ECC): E - Firmware Update Keys (AES): E

Table 13: Approved Services

4.4 Non-Approved Services

Name	Description	Algorithms	Role
TPM2_Create	Creation of an ordinary object	CKG	Object User
TPM2_Load	Loading an protected object	KAS-ECC-SSC	Object User
TPM2_LoadExternal	Loading an external object	KAS-ECC-SSC	None
TPM2_CreateLoaded	Creation and loading of an ordinary or a derived object	CKG	Object User
TPM2_RSA_Encrypt	RSA Encryption	RSA Key Transport	None
TPM2_RSA_Decrypt	RSA Decryption	RSA Key Transport	Object User

Name	Description	Algorithms	Role
TPM2_ECDH_ZGen	Shared Secret Calculation with TPM static key and provided public key (1e,1s)	KAS-ECC-SSC	Object User
TPM2_ZGen_2Phase	Ephemeral key pair derivation and Shared Secret Calculation with TPM ephemeral and static key and provided ephemeral and static key (2e,2s)	KAS-ECC-SSC	Object User
TPM2_HMAC	Performs a HMAC operation on user data	HMAC	Object User
TPM2_HMAC_Start	HMAC session start	HMAC	Object User
TPM2_SequenceUpdate	Sequence update	HMAC	Object User
TPM2_SequenceComplete	Sequence complete	HMAC	Object User
TPM2_EventSequenceComplete	Event sequence complete	HMAC	Object User
TPM2_Certify	Proves that an object with a specific Name is loaded in the TPM	RSA signature generation using SHA-1 ECDSA signature generation using SHA-1 HMAC	Object Administrator, Object User
TPM2_CertifyCreation	Proves the association between an object and its creation data	RSA signature generation using SHA-1 ECDSA signature generation using SHA-1 HMAC	Object Administrator, Object User
TPM2_Quote	Quotes PCR values	RSA signature generation using SHA-1 ECDSA signature generation using SHA-1 HMAC	Object User
TPM2_GetSessionAuditDigest	Returns a digital signature of the audit session digest	RSA signature generation using SHA-1 ECDSA signature generation using SHA-1 HMAC	Object User
TPM2_GetCommandAuditDigest	Returns the current value of the command audit digest	RSA signature generation using SHA-1 ECDSA signature generation using	Object User

Name	Description	Algorithms	Role
		SHA-1 HMAC	
TPM2_GetTime	Returns the current values of Time and Clock	RSA signature generation using SHA-1 ECDSA signature generation using SHA-1 HMAC	Object User
TPM2_EC_Ephemeral	Ephemeral key pair derivation	KAS-ECC-SSC	None
TPM2_VerifySignature	Uses loaded keys to validate a signature on a message with the message digest passed to the TPM.	HMAC	None
TPM2_Sign	Causes the TPM to sign an externally provided hash with the specified symmetric or asymmetric signing key.	RSA signature generation using SHA-1 ECDSA signature generation using SHA-1	Object User
TPM2_PolicySigned	Policy based on signing key	HMAC	None
TPM2_PolicySecret	Policy based on an entity's authValue	HMAC	Object User
TPM2_PolicyTicket	Policy based on ticket (produced by PolicySigned or PolicySecret)	HMAC	None
TPM2_PolicyAuthorize	Policy enabling policy to change	HMAC	None
TPM2_CreatePrimary	Creates a Primary Object	CKG	Object User
TPM2_NV_Certify	Certify contents of an NV Index	RSA signature generation using SHA-1 ECDSA signature generation using SHA-1	Object User

Table 14: Non-Approved Services

4.5 External Software/Firmware Loaded

The software/firmware load test is performed prior to loading external software or firmware on the security module. The firmware image is verified using an ECDSA signature verification algorithm, utilizing a 384-bit Firmware Update key.

5 Software/Firmware Security

5.1 Integrity Techniques

The integrity of the module is verified by comparing an HMAC-SHA-256 value calculated at run time. The integrity test uses a fixed key size of 160 bits.

5.2 Initiate on Demand

On demand integrity test may be performed by power cycling. The operator can call TPM2_Startup service from the Approved Services Table to perform on-demand integrity test. This service resets the module, resulting in the pre-operational self-tests to be re-performed.

6 Operational Environment

6.1 Operational Environment Type and Requirements

The module operates in a non-modifiable operational environment per FIPS 140-3 security level 1 specifications. The operator cannot modify the firmware components of the module.

Type of Operational Environment: Non-Modifiable

7 Physical Security

The TPM is implemented as a single integrated circuit (IC) device that attaches to standard system PCBs. It is manufactured using de-facto standard integrated circuit manufacturing technologies, producing a device that meets all commercial-grade power, temperature, reliability, shock and vibration specifications. The TPM IC physical package provides hardness, opacity and tamper-evidence protection conforming to FIPS 140-3 Physical Security Level 3. The TPM achieves this level of protection by implementing an enclosure that is both hard and opaque, as shown in the figures in Section 1. This type of IC package ensures that any physical tampering will always result in scratches, chipping, or other visible damage on the enclosure. Before the TPM is integrated into a target application system, it must be checked visually for tampering. After it is integrated, typically through soldering onto a PCB, it can be inspected for tampering by opening the application system enclosure and examining the TPM.

7.1 Mechanisms and Actions Required

Mechanism	Inspection Frequency	Inspection Guidance
Hard tamper-evident coating	Determined by the operator	Observe the coating surrounding the chip for any signs of damage

Table 15: Mechanisms and Actions Required

7.2 EFP/EFT Information

Temp/Voltage Type	Temperature or Voltage	EFP or EFT	Result
LowTemperature	-197.9°C	EFT	The module remained operational without producing errors
HighTemperature	200.4°C	EFT	The module remained operational without producing errors
LowVoltage	1.7V	EFT	Module shuts down
HighVoltage	3.47V	EFT	Module shuts down

Table 16: EFP/EFT Information

7.3 Hardness Testing Temperature Ranges

Temperature Type	Temperature
LowTemperature	-40°C
HighTemperature	105°C

Table 17: Hardness Testing Temperatures

8 Non-Invasive Security

This module does not implement any non-invasive security mechanism defined in SP 800-140F, therefore this section is not applicable.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
Flash	Storage location for firmware components and persistent SSPs.	Static
RAM	Storage location for runtime operations and transient SSPs.	Dynamic
Stack	Storage location for ephemeral keys.	Dynamic

Table 18: Storage Areas

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
TPM2_Load	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_EvictControl	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_Import	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_Create (Import)	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_Create (Export)	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key wrapping
TPM2_CreatePrimary (Import)	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_CreatePrimary (Export)	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key wrapping
TPM2_CreateLoaded	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key wrapping

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
TPM2_ContextLoad	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key wrapping
TPM2_ContextSave	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_ReadPublic	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key wrapping
TPM2_ObjectChangeAuth (Import)	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_ObjectChangeAuth (Export)	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key wrapping
TPM2_NV_ChangeAuth (Import)	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_NV_ChangeAuth (Export)	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key wrapping
TPM2_Rewrap (Import)	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_Rewrap (Export)	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key wrapping
TPM2_HierarchyChangeAuth	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_SetPrimaryPolicy	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_Duplicate (Import, Plain)	RAM	Entity using the module	Plaintext	Manual	Electronic	
TPM2_Duplicate (Import, Encrypted)	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_Duplicate (Export, Plain)	RAM	Entity using the module	Plaintext	Manual	Electronic	

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
TPM2_Duplicate (Export, Encrypted)	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key wrapping
TPM2_LoadExternal	Entity using the module	RAM	Plaintext	Automated	Electronic	
TPM2_MakeCredential	Entity using the module	RAM	Encrypted	Automated	Electronic	KTS (AES + HMAC) key unwrapping
TPM2_ActivateCredential	RAM	Entity using the module	Encrypted	Automated	Electronic	KTS (AES + HMAC) key wrapping

Table 19: SSP Input-Output Methods

The module requires two independent internal actions to output SSP in plaintext. The TPM2_Duplicate command performs the following actions:

1) Verification of the encryptedDuplication attribute of the key to be duplicated: encryptedDuplication attribute needs to be set to 0

2) Verification of the handle of the new parent of the key to be duplicated: new handle needs to be set to the NULL handle

9.3 SSP Zeroization Methods

The table below identifies every zeroization method that is available within the module. Please refer to the “Zeroization” column in SSP Table 2 for the zeroization method used for each specific SSP.

Zeroization Method	Description	Rationale	Operator Initiation
Stack Cleaning	Procedurally clears the stack after ephemeral key is no longer needed	Zeroize the stack contents in memory	Automatically by the module
TPM2_Clear	Removes all TPM context associated with a specific Owner	Zeroise objects in memory and persistent storage. Overwrites spSeed, shProof and ehProof with new values. Zeroize ownerAuth, ownerPolicy, endorsementAuth, endorsementPolicy, lockoutAuth, lockoutPolicy	By invoking the TPM2_Clear service
TPM2_ChangeEPS	Changes the current endorsement primary seed (EPS)	epSeed is overwritten by random values from the DRBG. ehProof, endorsementAuth and endorsementPolicy are	By invoking the TPM2_ChangeEPS service

Zeroization Method	Description	Rationale	Operator Initiation
		zeroized. Flushes any resident objects.	
TPM2_ChangePPS	Changes the current platform primary seed (PPS)	ppSeed is overwritten by random values from the DRBG. platformPolicy is zeroized. Flushes any resident objects.	By invoking the TPM2_ChangePPS service
TPM2_Startup	Can be used to reset the module and have all variables go to the default initialization state	All variables are overwritten back the the default values (zeroed).	By invoking the TPM2_Startup service
TPM2_FlushContext	Causes all context associated with a loaded object, sequence object, or session to be removed from TPM memory.	Clears objects from memory.	By invoking the TPM2_FlushContext service
TPM2_NV_UndefineSpace	Removes an Index from the TPM.	Index is removed from the TPM.	By invoking the TPM2_NV_UndefineSpace service
TPM2_HierarchyControl	This command enables and disables use of a hierarchy and its associated NV storage. The command allows phEnable, phEnableNV, shEnable, and ehEnable to be changed when the proper authorization is provided.	Zeroizes non-volatile stored values related to the disabled hierarchy	By invoking the TPM2_HierarchyControl service
Clear TPM	Persistent memory is zeroized using a proprietary method	Removes all module contents	For further information and instructions on clearing the flash, contact the platform manufacturer or Nuvoton support

Table 20: SSP Zeroization Methods

9.4 SSPs

The table below summarizes the Sensitive Security Parameters (SSPs) that are used by the cryptographic services implemented in the module.

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
ppSeed	KBKDF derivation keys to derive primary object's seedValue and sensitive data	512 bits - 256 bits	Seed Value - CSP	CTR_DRBG		KBKDF
epSeed	KBKDF derivation keys to derive primary object's seedValue and sensitive data	512 bits - 256 bits	Seed Value - CSP	CTR_DRBG		KBKDF
spSeed	KBKDF derivation keys to derive primary object's seedValue and sensitive data	512 bits - 256 bits	Seed Value - CSP	CTR_DRBG		KBKDF
nullSeed	KBKDF derivation keys to derive primary object's seedValue and sensitive data	512 bits - 256 bits	Seed Value - CSP	CTR_DRBG		KBKDF
phProof	Used as KBKDF derivation key to derive context encryption key. It's also used as a HMAC key to prove that an externally stored computation (context blob or a ticket) was created or checked by the TPM.	512 bits - 256 bits	Proof Values - CSP	CTR_DRBG		KBKDF
ehProof	Used as KBKDF derivation key to derive context encryption key. It's also used as a HMAC key to prove that an externally stored computation (context blob or a ticket) was created or checked by the TPM.	512 bits - 256 bits	Proof Values - CSP	CTR_DRBG		KBKDF
shProof	Used as KBKDF derivation key to derive context encryption key. It's also used as a HMAC key to prove	512 bits - 256 bits	Proof Values - CSP	CTR_DRBG		KBKDF

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	that an externally stored computation (context blob or a ticket) was created or checked by the TPM.					
nullProof	Used as KBKDF derivation key to derive context encryption key. It's also used as a HMAC key to prove that an externally stored computation (context blob or a ticket) was created or checked by the TPM.	512 bits - 256 bits	Proof Values - CSP	CTR_DRBG		KBKDF
platformAuth	Authorization data known to the hierarchy owner, required when using or changing the Hierarchy in Password or HMAC sessions.	Same as digest - Same as digest	Authorization Values - CSP			
endorsementAuth	Authorization data known to the hierarchy owner, required when using or changing the Hierarchy in Password or HMAC sessions	Same as digest - Same as digest	Authorization Values - CSP			
ownerAuth	Authorization data known to the hierarchy owner, required when using or changing the Hierarchy in Password or HMAC sessions	Same as digest - Same as digest	Authorization Values - CSP			
lockoutAuth	Authorization data known to the hierarchy owner, required when using or changing the Hierarchy in Password or HMAC sessions	Same as digest - Same as digest	Authorization Values - CSP			

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
platformPolicy	Authorization data known to the hierarchy owner, required when using or changing the Hierarchy in Policy sessions	Same as digest - Same as digest	Policies - CSP			
endorsementPolicy	Authorization data known to the hierarchy owner, required when using or changing the Hierarchy in Policy sessions	Same as digest - Same as digest	Policies - CSP			
ownerPolicy	Authorization data known to the hierarchy owner, required when using or changing the Hierarchy in Policy sessions	Same as digest - Same as digest	Policies - CSP			
lockoutPolicy	Authorization data known to the hierarchy owner, required when using or changing the Hierarchy in Policy sessions	Same as digest - Same as digest	Policies - CSP			
Asymmetric Signing Keys (authValue)	Authorization data known to the Object owner, required when using or changing the Asymmetric Signing Key Object.	Same as digest - Same as digest	Object Keys - CSP			
Asymmetric Signing Keys (seed value)	Unused (set to 0).	Same as digest - Same as digest	Object Keys - CSP	KBKDF		KBKDF
Asymmetric Signing Keys (sensitive data)	ECDSA/RSA Private Data for Signature Generation and Verification	ECDSA: P-256, P-384; RSA: 2048, 3072, 4096 - ECDSA:	Object Keys - CSP	ECDSA KeyGen RSA KeyGen		ECDSA SigGen RSA SigGen RSA SigGen Primitive ECDSA

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		128 or 192 bits; RSA: 112, 128 or 150 bits				SigGen Component
Asymmetric Signing Keys (authPolicy)	Command authentication data	Same as digest - Same as digest	Object Keys - CSP			
Asymmetric Signing Keys (public data)	ECDSA/RSA Public Data	ECDSA: P-256, P-384; RSA: 2048, 3072, 4096 - ECDSA: 128 or 192 bits; RSA: 112, 128 or 150 bits	Object Keys - PSP	ECDSA KeyGen RSA KeyGen		ECDSA KeyVer ECDSA SigVer RSA SigVer
Asymmetric Encryption Keys (authValue)	Authorization data known to the Object owner, required when using or changing the Object.	Same as digest - Same as digest	Object Keys - CSP			
Asymmetric Encryption Keys (seedValue)	Authorization data known to the Object owner, required when using or changing the Object.	Same as digest - Same as digest	Object Keys - CSP	KBKDF		KBKDF
Asymmetric Encryption Keys (sensitive data)	ECC/RSA Private Data	ECC: P-256, P-384; RSA: 2048, 3072, 4096 - ECC: 128 or 192 bits; RSA: 112, 128	Object Keys - CSP	CTR_DRBG KBKDF		KAS-ECC KAS-ECC-SSC KTS RSA

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		or 150 bits				
Asymmetric Encryption Keys (authPolicy)	Command authentication data	512 bits - 256 bits	Object Keys - CSP			
Asymmetric Encryption Keys (public data)	ECC/RSA Public Data	ECC: P-256, P-384; RSA: 2048, 3072, 4096 - ECC: 128 or 192 bits; RSA: 112, 128 or 150 bits	Object Keys - PSP			KAS-ECC KAS-ECC-SSC KTS RSA
Symmetric Encryption Keys (authValue)	Authorization data known to the Object owner, required when using or changing the Symmetric Encryption Key Object	Same as digest - Same as digest	Object Keys - CSP			
Symmetric Encryption Keys (seedValue)	Used to compute the unique field (If restricted decrypt key - by using HMAC, if not a restricted decrypt key - by hashing with the sensitive field)	Same as digest - Same as digest	Object Keys - CSP			
Symmetric Encryption Keys (sensitive data)	Symmetric encryption using AES	128 or 256 bits - 128 or 256 bits	Object Keys - CSP			AES-CFB128 AES-CTR AES-OFB
Symmetric Signing Keys (authValue)	Authorization data known to the Object owner, required when using or changing the Symmetric Signing Key Object	Same as digest - Same as digest	Object Keys - CSP			
Symmetric Signing Keys (seedValue)	Additionally used to compute the unique field (If restricted decrypt key - by	Same as digest - Same	Object Keys - CSP			

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	using HMAC, if not a restricted decrypt key - by hashing with the sensitive field)	as digest				
Symmetric Signing Keys (sensitive data)	Message Authentication Code using HMAC	160, 256, 384 bits - 128 or 256 bits	Object Keys - CSP			HMAC
Object Ephemeral Keys (symKey)	Symmetric encryption key (AES) protecting (encryption) the object sensitive data	128 or 256 bits - 128 or 256 bits	Ephemeral Key Wrapping Keys (Symmetric Encryption/Decryption) - CSP	KBKDF		AES-CFB128 KTS (AES + HMAC) key wrapping KTS (AES + HMAC) key unwrapping
Object Ephemeral Keys (hmacKey)	Symmetric signing key (HMAC) protecting (integrity) the encrypted data	160, 256, 384 bits - 128 or 256 bits	Ephemeral Key Wrapping Keys (MAC) - CSP	KBKDF		HMAC KTS (AES + HMAC) key wrapping KTS (AES + HMAC) key unwrapping
Duplication Ephemeral Keys (symKey)	Symmetric encryption key (AES) protecting (encryption) the object sensitive data	128 or 256 bits - 128 or 256 bits	Ephemeral Key Wrapping Keys (Symmetric Encryption/Decryption) - CSP		KAS-ECC KTS RSA	AES-CFB128 KTS (AES + HMAC) key wrapping KTS (AES + HMAC) key unwrapping
Duplication Ephemeral Keys (hmacKey)	Symmetric signing key (HMAC) protecting (integrity) the encrypted data	160, 256, 384 bits - 128 or 256 bits	Ephemeral Key Wrapping Keys (MAC) - CSP		KAS-ECC KTS RSA	HMAC KTS (AES + HMAC) key wrapping KTS (AES + HMAC) key unwrapping

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Duplication Ephemeral Keys (innerSymKey)	Symmetric encryption key (AES) for double protecting (encryption) the object sensitive data	128 or 256 bits - 128 or 256 bits	Ephemeral Key Wrapping Keys Ephemeral Key Wrapping Keys (Symmetric Encryption/Decryption) - CSP	CTR_DRBG		AES-CFB128 KTS (AES + HMAC) key wrapping KTS (AES + HMAC) key unwrapping
Context Ephemeral Keys (symKey)	Symmetric encryption key (AES) protecting (encryption) the the externally stored objects, sequence objects, and sessions	128 or 256 bits - 128 or 256 bits	Ephemeral Key Wrapping Keys Ephemeral Key Wrapping Keys (Symmetric Encryption/Decryption) - CSP	KBKDF		AES-CFB128 KTS (AES + HMAC) key wrapping KTS (AES + HMAC) key unwrapping
Context Ephemeral Keys (hmacKey)	Symmetric signing key (HMAC) protecting (integrity) the encrypted data	160, 256, 384 bits - 128 or 256 bits	Ephemeral Key Wrapping Keys (MAC) - CSP	KBKDF		HMAC KTS (AES + HMAC) key wrapping KTS (AES + HMAC) key unwrapping
Credential Ephemeral Keys (symKey)	Symmetric encryption key (AES) protecting (encryption) the the externally stored objects, sequence objects, and sessions	128 or 256 bits - 128 or 256 bits	Ephemeral Key Wrapping Keys (Symmetric Encryption/Decryption) - CSP		KAS-ECC KTS RSA	AES-CFB128 KTS (AES + HMAC) key wrapping KTS (AES + HMAC) key unwrapping
Credential Ephemeral Keys (hmacKey)	Symmetric signing key (HMAC) protecting (integrity) the encrypted data	160, 256, 384 bits - 128 or 256 bits	Ephemeral Key Wrapping Keys (MAC) - CSP		KAS-ECC KTS RSA	HMAC KTS (AES + HMAC) key wrapping KTS (AES + HMAC) key unwrapping

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Ephemeral Key Agreement Keys	ECC ephemeral keys used in Diffie-Hellman key exchange.	P-256, P-384 - 128 or 192 bits	Asymmetric Ephemeral Keys - CSP	CTR_DRBG KBKDF		KAS-ECC
Ephemeral User ECC Keys	ECC private key used for user cryptography support	P-256, P-384 - 128 or 192 bits	Asymmetric Ephemeral Keys - CSP	CTR_DRBG KBKDF		KAS-ECC-SSC
Endorsement Keys (private values)	Private key values for Digital Signature Generation/Verification	ECC: P-256, P-384; RSA: 2048, 3072, 4096 - ECC: 128 or 192 bits; RSA: 112, 128 or 150 bits	Endorsement Keys (Asymmetric) - CSP			ECDSA SigGen ECDSA SigVer
Endorsement Keys (public values)	Certificates containing the public RSA\ECC keys	ECC: P-256, P-384; RSA: 2048, 3072, 4096 - ECC: 128 or 192 bits; RSA: 112, 128 or 150 bits	Endorsement Keys (Asymmetric) - PSP			ECDSA SigGen ECDSA SigVer
Firmware Update Keys (ECC)	ECC Public Key Used to verify arguments of FU_Start & FU_Complete commands using ECDSA	P-256, P-384 - 128 or 192 bits	Firmware Update Keys - PSP			ECDSA SigVer
Firmware Update Keys (AES)	Used to decrypt input payload of FU_Load command	128 or 256 bits - 128 or 256 bits	Firmware Update Keys - CSP			AES-CTR

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
NV Index (authValue)	Authorization data used in authorization session, and extended into policyDigest on TPM2_PolicySecret command	Same as digest - Same as digest	NV Index - CSP			
NV Index (authPolicy)	Authorization data used in policy session	Same as digest - Same as digest	NV Index - CSP			
Session (salt)	KBKDF derivation key to derive sessionKey	384 bits - 256 bits	Session Keys - CSP			KBKDF
Session (sessionKey)	HMAC key to compute session HMAC	160, 256, 384 bits - 128 or 256 bits	Session Keys - CSP	KBKDF		HMAC KBKDF
Session (symKey)	Ephemeral symmetric encryption key for message parameter encrypt/decrypt (of the first sized buffer parameter, if a session-based encryption is used)	128 or 256 bits - 128 or 256 bits	Session Keys - CSP	KBKDF		AES-CFB128 AES-CTR AES-OFB
DRBG state	The CTR DRBG working state. Contains the current V and Key	384 bits - 256 bits	DRBG Keys - CSP			CTR_DRBG
DRBG Entropy Input	Bit stream produced from the entropy source, used as entropy input for the DRBG's seed	384 bits - 256 bits	DRBG Keys - CSP	Entropy Source		CTR_DRBG
Transient DRBG state	Local DRBG state used for pseud-random during CreatePrimary command	384 bits - 256 bits	DRBG Keys - CSP			CTR_DRBG

Table 21: SSP Table 1

The following table continues to summarize the Sensitive Security Parameters (SSPs) that are used by the cryptographic services implemented in the module.

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
ppSeed		Flash:Obfuscated	N/A	TPM2_ChangePPS	
epSeed		Flash:Obfuscated	N/A	TPM2_ChangeEPS	
spSeed		Flash:Obfuscated	N/A	TPM2_Clear	
nullSeed		RAM:Plaintext	Until reset	TPM2_Startup	
phProof		Flash:Obfuscated	N/A	TPM2_ChangePPS	
ehProof		Flash:Obfuscated	N/A	TPM2_Clear TPM2_ChangeEPS	
shProof		Flash:Obfuscated	N/A	TPM2_Clear	
nullProof		RAM:Plaintext	N/A	TPM2_Startup	
platformAuth	TPM2_HierarchyChangeAuth	RAM:Plaintext	Until reset	TPM2_Startup	
endorsementAuth	TPM2_HierarchyChangeAuth	Flash:Obfuscated	N/A	TPM2_Clear TPM2_ChangeEPS	
ownerAuth	TPM2_HierarchyChangeAuth	Flash:Obfuscated	N/A	TPM2_Clear	
lockoutAuth	TPM2_HierarchyChangeAuth	Flash:Obfuscated	N/A	TPM2_Clear	
platformPolicy	TPM2_SetPrimaryPolicy	RAM:Plaintext	N/A	TPM2_Startup	
endorsementPolicy	TPM2_SetPrimaryPolicy	Flash:Obfuscated	N/A	TPM2_Clear TPM2_ChangeEPS TPM2_Startup	
ownerPolicy	TPM2_SetPrimaryPolicy	Flash:Obfuscated	N/A	TPM2_Clear TPM2_Startup	
lockoutPolicy	TPM2_SetPrimaryPolicy	Flash:Obfuscated	N/A	TPM2_Clear TPM2_Startup	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Asymmetric Signing Keys (authValue)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Asymmetric Signing Keys (seed value):Used With Asymmetric Signing Keys (sensitive data):Used With Asymmetric Signing Keys (authPolicy):Used With Asymmetric Signing Keys (public data):Used With
Asymmetric Signing Keys (seed value)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Asymmetric Signing Keys (authValue):Used With Asymmetric Signing Keys (sensitive data):Used With Asymmetric Signing Keys (authPolicy):Used With Asymmetric Signing Keys (public data):Used With ppSeed:Derived From
Asymmetric Signing Keys (sensitive data)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Asymmetric Signing Keys (authValue):Used With Asymmetric Signing Keys (seed value):Used With Asymmetric Signing Keys

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)				(authPolicy):Used With Asymmetric Signing Keys (public data):Used With ppSeed:Derived From
Asymmetric Signing Keys (authPolicy)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Asymmetric Signing Keys (authValue):Used With Asymmetric Signing Keys (seed value):Used With Asymmetric Signing Keys (sensitive data):Used With Asymmetric Signing Keys (public data):Used With
Asymmetric Signing Keys (public data)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Asymmetric Signing Keys (authValue):Used With Asymmetric Signing Keys (seed value):Used With Asymmetric Signing Keys (sensitive data):Used With Asymmetric Signing Keys (authPolicy):Used With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	TPM2_Duplicate (Export, Encrypted)				
Asymmetric Encryption Keys (authValue)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Asymmetric Encryption Keys (seedValue):Used With Asymmetric Encryption Keys (sensitive data):Used With Asymmetric Encryption Keys (authPolicy):Used With Asymmetric Encryption Keys (public data):Used With
Asymmetric Encryption Keys (seedValue)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Asymmetric Encryption Keys (authValue):Used With Asymmetric Encryption Keys (sensitive data):Used With Asymmetric Encryption Keys (authPolicy):Used With Asymmetric Encryption Keys (public data):Used With ppSeed:Derived From
Asymmetric Encryption Keys (sensitive data)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Asymmetric Encryption Keys (authValue):Used With Asymmetric Encryption Keys (seedValue):Used

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)				With Asymmetric Encryption Keys (authPolicy):Used With Asymmetric Encryption Keys (public data):Used With ppSeed:Derived From
Asymmetric Encryption Keys (authPolicy)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Asymmetric Encryption Keys (authValue):Used With Asymmetric Encryption Keys (seedValue):Used With Asymmetric Encryption Keys (sensitive data):Used With Asymmetric Encryption Keys (public data):Used With
Asymmetric Encryption Keys (public data)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export,	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Asymmetric Encryption Keys (authValue):Used With Asymmetric Encryption Keys (seedValue):Used With Asymmetric Encryption Keys (sensitive data):Used With Asymmetric Encryption Keys (authPolicy):Used With Symmetric Encryption Keys

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Plain) TPM2_Duplicate (Export, Encrypted)				(authValue):Used With
Symmetric Encryption Keys (authValue)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Symmetric Encryption Keys (seedValue):Used With Symmetric Encryption Keys (sensitive data):Used With
Symmetric Encryption Keys (seedValue)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Symmetric Encryption Keys (authValue):Used With Symmetric Encryption Keys (sensitive data):Used With ppSeed:Derived From
Symmetric Encryption Keys (sensitive data)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyContr	Symmetric Encryption Keys (authValue):Used With Symmetric Encryption Keys

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	(Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ReadPublic TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Duplicate (Export, Plain) TPM2_Duplicate (Export, Encrypted)			ol Clear TPM	(seedValue):Used With ppSeed:Derived From
Symmetric Signing Keys (authValue)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Symmetric Signing Keys (seedValue):Used With Symmetric Signing Keys (sensitive data):Used With
Symmetric Signing Keys (seedValue)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import) TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext TPM2_HierarchyControl Clear TPM	Symmetric Signing Keys (authValue):Used With Symmetric Signing Keys (sensitive data):Used With ppSeed:Derived From
Symmetric Signing Keys (sensitive data)	TPM2_Load TPM2_LoadExternal TPM2_EvictControl TPM2_Create (Import)	RAM:Plaintext Flash:Obfuscated	N/A	TPM2_ChangeEPS TPM2_ChangePPS TPM2_Startup TPM2_FlushContext	Symmetric Signing Keys (authValue):Used With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	TPM2_Create (Export) TPM2_CreatePrimary (Import) TPM2_CreatePrimary (Export) TPM2_CreateLoaded TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export) TPM2_Rewrap (Import) TPM2_Rewrap (Export)			TPM2_HierarchyControl Clear TPM	Symmetric Signing Keys (seedValue):Used With ppSeed:Derived From
Object Ephemeral Keys (symKey)	TPM2_Load TPM2_Create (Import) TPM2_Create (Export) TPM2_CreateLoaded TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export)	Stack:Plaintext	Until no longer needed, or power reset.	Stack Cleaning	Object Ephemeral Keys (hmacKey):Used With Asymmetric Signing Keys (authValue):Encrypts Asymmetric Signing Keys (seed value):Encrypts Asymmetric Signing Keys (sensitive data):Encrypts Asymmetric Signing Keys (authPolicy):Encrypts Asymmetric Signing Keys (public data):Encrypts Asymmetric Encryption Keys (authValue):Encrypts Asymmetric Encryption Keys (seedValue):Encrypts Asymmetric Encryption Keys (sensitive data):Encrypts Asymmetric Encryption Keys (authPolicy):Encrypts Asymmetric Encryption Keys

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					(public data):Encrypts Symmetric Encryption Keys (authValue):Encrypts Symmetric Encryption Keys (seedValue):Encrypts Symmetric Encryption Keys (sensitive data):Encrypts Symmetric Signing Keys (authValue):Encrypts Symmetric Signing Keys (seedValue):Encrypts Symmetric Signing Keys (sensitive data):Encrypts Asymmetric Signing Keys (authValue):Decrypts Asymmetric Signing Keys (seed value):Decrypts Asymmetric Signing Keys (sensitive data):Decrypts Asymmetric Signing Keys (authPolicy):Decrypts Asymmetric Signing Keys (public data):Decrypts Asymmetric Encryption Keys (authValue):Decrypts Asymmetric Encryption Keys (seedValue):Decrypts Asymmetric Encryption Keys

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					(sensitive data):Decrypts Asymmetric Encryption Keys (authPolicy):Decrypts Asymmetric Encryption Keys (public data):Decrypts Symmetric Encryption Keys (authValue):Decrypts Symmetric Encryption Keys (seedValue):Decrypts Symmetric Encryption Keys (sensitive data):Decrypts Symmetric Signing Keys (authValue):Decrypts Symmetric Signing Keys (seedValue):Decrypts Symmetric Signing Keys (sensitive data):Decrypts Asymmetric Signing Keys (seed value):Derived From Asymmetric Encryption Keys (seedValue):Derived From Symmetric Encryption Keys (seedValue):Derived From Symmetric Signing Keys (seedValue):Derived From
Object Ephemeral Keys (hmacKey)	TPM2_Load TPM2_LoadExternal TPM2_Create (Import) TPM2_Create (Export)	Stack:Plaintext	Until no longer needed, or	Stack Cleaning	Object Ephemeral Keys (symKey):Used With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	TPM2_CreateLoaded TPM2_ObjectChangeAuth (Import) TPM2_ObjectChangeAuth (Export)		power reset.		Asymmetric Signing Keys (seed value):Derived From Asymmetric Encryption Keys (seedValue):Derived From Symmetric Encryption Keys (seedValue):Derived From Symmetric Signing Keys (seedValue):Derived From
Duplication Ephemeral Keys (symKey)	TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Import TPM2_Duplicate (Import, Plain) TPM2_Duplicate (Import, Encrypted)	Stack:Plaintext	Until no longer needed, or power reset.	Stack Cleaning	Duplication Ephemeral Keys (hmacKey):Used With Duplication Ephemeral Keys (innerSymKey):Used With Asymmetric Signing Keys (authValue):Encrypts Asymmetric Signing Keys (seed value):Encrypts Asymmetric Signing Keys (sensitive data):Encrypts Asymmetric Signing Keys (authPolicy):Encrypts Asymmetric Signing Keys (public data):Encrypts Asymmetric Encryption Keys (authValue):Encrypts Asymmetric Encryption Keys (seedValue):Encrypts Asymmetric

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Encryption Keys (sensitive data):Encrypts Asymmetric Encryption Keys (authPolicy):Encrypts Asymmetric Encryption Keys (public data):Encrypts Symmetric Encryption Keys (authValue):Encrypts Symmetric Encryption Keys (seedValue):Encrypts Symmetric Encryption Keys (sensitive data):Encrypts Symmetric Signing Keys (authValue):Encrypts Symmetric Signing Keys (seedValue):Encrypts Symmetric Signing Keys (sensitive data):Encrypts Asymmetric Signing Keys (authValue):Decrypts Asymmetric Signing Keys (seed value):Decrypts Asymmetric Signing Keys (sensitive data):Decrypts Asymmetric Signing Keys (authPolicy):Decrypts Asymmetric Signing Keys (public data):Decrypts Asymmetric

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Encryption Keys (authValue):Decrypts Asymmetric Encryption Keys (seedValue):Decrypts Asymmetric Encryption Keys (sensitive data):Decrypts Asymmetric Encryption Keys (authPolicy):Decrypts Asymmetric Encryption Keys (public data):Decrypts Symmetric Encryption Keys (authValue):Decrypts Symmetric Encryption Keys (seedValue):Decrypts Symmetric Encryption Keys (sensitive data):Decrypts Symmetric Signing Keys (authValue):Decrypts Symmetric Signing Keys (seedValue):Decrypts Symmetric Signing Keys (sensitive data):Decrypts Ephemeral Key Agreement Keys:Derived From DRBG state:Derived From
Duplication Ephemeral Keys (hmacKey)	TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Import TPM2_Duplicate (Import, Plain)	Stack:Plaintext	Until no longer needed, or	Stack Cleaning	Duplication Ephemeral Keys (symKey):Used With Duplication

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	TPM2_Duplicate (Import, Encrypted)		power reset.		Ephemeral Keys (innerSymKey):Used With Ephemeral Key Agreement Keys:Derived From DRBG state:Derived From
Duplication Ephemeral Keys (innerSymKey)	TPM2_Rewrap (Import) TPM2_Rewrap (Export) TPM2_Import TPM2_Duplicate (Import, Plain) TPM2_Duplicate (Import, Encrypted)	Stack:Plaintext	Until no longer needed, or power reset.	Stack Cleaning	Duplication Ephemeral Keys (symKey):Used With Duplication Ephemeral Keys (hmacKey):Used With Asymmetric Signing Keys (authValue):Encrypts Asymmetric Signing Keys (seed value):Encrypts Asymmetric Signing Keys (sensitive data):Encrypts Asymmetric Signing Keys (authPolicy):Encrypts Asymmetric Signing Keys (public data):Encrypts Asymmetric Encryption Keys (authValue):Encrypts Asymmetric Encryption Keys (seedValue):Encrypts Asymmetric Encryption Keys (sensitive data):Encrypts Asymmetric Encryption Keys (authPolicy):Encrypts

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Asymmetric Encryption Keys (public data):Encrypts Symmetric Encryption Keys (authValue):Encrypts Symmetric Encryption Keys (seedValue):Encrypts Symmetric Encryption Keys (sensitive data):Encrypts Symmetric Signing Keys (authValue):Encrypts Symmetric Signing Keys (seedValue):Encrypts Symmetric Signing Keys (sensitive data):Encrypts Asymmetric Signing Keys (authValue):Decrypts Asymmetric Signing Keys (seed value):Decrypts Asymmetric Signing Keys (sensitive data):Decrypts Asymmetric Signing Keys (authPolicy):Decrypts Asymmetric Signing Keys (public data):Decrypts Asymmetric Encryption Keys (authValue):Decrypts Asymmetric Encryption Keys (seedValue):Decrypts

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Asymmetric Encryption Keys (sensitive data):Decrypts Asymmetric Encryption Keys (authPolicy):Decrypts Asymmetric Encryption Keys (public data):Decrypts Symmetric Encryption Keys (authValue):Decrypts Symmetric Encryption Keys (seedValue):Decrypts Symmetric Encryption Keys (sensitive data):Decrypts Symmetric Signing Keys (authValue):Decrypts Symmetric Signing Keys (seedValue):Decrypts Symmetric Signing Keys (sensitive data):Decrypts
Context Ephemeral Keys (symKey)	TPM2_ContextLoad TPM2_ContextSave	Stack:Plaintext	Until no longer needed, or power reset.	Stack Cleaning	Context Ephemeral Keys (hmacKey):Used With Session (salt):Encrypts Session (salt):Decrypts phProof:Derived From
Context Ephemeral Keys (hmacKey)	TPM2_ContextLoad TPM2_ContextSave	Stack:Plaintext	Until no longer needed, or power reset.	Stack Cleaning	Context Ephemeral Keys (symKey):Used With phProof:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Credential Ephemeral Keys (symKey)	TPM2_MakeCredential TPM2_ActivateCredential	Stack:Plaintext	Until no longer needed, or power reset.	Stack Cleaning	Credential Ephemeral Keys (hmacKey):Used With Asymmetric Signing Keys (authValue):Encrypts Asymmetric Signing Keys (seed value):Encrypts Asymmetric Signing Keys (sensitive data):Encrypts Asymmetric Signing Keys (authPolicy):Encrypts Asymmetric Signing Keys (public data):Encrypts Asymmetric Encryption Keys (authValue):Encrypts Asymmetric Encryption Keys (seedValue):Encrypts Asymmetric Encryption Keys (sensitive data):Encrypts Asymmetric Encryption Keys (authPolicy):Encrypts Asymmetric Encryption Keys (public data):Encrypts Symmetric Encryption Keys (authValue):Encrypts Symmetric Encryption Keys (seedValue):Encrypts Symmetric Encryption Keys

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					(sensitive data):Encrypts Symmetric Signing Keys (authValue):Encrypts Symmetric Signing Keys (seedValue):Encrypts Symmetric Signing Keys (sensitive data):Encrypts Asymmetric Signing Keys (authValue):Decrypts Asymmetric Signing Keys (seed value):Decrypts Asymmetric Signing Keys (sensitive data):Decrypts Asymmetric Signing Keys (authPolicy):Decrypts Asymmetric Signing Keys (public data):Decrypts Asymmetric Encryption Keys (authValue):Decrypts Asymmetric Encryption Keys (seedValue):Decrypts Asymmetric Encryption Keys (sensitive data):Decrypts Asymmetric Encryption Keys (authPolicy):Decrypts Asymmetric Encryption Keys (public data):Decrypts Symmetric Encryption Keys

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					(authValue):Decrypts Symmetric Encryption Keys (seedValue):Decrypts Symmetric Encryption Keys (sensitive data):Decrypts Symmetric Signing Keys (authValue):Decrypts Symmetric Signing Keys (seedValue):Decrypts Symmetric Signing Keys (sensitive data):Decrypts Ephemeral Key Agreement Keys:Derived From DRBG state:Derived From
Credential Ephemeral Keys (hmacKey)	TPM2_MakeCredential TPM2_ActivateCredential	Stack:Plaintext	Until no longer needed, or power reset.	Stack Cleaning	Credential Ephemeral Keys (symKey):Used With Ephemeral Key Agreement Keys:Derived From DRBG state:Derived From
Ephemeral Key Agreement Keys		Stack:Plaintext	Until no longer needed, or power reset.	Stack Cleaning	
Ephemeral User ECC Keys	TPM2_Load TPM2_LoadExternal	Stack:Plaintext	Until no longer needed, or power reset.	Stack Cleaning	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Endorsement Keys (private values)	TPM2_CreatePrimary (Import)	Flash:Obfuscated	N/A	TPM2_ChangeEPS Clear TPM	Endorsement Keys (public values):Paired With epSeed:Used With
Endorsement Keys (public values)		Flash:Obfuscated	N/A	TPM2_ChangeEPS Clear TPM	Endorsement Keys (private values):Paired With epSeed:Used With
Firmware Update Keys (ECC)		Flash:Obfuscated	N/A	Clear TPM	Firmware Update Keys (AES):Used With
Firmware Update Keys (AES)		Flash:Obfuscated	N/A	Clear TPM	Firmware Update Keys (ECC):Used With
NV Index (authValue)	TPM2_NV_ChangeAuth (Import) TPM2_NV_ChangeAuth (Export)	Flash:Obfuscated	N/A	TPM2_Clear TPM2_ChangeEPS TPM2_ChangePPS TPM2_NV_UndefineSpace Clear TPM	NV Index (authPolicy):Used With
NV Index (authPolicy)	TPM2_NV_ChangeAuth (Import) TPM2_NV_ChangeAuth (Export)	Flash:Obfuscated	N/A	TPM2_Clear TPM2_ChangeEPS TPM2_ChangePPS TPM2_NV_UndefineSpace Clear TPM	NV Index (authValue):Used With
Session (salt)	TPM2_ContextLoad TPM2_ContextSave	Stack:Plaintext	N/A	Stack Cleaning	
Session (sessionKey)		RAM:Plaintext	N/A	TPM2_Startup TPM2_FlushContext	phProof:Used With ehProof:Used With shProof:Used With nullProof:Used With Session (salt):Derived From
Session (symKey)		Stack:Plaintext	N/A	Stack Cleaning	Session (sessionKey):Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
DRBG state		RAM:Plaintext Flash:Obfuscated	N/A	TPM2_Startup Clear TPM	DRBG Entropy Input:Derived From
DRBG Entropy Input		RAM:Plaintext Flash:Obfuscated	N/A	TPM2_Startup Clear TPM	DRBG state:Paired With
Transient DRBG state		RAM:Plaintext Flash:Obfuscated	N/A	TPM2_Startup Clear TPM	DRBG Entropy Input:Derived From

Table 22: SSP Table 2

10 Self-Tests

10.1 Pre-Operational Self-Tests

The Module implements the following tests during power-on:

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
HMAC-SHA2-256 (A4792)	HMAC-SHA2-256	Message Authentication Code (MAC)	SW/FW Integrity	Successful boot	Performed on system startup

Table 23: Pre-Operational Self-Tests

The module does not provide any cryptographic services prior to this test.

10.2 Conditional Self-Tests

The Module implements the following conditional tests:

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Counter DRBG (A4792)	256-bit key	KAT	CAST	Successful boot	Random Number Generation	Upon first invocation of service that uses the algorithm.
HMAC-SHA2-384 (A4792)	384 bit keys	KAT	CAST	Successful boot	Verify	Upon first invocation of service that uses the algorithm.
KDF SP800-108 (A4792)	SHA2-256	KAT	CAST	Successful boot	Key Derivation	Upon first invocation of service that uses the algorithm.
KDA OneStep Sp800-56Cr1 (A4792)	SHA2-256	KAT	CAST	Successful boot	Key Derivation	Upon first invocation of service that uses the algorithm.
SHA-1 (A4792)	SHA-1	KAT	CAST	Successful boot	Message Digest	Upon first invocation of service that uses the algorithm.
SHA2-256 (A4792)	SHA2-256	KAT	CAST	Successful boot	Message Digest	Upon first invocation of service that

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						uses the algorithm.
SHA2-384 (A4792)	SHA2-384	KAT	CAST	Successful boot	Message Digest	Upon first invocation of service that uses the algorithm.
AES-CFB128 (A4792)	128, 256 bit keys	KAT	CAST	Successful boot	Encryption / Decryption	Upon first invocation of service that uses the algorithm.
AES-CTR (A4792)	128, 256 bit keys	KAT	CAST	Successful boot	Encryption / Decryption	Upon first invocation of service that uses the algorithm.
AES-OFB (A4792)	128, 256 bit keys	KAT	CAST	Successful boot	Encryption / Decryption	Upon first invocation of service that uses the algorithm.
RSA SigGen (FIPS186-4) (A4792)	2048-bit modulus; Hash: SHA2-256	KAT	CAST	Successful boot	Signature Generation	Upon first invocation of service that uses the algorithm.
RSA SigVer (FIPS186-4) (A4792)	2048-bit modulus; Hash: SHA2-256	KAT	CAST	Successful boot	Signature Verification	Upon first invocation of service that uses the algorithm.
ECDSA SigGen (FIPS186-4) (A4792)	Curve: P-256; Hash: SHA2-256	KAT	CAST	Successful boot	Signature Generation	Upon first invocation of service that uses the algorithm.
ECDSA SigVer (FIPS186-4) (A4792)	Curve: P-256; Hash: SHA2-256	KAT	CAST	Successful boot	Signature Verification	Upon first invocation of service that uses the algorithm.
KAS-ECC Sp800-56Ar3 (A4792)	Curve: P-256	KAT	CAST	Successful boot	Shared Secret Computation	Upon first invocation of service that

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						uses the algorithm.
KTS-IFC (A4792)	2048-bit modulus	KAT	CAST	Successful boot	RSA Key Transport	Upon first invocation of service that uses the algorithm.
ECDSA KeyGen (FIPS186-4) (A4792)	P-256, P-384 curves	PCT	PCT	Key pair returned to caller	Signature Generation / Signature Verification	Performed every time ECC key pair is generated
RSA KeyGen (FIPS186-4) (A4792)	2048, 3072, 4096 bit keys	PCT	PCT	Key pair returned to caller	Encryption / Decryption tested for RSA key pairs generated for approved key transport and Signature Generation / Signature Verification tested for RSA key pairs generated for digital signatures	Performed every time RSA key pair is generated
ECDSA SigVer (SW/FW Load Test)	P-384 Curves	SW/FW Load Test	SW/FW Load	Successful Firmware load	Firmware update test during the firmware update. The digital signature is verified on the firmware image using an ECDSA signature verification algorithm, utilizing a 384-bit	Firmware Update

Table 24: Conditional Self-Tests

No services are available, and input and output are inhibited while performing the self-test.

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A4792)	Message Authentication Code (MAC)	SW/FW Integrity	On demand	Manually

Table 25: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Counter DRBG (A4792)	KAT	CAST	On demand	Manually
HMAC-SHA2-384 (A4792)	KAT	CAST	On demand	Manually
KDF SP800-108 (A4792)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KDA OneStep Sp800-56Cr1 (A4792)	KAT	CAST	On demand	Manually
SHA-1 (A4792)	KAT	CAST	On demand	Manually
SHA2-256 (A4792)	KAT	CAST	On demand	Manually
SHA2-384 (A4792)	KAT	CAST	On demand	Manually
AES-CFB128 (A4792)	KAT	CAST	On demand	Manually
AES-CTR (A4792)	KAT	CAST	On demand	Manually
AES-OFB (A4792)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-4) (A4792)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-4) (A4792)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-4) (A4792)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-4) (A4792)	KAT	CAST	On demand	Manually
KAS-ECC Sp800-56Ar3 (A4792)	KAT	CAST	On demand	Manually
KTS-IFC (A4792)	KAT	CAST	On demand	Manually
ECDSA KeyGen (FIPS186-4) (A4792)	PCT	PCT	N/A	N/A
RSA KeyGen (FIPS186-4) (A4792)	PCT	PCT	N/A	N/A
ECDSA SigVer (SW/FW Load Test)	SW/FW Load Test	SW/FW Load	N/A	N/A

Table 26: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
General Failure	General Failure	Endorsement Key creation failure, Internal NV inconsistency Fingerprint value in TPM2_ContextLoad doesn't	Platform Reset or power cycle	returns TPM_RC_FAILURE

Name	Description	Conditions	Recovery Method	Indicator
		match Post-field upgrade problem		
Self-Test Failure	Failure in conditional CAST, Conditional PCT or FW Integrity Test failure	Internal integrity error - indicative of fault injection attack or internal functional fault	Power cycle	returns SELF_TEST_FAILURE

Table 27: Error States

If a conditional or power-on self-test fails, the Module enters an error state where both data output and cryptographic services are disabled. The Module can recover from this error state once all self-tests pass after a platform reset or power cycle.

10.5 Operator Initiation of Self-Tests

The module allows operators to initiate the pre-operational or conditional cryptographic algorithm self-tests on demand for periodic testing, by power-cycling the module.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The module is not considered to be initialized until the following are completed:

- The module must be connected on the PCB as described in the Module technical specifications. The connection must ensure one-to-one binding with the platform.
- The platform on which the module is installed should include BIOS and OS that initialize and control TPM hierarchies and set hierarchy's authorization value and policy. If the platform does not have such BIOS and OS, the crypto-officer shall install software to manage TPM hierarchies and set the hierarchy's authorization and policy.

11.2 Administrator Guidance

The administrator guidance can be found within the TCG TPM v2.0 Provisioning Guidance.

11.3 Non-Administrator Guidance

The module may be operated as described in TCG TPM2.0 Revision 1.59.

12 Mitigation of Other Attacks

The module does not claim any mitigation of other attacks.

References

FIPS140-3	FIPS PUB 140-3 - Security Requirements for Cryptographic Modules March 2019 https://doi.org/10.6028/NIST.FIPS.140-3
FIPS140-3_IG	Implementation Guidance for FIPS PUB 140-3 and the Cryptographic Module Validation Program January 29, 2024 https://csrc.nist.gov/CSRC/media/Projects/cryptographic-module-validation-program/documents/fips_140-3/FIPS_140-3_IG.pdf
TCG TPM v2.0 Provisioning Guidance Version 1.0 Revision 1.0	TCG TPM v2.0 Provisioning Guidance Version 1.0 Revision 1.0 March 15, 2017 https://trustedcomputinggroup.org/tcg-tpm-v2-0-provisioning-guidance
TPM Library Specification	TPM Library specification, Family “2.0”, Revision 1.59 November 2019 https://trustedcomputinggroup.org/resource/tpm-library-specification/
Platform TPM Profile Specification	TCG PC Client Platform TPM Profile Specification for TPM 2.0 September 4, 2020 https://trustedcomputinggroup.org/wp-content/uploads/PC-Client-Specific-Platform-TPM-Profile-for-TPM-2p0-v1p05p_r14_pub.pdf
TCG TPM2.0 Revision 1.59	TCG TPM2.0 Revision 1.59, version 1.4 January 9, 2023 https://trustedcomputinggroup.org/wp-content/uploads/TPM-2.0-Library-Spec-v1.59-Errata-v1.4_pub.pdf
FIPS 180-4	Secure Hash Standard (SHS) March 2012 https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf
FIPS 186-4	Digital Signature Standard (DSS) February 2023 https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf
FIPS 197	Advanced Encryption Standard November 2001 https://csrc.nist.gov/publications/fips/fips197/fips-197.pdf
FIPS 198-1	The Keyed Hash Message Authentication Code (HMAC) July 2008 https://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf
PKCS#1	Public Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1 February 2003 https://www.ietf.org/rfc/rfc3447.txt

SP 800-38Arev1
NIST Special Publication 800-38A - Recommendation for Block Cipher Modes of Operation Methods and Techniques
December 2001
https://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf

SP 800-38B
NIST Special Publication 800-38B - Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication
May 2005
https://csrc.nist.gov/publications/detail/sp/800-38b/final

SP 800-38C
NIST Special Publication 800-38C - Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality
May 2004
https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38c.pdf

SP 800-38D
NIST Special Publication 800-38D - Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC
November 2007
https://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf

SP 800-56Arev3
NIST Special Publication 800-56A Revision 2 - Recommendation for Pair Wise Key Establishment Schemes Using Discrete Logarithm Cryptography
April 2018
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Ar3.pdf

SP 800-57rev5
NIST Special Publication 800-57 Part 1 Revision 5 - Recommendation for Key Management Part 1: General
May 2020
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r5.pdf

SP 800-90Arev1
NIST Special Publication 800-90A - Revision 1 - Recommendation for Random Number Generation Using Deterministic Random Bit Generators
June 2015
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90Ar1.pdf

SP 800-90B
NIST Special Publication 800-90B - Recommendation for the Entropy Sources Used for Random Bit Generation
January 2018
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90B.pdf

SP 800-131Arev2
NIST Special Publication 800-131A Revision 2 - Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths
March 2019
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-131Ar2.pdf

SP 800-133rev2
NIST Special Publication 800-133 Revision 2 - Recommendation for Cryptographic Key Generation
June 2020
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-133r2.pdf

SP 800-135rev1
NIST Special Publication 800-135 Revision 1 - Recommendation for Existing Application-Specific Key Derivation Functions
December 2011
https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-135r1.pdf

SP 800-140Br1 NIST Special Publication 800-140Br1 - CMVP Security Policy Requirements
November 2023
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-140Br1.pdf