HP Inc.

HP Endpoint Security Controller Cryptographic Library

Hardware Version 2.1.4

FIPS 140-3 Non-Proprietary Security Policy

Document Version 1.1

Last update: 2025-09-15

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 Hardware version 2.1.4 of the HP Endpoint Security Controller Cryptographic Library. 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. This document also contains the security rules under which the module must operate and describes how this module meets the requirements as specified in FIPS PUB 140-3 (Federal Information Processing Standards Publication 140-3) for a Security Level 1 module.

1.2 Security Levels

Table 1 describes the individual security areas of FIPS 140-3, as well as the Security Levels of those individual areas:

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	N/A
6	Operational environment	1
7	Physical security	1
8	Non-invasive security	N/A
9	Sensitive security parameter management	1
10	Self-tests	1
11	Life-cycle assurance	1
12	Mitigation of other attacks	N/A
	Overall Level	1

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The HP Endpoint Security Controller Cryptographic Library (hereafter referred to as “the module”) is a Hardware Single Chip cryptographic module. More specifically, the module is considered a sub-chip cryptographic subsystem as defined in IG 2.3.B. This module validation is a re-branding of a sub-chip cryptographic subsystem that was previously validated under Certificate #5008.

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 cryptographic boundary encompasses the entire physical chip.

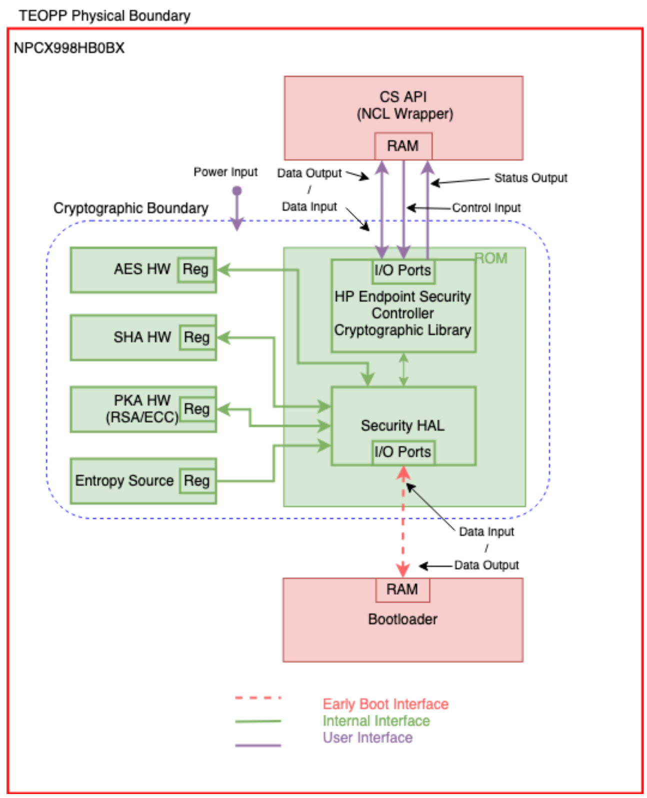

Figure 1: Block Diagram

Tested Operational Environment’s Physical Perimeter (TOEPP):

The red outline in Figure 1 above indicates the Tested Operational Environment’s Physical Perimeter (TOEPP).

Figure 2: Nuvoton NPCX998HB0BX

Figure 2 shows a picture of the NPCX998HB0BX (e.g., EC) in which the sub-chip module is embedded.

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
Notebook Embedded Controller (EC)	2.1.4	N/A	Nuvoton NPCX998HB0BX	N/A

Table 2: Tested Module Identification – Hardware

The EC component is rebranded as the HP Endpoint Security Controller and is used in select HP Commercial PC products including Notebooks, Desktops, Desktop Workstations, Mobile Workstations, Retail Point of Sale Terminals, and Thin Clients.

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:

The module supports approved services in the approved mode of operation. There are no non-approved services supported by the module.

Mode Name	Description	Type	Status Indicator
Approved Mode	Only approved algorithms are used	Approved	1

Table 3: Modes List and Description

2.5 Algorithms

Approved Algorithms:

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

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A2825	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A2825	Key Length - 128, 192, 256	SP 800-38C
AES-CFB128	A2825	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CMAC	A2825	Direction - Generation, Verification Key Length - 128, 192, 256	SP 800-38B
AES-CTR	A2825	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A2825	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A2825	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.2 Key Length - 128, 192, 256	SP 800-38D
AES-GMAC	A2825	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.2 Key Length - 128, 192, 256	SP 800-38D
AES-OFB	A2825	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
ECDSA KeyGen (FIPS186-4)	A2825	Curve - P-256, P-384, P-521	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A2825	Curve - P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A2825	Component - No, Yes Curve - P-256, P-384, P-521	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A2825	Component - No Curve - P-256, P-384, P-521	FIPS 186-4
Hash DRBG	A2825	Prediction Resistance - No, Yes Mode - SHA2-512	SP 800-90A Rev. 1
HMAC-SHA2-256	A2825	Key Length - Key Length: 256-512 Increment 8	FIPS 198-1
HMAC-SHA2-384	A2825	Key Length - Key Length: 256-512 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-512	A2825	Key Length - Key Length: 256-512 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A2825	Domain Parameter Generation Methods - P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KTS-IFC	A2825	Modulo - 2048, 3072 Key Generation Methods - rsakpg2-basic Scheme - KTS-OAEP-basic - KAS Role - initiator, responder Key Length - 1024	SP 800-56B Rev. 2
RSA SigGen (FIPS186-4)	A2825	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072	FIPS 186-4
RSA SigVer (FIPS186-4)	A2825	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072	FIPS 186-4
SHA2-256	A2825	-	FIPS 180-4
SHA2-384	A2825	-	FIPS 180-4
SHA2-512	A2825	-	FIPS 180-4

Table 4: Approved Algorithms

Vendor-Affirmed Algorithms

Name	Properties	Implementation	Reference
CKG (ECDSA/ECDH)	Type:Asymmetric Curves:P-256, P-384, P-521	N/A	CKG for asymmetric keys as per SP 800-133Rev2 section 4 example 1 with no post processing on the U value

Table 5: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

Non-Approved, Not Allowed Algorithms:

N/A for this module.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
AES-CBC	BC-UnAuth	AES Encryption and AES Decryption	Key Size:128, 192, 256 bits Key Strength:128, 192, 256 bits	AES-CBC: (A2825)
AES-CCM	BC-Auth	Authenticated AES Encryption and AES Decryption	Key Size:128, 192, 256 bits Key Strength:128, 192, 256 bits	AES-CCM: (A2825)
AES-CFB128	BC-UnAuth	AES Encryption and AES Decryption	Key Size:128, 192, 256 bits Key Strength:128, 192, 256 bits	AES-CFB128: (A2825)
AES-CMAC	MAC	CMAC Message Authentication Code Generation and CMAC Message Authentication Code Verification	Key Size:128, 192, 256 bits	AES-CMAC: (A2825)
AES-CTR	BC-UnAuth	AES Encryption and AES Decryption	Key Size:128, 192, 256 bits Key Strength:128, 192, 256 bits	AES-CTR: (A2825)
AES-ECB	BC-UnAuth	AES Encryption and AES Decryption	Key Size:128, 192, 256 bits Key Strength:128, 192, 256 bits	AES-ECB: (A2825)
AES-GCM	BC-Auth	Authenticated AES Encryption and AES Decryption	Key Size:128, 192, 256 bits Key Strength:128, 192, 256 bits	AES-GCM: (A2825)
AES-GMAC	MAC	GMAC Message Authentication Code Generation and GMAC Message Authentication Code Verification	Key Size:128, 192, 256 bits Key Strength:128, 192, 256 bits	AES-GMAC: (A2825)

Name	Type	Description	Properties	Algorithms
AES-OFB	BC-UnAuth	AES Encryption and AES Decryption	Key Size:128, 192, 256 bits Key Strength:128, 192, 256 bits	AES-OFB: (A2825)
HMAC	MAC	HMAC Message Authentication Code Generation	Key Size:256, 384, 512 bits Key Strength:256, 384, 512 bits	HMAC-SHA2-256: (A2825) HMAC-SHA2-384: (A2825) HMAC-SHA2-512: (A2825)
RSA SigGen	DigSig-SigGen	RSA Signature Generation	Signature Types:PKCS#1 v1.5, RSA-PSS Message Digest:SHA2-256, SHA2-384, SHA2-512 Modulus Size:2048, 3072	RSA SigGen (FIPS186-4): (A2825)
RSA SigVer	DigSig-SigVer	RSA Signature Verification	Signature Types:PKCS#1 v1.5, RSA-PSS Message Digest:SHA2-256, SHA2-384, SHA2-512 Modulus Size:2048, 3072	RSA SigVer (FIPS186-4): (A2825)
KTS-IFC (Wrap)	KTS-Wrap	RSA Key Transport (key wrapping)	Scheme:KTS-OAEP-basic Modulus Size:2048, 3072	KTS-IFC: (A2825)
KTS-IFC (Unwrap)	KTS-Wrap	RSA Key Transport (key unwrapping)	Scheme:KTS-OAEP-basic Modulus Size:2048, 3072	KTS-IFC: (A2825)
ECDSA KeyGen	AsymKeyPair-KeyGen	ECDSA Key Generation	Generation Method:B.4.2 Testing Candidates Curves:P-256, P-384, P-521	ECDSA KeyGen (FIPS186-4): (A2825)
ECDSA KeyVer	AsymKeyPair-KeyVer	ECDSA Key Verification	Curves:P-256, P-384, P-521	ECDSA KeyVer (FIPS186-4): (A2825)
ECDSA SigGen	DigSig-SigGen	ECDSA Signature Generation	Message Digest:SHA2-256, SHA2-384, SHA2-512 Curves:P-256, P-384, P-521	ECDSA SigGen (FIPS186-4): (A2825)
ECDSA SigVer	DigSig-SigVer	ECDSA Signature Verification	Message Digest:SHA2-256, SHA2-384, SHA2-512	ECDSA SigVer (FIPS186-4): (A2825)

Name	Type	Description	Properties	Algorithms
			Curves:P-256, P-384, P-521	
ECDSA SigGen Component	DigSig-SigGen	ECDSA Signature Generation Component	Curves:P-256, P-384, P-521	ECDSA SigGen (FIPS186-4): (A2825)
SHS	SHA	Message Digest Generation		SHA2-256: (A2825) SHA2-384: (A2825) SHA2-512: (A2825)
KAS-ECC-SSC	KAS-SSC	EC Diffie-Hellman Shared Secret Computation	Scheme:ephemeralUnified Curves:P-256, P-384, P-521	KAS-ECC-SSC Sp800-56Ar3: (A2825)
Hash_DRBG	DRBG	Random Number Generation	Mode:SHA2-512	Hash DRBG: (A2825)

Table 6: Security Function Implementations

2.7 Algorithm Specific Information

The module’s AES-GCM implementation conforms to IG C.H scenario 2. The module uses the approved Hash_DRBG to generate the IV with a length of 96-bits. The entropy source producing the DRBG seed is located inside the module’s cryptographic boundary.

Steps to comply with the SP800-56Brev2 assurances can be found in section 11.3 Non-Administrator Guidance.

Compliance to FIPS 186-5 is met using FIPS 186-4 CAVP certs as allowed by additional comment 2 of IG C.K.

2.8 RBG and Entropy

The module employs a Hash_DRBG using a SHA-512 PRF. Per section 10.1.1.1 of [SP800-90A], the internal state of the Hash_DRBG is the V, C, and reseed counter. The Hash_DRBG is seeded by the physical entropy source which provides 256-bits of entropy to seed and reseed the DRBG during initialization and reseeding. The estimated amount of entropy per entropy output bit is ~0.6/bit. The DRBG internal state is not accessible by non-DRBG functions. All random values used by approved security functions, SSP generation, or SSP establishment method are provided by the Hash_DRBG.

Cert Number	Vendor Name
E114	Nuvoton

Table 7: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Nuvoton NTCES02	Physical	NPCX998HB0BX	1 bit	0.6 bits	The entropy pool is filled with random bits provided by an SP800-90B compliant entropy source whose noise source is from Ring Oscillators in hardware TRNG.

Table 8: Entropy Sources

2.9 Key Generation

The module generates Keys and SSPs in accordance with FIPS 140-3 IG D.H. The cryptographic module performs Cryptographic Key Generation (CKG) for asymmetric keys as per [SP800-133rev2] (vendor affirmed), compliant with [FIPS186-4] and using DRBG compliant with [SP800-90Arev1]. A seed (i.e., the random value) used in asymmetric key generation is obtained from [SP800-90Arev1] DRBG as described in Section 4 of [SP800-133rev2]. The key generation service for ECDSA, as well as the [SP 800-90Arev1] DRBG have been ACVT tested with algorithm certificates found in Table 3.

2.10 Key Establishment

The module provides the following key/SSP establishment services:

1. The module implements KAS-ECC-SSC EC Diffie-Hellman Shared Secret Computation compliant to [SP800-56Arev3] and IG D.F Scenario (2) path (1).
    - The shared secret computation provides between 128 and 256 bits of encryption strength.
2. Within the TOEPP, the module offers RSA key wrapping and unwrapping using KTS-OAEP-basic scheme. The implementation supports 2048 and 3072 modulus size, with both key encapsulation and un-encapsulation supported. The module does not implement key confirmation. See section 11.3 Non-Administrator Guidance.

The SSP establishment methodology provides 112 or 128 bits of encryption strength.

2.11 Industry Protocols

N/A for this module.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

The underlying logical interfaces of the module are the module's C language Application Programming Interfaces (APIs). All data input and data output, status ports and control ports are directed through the interface of the module's logical component, which are the APIs while the physical interface is considered the I/O ports of the sub-chip module through which the data input and data output, status output and control input traverse.

Physical Port	**Logical Interface(s)**	**Data That Passes**
I/O Ports	Data Input	Data inputs are provided in the variables passed in the API and callable service invocations, generally through caller-supplied buffers.
I/O Ports	Data Output	Data outputs are provided in the variables passed in the API and callable service invocations, generally through caller-supplied buffers.
I/O Ports	Control Input	Control inputs which control the operation of the module are provided through dedicated parameters.
I/O Ports	Status Output	Status output is provided in return codes and through messages. Documentation for each API lists possible return codes. A complete list of all return codes returned by the C language APIs within the module is provided in the header files and the API documentation. Messages are documented also in the API documentation.
Power Port	Power	Power interface is provided internally by TEOPP in which the cryptographic module is embedded.

Table 9: Ports and Interfaces

The module does not implement a Control Output Interface.

3.2 Trusted Channel Specification

The module does not transmit unprotected SSPs over any of its interfaces. All authentication data is transmitted between the module and the other endpoints in protected manner on both the contact and contactless interfaces.

3.3 Control Interface Not Inhibited

The control interface is inhibited while in the error state without any exceptions.

4 Roles, Services, and Authentication

4.1 Authentication Methods

FIPS 140-3 does not require authentication mechanism for level 1 modules. Therefore, the module does not implement an authentication mechanism.

N/A for this module.

4.2 Roles

The module supports two authorized roles: A Crypto Officer Role and a User Role. No support is provided for a Maintenance operator. The module does not implement a bypass mode nor concurrent operators.

Name	**Type**	**Operator Type**	**Authentication Methods**
Crypto Officer	Role	CO	None
User	Role	User	None

Table 10: Roles

When a device is delivered, the Crypto Officer is responsible for initializing the module i.e., configure the device by properly setting up key registers for storage of keys/CSPs. The Crypto Officer is implicitly assumed. The User can perform services from Table 5 and 5a only after the Crypto Officer takes possession by initializing it, thus creating data to be protected is generated. The Users of the module are software applications that implicitly assume the User Role when requesting any cryptographic services provided by the module.

4.3 Approved Services

The module only implements Approved security functions in an Approved mode. The Table 5 below lists services available. The module provides an approved service indicator by receiving a return code of “NCL_STATUS_OK to indicate that the service executed an approved security function.

NOTE: The module does not implement any non-Approved Algorithms in the Approved Mode of Operation (neither with nor without security claim). The module does not implement any non-approved security functions.

The abbreviations of the access rights to keys and SSPs have the following interpretation:

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 zeroises the SSP.

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
AES Encryption	Data Encryption	NCL STATUS OK	AES key and plain text	cipher text	AES-CBC AES-CCM AES-CFB128 AES-CTR AES-ECB AES-GCM AES-OFB	User - AES key: W,E
AES Decryption	AES Decryption	NCL STATUS OK	AES key and cipher text	plain text	AES-CBC AES-CCM AES-CFB128 AES-CTR AES-ECB AES-GCM AES-OFB	User - AES key: W,E
CMAC Message Authentication Code Generation	Message Authentication Code Generation	NCL STATUS OK	AES key and message	MAC	AES-CMAC	User - AES key: W,E
CMAC Message Authentication Code Verification	Message Authentication Code Verification	NCL STATUS OK	MAC and Message	"VALID" or "INVALID"	AES-CMAC	User - AES key: W,E
GMAC Message Authentication Code Generation	Message Authentication Code Generation	NCL STATUS OK	AES key, AAD	authentication tag	AES-GMAC	User - AES key: W,E
GMAC Message Authentication Code Verification	Message Authentication Code Verification	NCL STATUS OK	AES key, AAD, IV, tag	"PASS" or "FAIL"	AES-GMAC	User - AES key: W,E
HMAC Message Authentication Code Generation	Message Authentication Code Generation	NCL STATUS OK	HMAC key and message	MAC	HMAC	User - HMAC Key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Message Digest Generation	SHS Message Digest Generation	NCL STATUS OK	message	digest (hash value)	SHS	User
RSA Key Transport (key wrapping)	Key Wrapping using KTS-OAEP-basic	NCL STATUS OK	RSA public key and key to be wrapped	encrypted key	KTS-IFC (Wrap)	User - RSA KTS public key: W,E
RSA Key Transport (key unwrapping)	Key Un-wrapping using KTS-OAEP-basic	NCL STATUS OK	RSA private key and key to be un-wrapped	plaintext key	KTS-IFC (Unwrap)	User - RSA KTS private key: W,E
RSA Digital Signature Generation	Digital Signature Generation	NCL STATUS OK	RSA private key and message	signature	RSA SigGen Hash_DRBG	User - RSA Sig private key: W,E
RSA Digital Signature Verification	Digital Signature Verification	NCL STATUS OK	RSA public key and signature	True or False	RSA SigGen	User - RSA Sig public key: W,E
ECDSA Digital Signature Generation	Digital Signature Generation	NCL STATUS OK	ECDSA private key and message	signature	ECDSA SigGen Hash_DRBG	User - ECDSA private key: W,E
ECDSA Digital Signature Generation Component	Digital Signature Generation Component	NCL STATUS OK	ECDSA private key and message digest	signature	ECDSA SigGen Component Hash_DRBG	User - ECDSA private key: W,E
ECDSA Digital Signature Verification	Digital Signature Verification	NCL STATUS OK	ECDSA public key and signature	True or False	ECDSA SigVer	User - ECDSA public key: W,E
ECDSA Key Generation	Asymmetric Key Pair Generation	NCL STATUS OK	Curve size	generated private and public key pair	ECDSA KeyGen Hash_DRBG	User - ECDSA private key: G,R - ECDSA

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						public key: G,R
ECDSA Key Verification	Asymmetric Public Key Verification	NCL STATUS OK	Public Key	True or False	ECDSA KeyVer	User - ECDSA public key: W,E - ECDH public key (including intermediate key generation values): W,E
EC Diffie-Hellman Shared Secret Computation	Shared Secret Computation using Elliptic Curve Cryptography	NCL STATUS OK	received public key and possessed private key	shared secret	KAS-ECC-SSC	User - ECDH public key (including intermediate key generation values): W,E - ECDH private key (including intermediate key generation values): E - ECC Shared Secret: G,R
Random Number Generation	Deterministic Random Number Generation	NCL STATUS OK	Seed	random numbers	Hash_DRBG	User - Entropy Input String + Nonce: W - DRBG internal state (i.e., Hash_DRB G V and C values), Seed: G
Module Version Info	Outputs Module Name	N/A	None	Module Name + Module Version Number	None	User

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	+ Version Number					
SSP Zeroisation	zeroizes crypto function context and releases memory space	N/A	handle of crypto function context	zeroized and released memory space	None	User - AES key: Z - RSA KTS private key: Z - RSA KTS public key: Z - RSA Sig private key: Z - RSA Sig public key: Z - ECDSA private key: Z - ECDSA public key: Z - HMAC Key: Z - ECDH private key (including intermediate key generation values): Z - ECDH public key (including intermediate key generation values): Z - ECC Shared Secret: Z - Entropy Input String + Nonce: Z - DRBG internal state (i.e., Hash_DRB G V and C values), Seed: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Show-Status	Outputs Operational/ Error status of the module	N/A	None	Operational/Error status	None	User
Self-test	Executes on-demand self-test and outputs Pass/Fail status	NCL STATUS OK	None	Pass/Fail status	AES-CBC AES-CCM HMAC RSA SigGen RSA SigVer KTS-IFC (Wrap) KTS-IFC (Unwrap) ECDSA SigGen ECDSA SigVer SHS KAS-ECC-SSC Hash_DRBG	User - HMAC Key: E - AES key: E - RSA KTS private key: E - RSA KTS public key: E - RSA Sig private key: E - RSA Sig public key: E - ECDSA private key: E - ECDSA public key: E - ECDH private key (including intermediate key generation values): E - ECDH public key (including intermediate key generation values): E - DRBG internal state (i.e., Hash_DRB G V and C values), Seed: E

Table 11: Approved Services

4.4 Non-Approved Services

N/A for this module.

4.5 External Software/Firmware Loaded

N/A for this module.

5 Software/Firmware Security

5.1 Integrity Techniques

The module’s executable code is programmed in a masked ROM which is a type of Read-Only Memory (ROM) where content is programmed by the integrated circuit manufacturer during the silicon manufacturing (rather than by the Operator of the module). The memory technology is non reconfigurable memory as defined in IG 5.A, which will not have any change or degradation of data for a minimum of 10 years after manufactured date. As such, it is considered a hardware only module with a non-modifiable operational environment. The requirements of this area are not applicable to the module.

5.2 Initiate on Demand

The module does not implement any software/firmware integrity test. The requirements of this area are not applicable to the module.

6 Operational Environment

6.1 Operational Environment Type and Requirements

The HP Endpoint Security Controller operates in a non-modifiable operational environment. The module is programmed by the manufacturer during the silicon manufacturing (rather than by the user). It maintains its own memory region which can only be accessed by the module. There is no additional application present within the operating environment. The module does not spawn any cryptographic processes.

Type of Operational Environment: Non-Modifiable

7 Physical Security

7.1 Mechanisms and Actions Required

The HP Endpoint Security Controller Cryptographic Library is a Hardware cryptographic module in a single chip embodiment. More specifically, the module is considered a sub-chip cryptographic subsystem.

The module consists of production-grade components that include standard passivation techniques (e.g., a conformal coating applied over the module's circuitry to protect against environmental or other physical damage). The module does not implement a maintenance role and has no maintenance access interface.

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 12: Mechanisms and Actions Required

8 Non-Invasive Security

Currently, the non-invasive security is not required by FIPS 140-3 (see NIST SP 800-140F). The requirements of this area are not applicable to the module.

9 Sensitive Security Parameters Management

9.1 Storage Areas

The module does not provide persistent storage for keys/SSPs. Keys/SSPs are stored in memory only and are received for use by the module only at the request of the User firmware.

Storage Area Name	Description	Persistence Type
RAM	Stored in volatile memory	Dynamic

Table 13: Storage Areas

9.2 SSP Input-Output Methods

Keys/SSPs entered or output the module are electronically entered in plaintext form from the invoking User firmware running on the same device. No Keys/SSPs are entered or output from the module to outside the TOEPP. According to IG 2.3.B, Transferring SSPs including the entropy input between a sub-chip cryptographic subsystem and an intervening functional subsystem for Security Levels 1 and 2 on the same single chip is considered as not having Sensitive Security Parameter Establishment crossing the HMI of the sub-chip module per IG 9.5.A.

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
API input	Within the TOEPP	RAM	Plaintext	Automated	Electronic	
API output	RAM	Within the TOEPP	Plaintext	Automated	Electronic	

Table 14: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Keys and SSPs are explicitly zeroized automatically when structure associated with the cipher is deallocated or implicitly when the device is powered down thereby rendering the data irretrievable. Interface with the module is inhibited while zeroization is being performed. For Keys and SSPs explicitly zeroized automatically the successful completion of a requested service suffices as the implicit indicator that zeroisation has completed.

Zeroization Method	Description	Rationale	Operator Initiation
Module Reset	Power cycles the module	All SSPs in RAM are cleared after power reset	Initiated by operator
Deallocate Structure	Automatic zeroization when structure is deallocated	Wipes the SSP's contents in memory	Automatically by the module

Table 15: SSP Zeroization Methods

9.4 SSPs

The following summarizes the keys and Sensitive Security Parameters (SSPs) that are used by the cryptographic services implemented in the module. Modification of PSPs by unauthorized operators is prohibited.

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
AES key	AES Symmetric key used in Data Encryption, Data Decryption and Message Authentication Code Generation and verification	128, 192, 256 bits - 128, 192, 256 bits	Symmetric - CSP			AES-CBC AES-CCM AES-CFB128 AES-CMAC AES-CTR AES-ECB AES-GCM AES-GMAC
RSA KTS private key	Key Wrapping and Un-wrapping	2048, 3072 bits - 112 to 128 bits	Asymmetric key pair - CSP			KTS-IFC (Wrap) KTS-IFC (Unwrap)
RSA KTS public key	Key Wrapping and Un-wrapping	2048, 3072 bits - 112 to 128 bits	Asymmetric key pair - PSP			KTS-IFC (Wrap) KTS-IFC (Unwrap)
RSA Sig private key	Signature Generation and Verification	2048, 3072 bits - 112 to 128 bits	Asymmetric key pair - CSP			RSA SigGen RSA SigVer
RSA Sig public key	Signature Generation and Verification	2048, 3072 bits - 112 to 128 bits	Asymmetric key pair - PSP			RSA SigGen RSA SigVer
ECDSA private key	Key Verification, Signature Generation and Verification	P-256, P-384, P-521 curves - 112 to 256 bits	Asymmetric key pair - CSP	ECDSA KeyGen Hash_DRBG		ECDSA SigGen ECDSA SigVer
ECDSA public key	Key Verification, Signature Generation and Verification	P-256, P-384, P-521 curves - 112 to 256 bits	Asymmetric key pair - PSP	ECDSA KeyGen Hash_DRBG		ECDSA KeyVer ECDSA SigGen ECDSA SigVer

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
HMAC Key	Hashed Message Authentication Code Generation	112 bits or greater - 112 bits or greater	Symmetric - CSP			HMAC
ECDH private key (including intermediate key generation values)	ECDH Shared Secret Computation	P-256, P-384, P-521 curves - 112 to 256-bits	Asymmetric key pair - CSP	ECDSA KeyGen Hash_DRBG		KAS-ECC-SSC
ECDH public key (including intermediate key generation values)	ECDH Shared Secret Computation	P-256, P-384, P-521 curves - 112 to 256-bits	Asymmetric key pair - PSP	ECDSA KeyGen Hash_DRBG		ECDSA KeyVer KAS-ECC-SSC
ECC Shared Secret	ECDH Shared Secret Computation	112 to 256-bits - 112 to 256-bits	Asymmetric shared secret - CSP		KAS-ECC-SSC	
Entropy Input String + Nonce	Seed DRBG	256-bits - 256-bits	DRBG - CSP			Hash_DRBG
DRBG internal state (i.e., Hash_DRB G V and C values), Seed	Maintaining DRBG internal state	256-bits - 256-bits	DRBG - CSP			Hash_DRBG

Table 16: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES key	API input	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	
RSA KTS private key	API input	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	RSA KTS public key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
RSA KTS public key	API input	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	RSA KTS private key:Paired With
RSA Sig private key	API input	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	RSA Sig public key:Paired With
RSA Sig public key	API input	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	RSA Sig private key:Paired With
ECDSA private key	API input API output	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	DRBG internal state (i.e., Hash_DRB G V and C values), Seed:Derived From ECDSA public key:Paired With
ECDSA public key	API input API output	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	DRBG internal state (i.e., Hash_DRB G V and C values), Seed:Derived From ECDSA private key:Paired With
HMAC Key	API input	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	
ECDH private key (including intermediate key generation values)	API input API output	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	DRBG internal state (i.e., Hash_DRB G V and C values), Seed:Derived From ECDH public key (including intermediate key generation values):Paired With
ECDH public key (including intermediate key generation values)	API input API output	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	DRBG internal state (i.e., Hash_DRB G V and C values), Seed:Derived From ECDH private key

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					(including intermediate key generation values):Paired With
ECC Shared Secret	API output	RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	
Entropy Input String + Nonce		RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	
DRBG internal state (i.e., Hash_DRB G V and C values), Seed		RAM:Plaintext	Until deallocated or on module reset	Module Reset Deallocate Structure	Entropy Input String + Nonce:Derived From

Table 17: SSP Table 2

10 Self-Tests

10.1 Pre-Operational Self-Tests

Self-tests ensure that the module is not corrupted and that the cryptographic algorithms work as expected. While the module is executing the self-test, no services are not available, and input and output are inhibited. The module will boot only after successfully passing the HMAC-SHA2-512 and SHA2-256 CASTs. If an error is detected in any self-test, the module will enter the Error State.

N/A for this module.

The module is solely implemented in hardware (i.e., only contains executable code that is stored in non-reconfigurable masked ROM[1]). As such, the module does not perform any pre-operational software/firmware integrity test, but instead performs a Cryptographic Algorithm Self-Test on the HMAC-SHA2-512 and SHA2-256 algorithms when the module is powered on.

The module does not implement a pre-operational bypass test nor pre-operational critical functions test.

10.2 Conditional Self-Tests

The module conducts conditional cryptographic algorithm self-test prior to the first operational use of each cryptographic algorithm. The table below describe the conditional tests supported by the module.

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-512 (A2825)	HMAC-SHA2-512 MAC Generation KAT	KAT	CAST	NCL STATUS OK	MAC Generation	Performed when the module is powered on
SHA2-256 (A2825)	SHA2-256 Message Digest KAT	KAT	CAST	NCL STATUS OK	Message Digest	Performed when the module is powered on
AES-CCM (A2825)	AES-CCM Encryption KAT using 128-bit key	KAT	CAST	NCL STATUS OK	AES Encryption	Prior to the first operational use of the algorithm
AES-CBC (A2825)	AES-CBC Decryption KAT using 128-bit key	KAT	CAST	NCL STATUS OK	AES Decryption	Prior to the first operational use of the algorithm

[1] A masked ROM is a type of Read-Only Memory (ROM) where content is programmed by the integrated circuit manufacturer during the silicon manufacturing.

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
RSA SigGen (FIPS186-4) (A2825)	Signature Generation KAT with 2048-bit key and SHA2-256	KAT	CAST	NCL STATUS OK	RSA Signature Generation	Prior to the first operational use of the algorithm
RSA SigVer (FIPS186-4) (A2825)	PKCS#1 v1.5 Signature Verification KAT with 2048 -bit key and SHA2-256 PKCS#1 v1.5	KAT	CAST	NCL STATUS OK	RSA Signature Verification	Prior to the first operational use of the algorithm
KTS-IFC (A2825)	KTS-OAEP-basic Encryption/Decryption KAT with 2048 -bit key and SHA2-256	KAT	CAST	NCL STATUS OK	KTS-OAEP-basic Encryption and Decryption	Prior to the first operational use of the algorithm
ECDSA KeyGen (FIPS186-4) (A2825)	Pairwise consistency test	PCT	PCT	NCL STATUS OK	Pairwise consistency test	Performed upon generation of a new ECDSA key pair
ECDSA SigGen (FIPS186-4) (A2825)	ECDSA Signature Generation KAT with P-256 curve and SHA2-256	KAT	CAST	NCL STATUS OK	ECDSA Signature Generation	Prior to the first operational use of the algorithm
ECDSA SigVer (FIPS186-4) (A2825)	ECDSA Signature Verification KAT with P-256 curve and SHA2-256	KAT	CAST	NCL STATUS OK	ECDSA Signature Verification	Prior to the first operational use of the algorithm
KAS-ECC-SSC Sp800-56Ar3 (A2825)	ECDH shared secret computation KAT with P-256 curve	KAT	CAST	NCL STATUS OK	ECDH shared secret computation	Prior to the first operational use of the algorithm
Hash DRBG (A2825)	Hash_DRBG random number generation KAT using predefined data.	KAT	CAST	NCL STATUS OK	Hash_DRBG random number generation	Prior to the first operational use of the algorithm

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
ENT	RCT (Repetition Count Test)	RCT	CAST	NCL STATUS OK	Continuous Health Test	Performed when the module is powered on
ENT	APT (Adaptive Proportion Test)	APT	CAST	NCL STATUS OK	Continuous Health Test	Performed when the module is powered on

Table 18: Conditional Self-Tests

The module does not implement a Software/Firmware Load Test, Manual Entry Test, Conditional Bypass Test nor Conditional Critical Functions Test.

10.3 Periodic Self-Test Information

During runtime, operators can initiate the conditional self-tests on demand by calling NCL_MISC_SelfTest and passing the algorithm as an argument.

The module’s entropy source is powered on only momentarily to seed the module’s SP800-90B DRBG. The module performs ENT health tests defined in Section 4 of SP800-90B on the generated output prior to seeding the SP800-90B DRBG. After completing its execution, the entropy source powers down.

N/A for this module.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-512 (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the algorithm as an argument
SHA2-256 (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the algorithm as an argument
AES-CCM (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the algorithm as an argument
AES-CBC (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
				algorithm as an argument
RSA SigGen (FIPS186-4) (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the algorithm as an argument
RSA SigVer (FIPS186-4) (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the algorithm as an argument
KTS-IFC (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the algorithm as an argument
ECDSA KeyGen (FIPS186-4) (A2825)	PCT	PCT	N/A	N/A
ECDSA SigGen (FIPS186-4) (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the algorithm as an argument
ECDSA SigVer (FIPS186-4) (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the algorithm as an argument
KAS-ECC-SSC Sp800-56Ar3 (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the algorithm as an argument
Hash DRBG (A2825)	KAT	CAST	On demand	By calling NCL_MISC_SelfTest and passing the algorithm as an argument

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
ENT	RCT	CAST	On demand	Powering the chip off and on
ENT	APT	CAST	On demand	Powering the chip off and on

Table 19: Conditional Periodic Information

10.4 Error States

For any of the conditional self-tests, the module enters an error state upon failing the self-test. A failure in the conditional CAST or conditional PCT results in “NCL_STATUS_FAIL”. Likewise, a failure of the ENT health tests will result in an “ENTROPY_SRC_ERROR” status returned to the user. When in the error state, no cryptographic services are provided, control and data output is prohibited. The only method to clear this error state is to power cycle the device and then successfully pass the conditional self-tests.

Name	Description	Conditions	Recovery Method	Indicator
NCL_STATUS_FAIL	When in this error state, no cryptographic services are provided, control and data output is prohibited.	Failure in conditional self-test (conditional CAST or conditional PCT)	The only method to clear this error state is to power cycle the device and then successfully pass the conditional self-tests.	NCL_STATUS_FAIL
ENTROPY_SRC_ERROR	When in this error state, no cryptographic services are provided, control and data output is prohibited.	Failure of the ENT health test	The only method to clear this error state is to power cycle the device and then successfully pass the conditional self-tests.	ENTROPY_SRC_ERROR

Table 20: Error States

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

As explained in Section 10.1 Pre-Operational Self-Tests, the module is placed in a masked ROM by manufacturer during the silicon manufacturing. The module is delivered as part of the Nuvoton NPCX998HB0BX platform (listed in Table 2). During manufacturing – each chip is tested to make sure the masked ROM was manufactured correctly; this is done using CRC32 algorithm on the entire masked ROM code on each device before it is shipped out.

During execution – As part of the device boot process, the code is verified by a dedicated hardware inside the chip that checks every byte of code compared to a known parity bit. If any byte fails, the parity test then an internal error is generated; the error is handled by the application (User) firmware.

11.2 Administrator Guidance

The module is configured to be operational by default. If the device starts up successfully and has successfully passed the HMAC-SHA2-512 and SHA2-256 CAST, it is operating correctly and can begin servicing User requests.

11.3 Non-Administrator Guidance

The entity using the IUT must obtain required assurances listed in section 6.4 of SP 800-56BRev2 by performing the following steps:

1. The entity requesting the RSA key unwrapping (un-encapsulation) service from the module, shall only use an RSA private key that was generated by an active FIPS validated module that implements FIPS 186-5 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, the entity using the module, shall verify the validity of the peer's public key using any method specified in section 6.4.2.1 of the SP 800-56BRev2.

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.

11.4 Design and Rules

N/A for this module.

11.5 Maintenance Requirements

N/A for this module.

11.6 End of Life

Once the module reaches its end-of-life stage (End of Life (EOL) date for the HP Endpoint Security Controller Cryptographic Library is 10 years from manufacturing date) or sanitation is initiated by the module’s Operator, it is the Operator’s responsibility to clear all existing SSPs from the module. This can be achieved by either performing a full device reset, or by explicitly invoking the following sequence of APIs to clear the data from all modules:

- NCL_SHA_Clear - For each of existing SHA and HMAC contexts

- NCL_DRBG_Clear - For each of existing DRBG contexts
- NCL_AES_Clear - For each of existing AES contexts
- NCL_RSA_Clear - For each of existing RSA contexts
- NCL_ECC_Clear - For each of existing ECDSA and ECDH contexts

12 Mitigation of Other Attacks

The module does not implement security mechanisms to mitigate other attacks.

Glossary and Abbreviations

AES	Advanced Encryption Standard
ACVP	Algorithm Certification Validation Program
CBC	Cipher Block Chaining
CAST	Cryptographic Algorithm Self-Test
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CFB	Cipher Feedback
CMAC	Cipher-based Message Authentication Code
CMVP	Cryptographic Module Validation Program
CSP	Critical Security Parameter
CTR	Counter Mode
DRBG	Deterministic Random Bit Generator
ECB	Electronic Code Book
ECC	Elliptic Curve Cryptography
ENT	Entropy Source
EOL	End Of Life
FIPS	Federal Information Processing Standards Publication
GCM	Galois Counter Mode
HMAC	Hash Message Authentication Code
KAS	Key Agreement Scheme
KAT	Known Answer Test
MAC	Message Authentication Code
NIST	National Institute of Science and Technology
OFB	Output Feedback
PSS	Probabilistic Signature Scheme
RSA	Rivest, Shamir, Addleman
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SSC	Shared Secret Computation
TOEPP	Tested Operational Environment’s Physical Perimeter

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 2024 https://csrc.nist.gov/CSRC/media/Projects/cryptographic-module-validation-program/documents/fips 140-3/FIPS 140-3 IG.pdf
FIPS180-4	Secure Hash Standard (SHS) March 2012 http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf
FIPS186-5	Digital Signature Standard (DSS) February 2023 http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-5.pdf
FIPS197	Advanced Encryption Standard November 2001 http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf
FIPS198-1	The Keyed Hash Message Authentication Code (HMAC) July 2008 http://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 http://www.ietf.org/rfc/rfc3447.txt
RFC3394	Advanced Encryption Standard (AES) Key Wrap Algorithm September 2002 http://www.ietf.org/rfc/rfc3394.txt
RFC5649	Advanced Encryption Standard (AES) Key Wrap with Padding Algorithm September 2009 http://www.ietf.org/rfc/rfc5649.txt
SP800-38A	NIST Special Publication 800-38A - Recommendation for Block Cipher Modes of Operation Methods and Techniques December 2001 http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf
SP800-38B	NIST Special Publication 800-38B - Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication May 2005 http://csrc.nist.gov/publications/nistpubs/800-38B/SP_800-38B.pdf

SP800-38C	NIST Special Publication 800-38C - Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality May 2004 http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38c.pdf
SP800-38D	NIST Special Publication 800-38D - Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC November 2007 http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf
SP800-38F	NIST Special Publication 800-38F - Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping December 2012 http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38F.pdf
SP800-56Arev3	NIST Special Publication 800-56A Revision 3 - Recommendation for Pair Wise Key Establishment Schemes Using Discrete Logarithm Cryptography April 2018 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Ar3.pdf
SP800-56Brev2	Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography March 2019 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Br2.pdf
SP800-90Ar1	NIST Special Publication 800-90A - Revision 1 - Recommendation for Random Number Generation Using Deterministic Random Bit Generators June 2015 http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90Ar1.pdf
SP800-90B	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
SP800-133rev2	NIST Special Publication 800-133 - Recommendation for Cryptographic Key Generation December 2012 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-133r2.pdf
SP800-140Br1	NIST Special Publication 800-140Br1 - CMVP Security Policy Requirements November 2023 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-140Br1.pdf