Chunghwa Telecom laboratories

HiPKI SafGuard 2000 HSM

FIPS 140-3 Non-Proprietary Security Policy

Document Version: 2.13
Last update: 2026-08-13

中華電信研究院
Chunghwa Telecom Laboratories

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

The Chunghwa Telecom Laboratories HiPKI SafGuard 2000 HSM is a hardware security module used in a PKI system. The hardware security module (HSM) provides rapid cryptographic functionality to the operators of the system. Crypto Officers (COs) and Users are authenticated using a smart card and password. The smart card reader is located within the boundary of the module. The boundary of the HiPKI SafGuard 2000 HSM is the physical hardware box itself. All cryptographic module components are included inside this boundary.

1.2 Security Levels

The module meets the overall requirements for FIPS 140-3 Level 3.

Section	Title	Security Level
1	General	3
2	Cryptographic module specification	3
3	Cryptographic module interfaces	3
4	Roles, services, and authentication	3
5	Software/Firmware security	3
6	Operational environment	N/A
7	Physical security	3
8	Non-invasive security	N/A
9	Sensitive security parameter management	3
10	Self-tests	3
11	Life-cycle assurance	3
12	Mitigation of other attacks	N/A
	Overall Level	3

Table 1: Security Levels

1.3 Additional Information

This Security Policy describes how this module complies with the eleven sections of the standard. For more information on the FIPS 140-3 standard and validation program please refer to the NIST website at https://csrc.nist.gov/projects/cryptographic-module-validation-program.

For more information about Chunghwa Telecom Co. Ltd. please visit http://www.chttl.com.tw.

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:
This is a non-proprietary security policy developed for the Chunghwa Telecom Ltd. HiPKI SafGuard 2000 HSM. It describes how the HiPKI SafGuard 2000 meets the requirements for a FIPS 140-3 level 3 validation as specified in the FIPS 140-3 standard. This Security Policy is part of the evidence documentation package to be submitted to the validation lab.

Module Type: Hardware

Module Embodiment: Multi-Chip Standalone

Cryptographic Boundary:

The physical cryptographic boundary is defined as the module case, and the module runs on a limited operating environment.

Tested Operational Environment's Physical Perimeter (TOEPP) :

The hardware box determines the physical boundary of the cryptographic module. The HSM is comprised of a cryptographic acceleration engine, peripheral management module, a power source module, a tamper circuit, a smart card socket, a keypad, and an OLED display.

A photograph of the HiPKI SafGuard 2000 HSM which is approximately to scale, is included below.

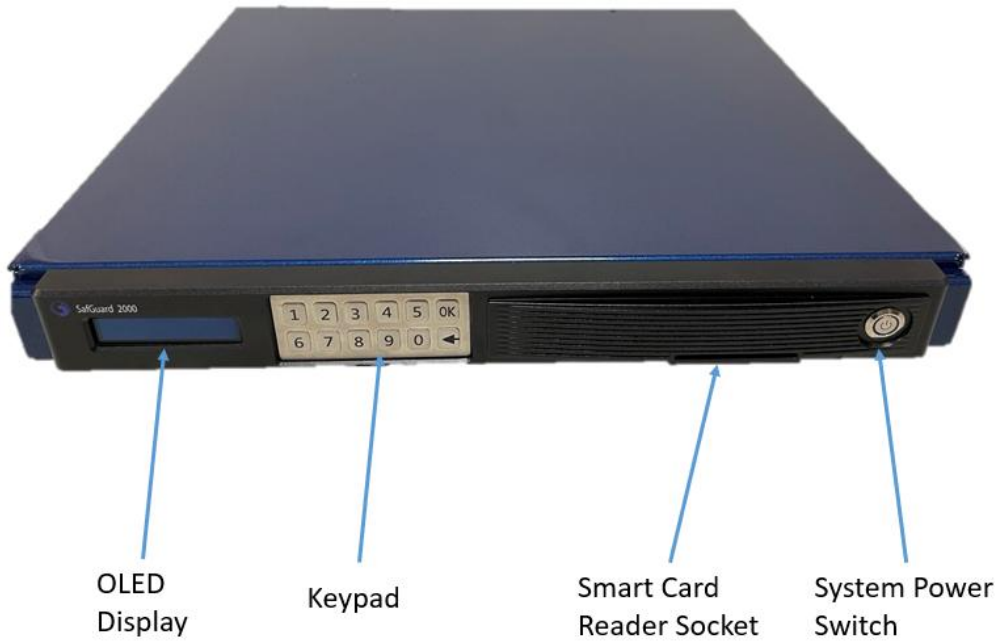

Figure 1: Front view of HiPKI SafGuard 2000 HSM

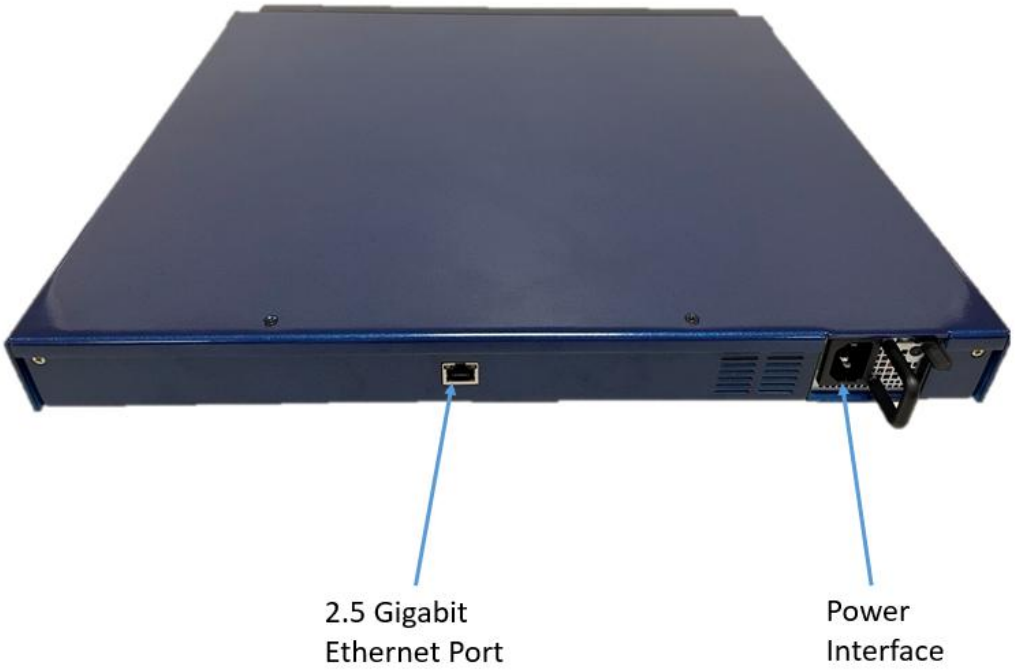

Figure 2: Rear view of HiPKI SafGuard 2000 HSM

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
HiPKI SafGuard 2000 HSM	HSM-HW-10	HSM-FW-10	Marvell	

Table 2: Tested Module Identification – Hardware

2.3 Excluded Components

None

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Types of Services	Status Indicator
FIPS mode	Only FIPS mode	Approved	Approved mode is indicated by the OLED screen displaying - FIPS mode

Table 3: Modes List and Description

The module operates only in a FIPS approved mode. Approved mode is indicated by the OLED screen displaying “FIPS mode” when the module is powered on and passed the firmware integrity checks and KAT tests.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CBC, AES-ECB	A2892	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW, AES-KWP	A2894	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
ECDSA KeyGen (FIPS186-4), ECDSA KeyVer (FIPS186-4)	A2894	Curve - P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4), ECDSA SigVer (FIPS186-4)	A2894	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-256	FIPS 186-4
HMAC DRBG	A2894	Mode - SHA2-256	SP 800-90A Rev. 1
HMAC-SHA2-256	A2892	Key Length - Key Length: 256-448 Increment 8	FIPS 198-1
KTS-IFC	A2894	Modulo - 2048, 3072, 4096 Scheme - KTS-OAEP-basic -	SP 800-56B Rev. 2
RSA KeyGen (FIPS186-4)	A2894	Key Generation Mode - B.3.3 Modulo - 2048, 3072, 4096 Primality Tests - Table C.2 Private Key Format - Standard	FIPS 186-4
RSA SigGen (FIPS186-4), RSA SigVer (FIPS186-4)	A2894	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
SHA2-256	A2892	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 4: Approved Algorithms

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Symmetric:AES Asymmetric:RSA, ECDSA	using output from DRBG	SP 800-133Rev2 and IG D.I.

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

Security function implementations are listed below which is associated to the approved algorithms defined in Section 2.5 Algorithms.

Name	Type	Description	Properties	Algorithms
KeyWrap	KTS-Wrap	SP 800-38D and SP 800-38F. KTS (key wrapping and unwrapping) per IG D.G.	Key size:128, 192, and 256-bit keys providing 128, 192, or 256 bits of encryption strength	AES-KW: (A2894) Key size: 128, 192, 256 AES-KWP: (A2894) Key size: 128, 192, 256
KeyTransport	KTS-Encap	SP 800-56Brev2. KTS-IFC (key encapsulation and un-encapsulation) per IG D.G.	Key size:2048, 3072, 4096-bit keys with 112-150-bit key strength	KTS-IFC: (A2894) Key size: 2048, 3072, 4096
GenAES	CKG	Generate AES Key	Key size:128, 192, and 256-bit keys providing 128, 192, or 256 bits of encryption strength	HMAC DRBG: (A2894) Key size: 128, 192, 256
GenRSA	CKG	Generate asymmetric keys (RSA)	Key size:2048, 3072, 4096-bit keys with 112-150-bit key strength	RSA KeyGen (FIPS186-4): (A2894) Key size: 2048, 3072, 4096
GenECDSA	CKG	Generate asymmetric keys (ECDSA)	Key size:P224, P256, P384, P521 curves with 112-256-bit key strength	ECDSA KeyGen (FIPS186-4): (A2894) Curve: P224, P256, P384, P521
MsgAuth	MAC	Verify audit log	Key size:256-bit	HMAC-SHA2-256: (A2892) Key size: 256-bit
GenRand	DRBG	Get random number		HMAC DRBG: (A2894)
ECDSASignVer	DigSig-SigGen DigSig-SigVer	Digital Signature generation and verify	Key size:P224, P256, P384, P521 curves with 112-256-bit key strength	ECDSA SigGen (FIPS186-4): (A2894) Curves : P224, P256, P384, P521 ECDSA KeyVer (FIPS186-4): (A2894) Curves: P224, P256, P384, P521
RSASignVer	DigSig-SigGen DigSig-SigVer	Digital signature generation and verify	Key size:2048, 3072, 4096-bit keys with 112-150-bit key strength	RSA SigGen (FIPS186-4): (A2894) Key size: 2048, 3072, 4096-bit RSA SigVer (FIPS186-4):

Name	Type	Description	Properties	Algorithms
				(A2894) Key size: 2048, 3072, 4096-bit

Table 6: Security Function Implementations

2.7 Algorithm Specific Information

None

2.8 RBG and Entropy

Cert Number	Vendor Name
E8	Red Hat, Inc.

Table 7: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
SP800-90B compliant ENT (NP)	Non-Physical	Red Hat Enterprise Linux 8 DRBG	256 bits	256 bits	None

Table 8: Entropy Sources

A NIST approved deterministic random bit generator based on a hash function as specified in SP800-90Arev1 is used. The Approved DRBG used for random number generation is a HMAC_DRBG with prediction resistance. The random numbers used for key generation are all generated by HMAC_DRBG. Per section 10.1.2.1 of SP800-90Arev1, the internal state of HMAC_DRBG is the value V and Key.
The deterministic random bit generators are seeded by "*reseed*". The *reseed* function is invoked automatically in the module when the requests counter of random numbers is greater than or equal to the reseed interval in order to adjust the internal state of HMAC_DRBG. The non-physical entropy source outside the physical cryptographic boundary provides the random bits. The output of entropy pool provides 256-bits of entropy to seed and reseed SP800-90B DRBG during initialization (seed) and reseeding (reseed). No assurance of the minimum strength of generated SSPs.

2.9 Key Generation

To generate RSA and ECDSA keys for application purpose, the module implements asymmetric key generation services compliant with FIPS 186-4 and performs Cryptographic Key Generation (CKG) for asymmetric keys as per section 5.1 of SP800-133rev2(vendor affirmed), in accordance with FIPS 140-3 IG D.H. To generate AES keys for application purpose, Session key, Master key, Clone key and Audit key, the module implements symmetric key generation services compliant with section 6.1 of SP800-133rev2 (vendor affirmed), in accordance with FIPS 140-3 IG D.H. The random value used in both symmetric and asymmetric key generation is obtained from the DRBG compliant with SP800-90Arev1.

2.10 Key Establishment

The module provides RSA OAEP public-key encryption scheme compliant with SP800-56Brev2, in accordance with scenario 1 of IG D.F; the module provides SP800-38F approved Key Transport Scheme (KTS) according to IG D.G. The key transport method is provided using an AES-KWP key wrapping algorithm.
According to SP800-57, RSA OAEP public-key encryption scheme provides 112 bits of encryption strength and AES key wrapping provides 128 bits of encryption strength.

2.11 Industry Protocols

None

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

The table below describes the relationship between the interfaces.

Physical Port	Logical Interface(s)	Data That Passes
Ethernet, Smart Card Interface, and Front Panel Keypad	Data Input	plaintext data, ciphertext data, SSPs, smart card PIN
Ethernet and Smart Card Interface	Data Output	plaintext data, ciphertext data, SSPs, status
Ethernet	Control Input	Command
Ethernet and OLED display	Status Output	Status
AC power source interfaces	Power	N/A
Power switch	Control Input	N/A

Table 9: Ports and Interfaces

3.2 Trusted Channel Specification

RSA KTS-OAEP algorithm is applied to protect the session key which is used to implement a trusted channel via the internet. RSA Client OAEP key pair is generated from the host and the public key is transmitted to the module during the initialization process. When operator starts to connect to the module in FIPS mode, the first step is to generate connection and the Host role signs in to exchange the session key which is an AES 128 key wrapped with public key. The trusted channel is established when the wrapped session key is obtained by the Host and unwrapped with the private key. The role should be changed to CO/User to operation CO/User services. Messages with SSP information are encrypted with AES-128-CBC (Cert. # A2892). The module API always checks the encrypted messages in this connection without exception. The session key is cleared after disconnection, and the trusted channel is closed. To prevent any unauthorized tampering, the session key is re-generated every time when starting a new connection.

4 Roles, Services, and Authentication

The module supports CO role, User role and Host role. The HiPKI SafGuard 2000 HSM implements two authentication methods, one is key transport scheme for trusted channel, and the other is two-factor identity authentication. Two-factor identity–based authentication using a combination of signature and passwords. Identity-based authentication occurs by two methods. One is to enter a smart card and 8 digits PIN for each smart card and perform a signature with a private key store on the card, the other is to do a signature from a software private key. Of at least 2 COs and up to a maximum of 3 COs. Each CO, upon successful entry of a password, transmits the signature in the HSM to authenticate to the role.

4.1 Authentication Methods

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
Two-factor identity-based authentication (Smart Card)	Signature and 8 digits PIN and 12 printable characters.	RSA SigVer (FIPS186-4) (A2894)	Probability of 1/(2^112 X 10^8) in a single random attempt that the PIN must be 8 digits long.	Probability of 3/(2^112 X 10^8) in multiple random attempts in a minute since the account is locked after 3 failed login attempts
Two-factor identity-based authentication (Software)	Signature and 12 printable characters.	RSA SigVer (FIPS186-4) (A2894)	Probability of 1/(2^112 X 95^12) in a single random attempt that the password policy requires 12 printable characters	Probability of 3/(2^112 X 95^12) in multiple random attempts in a minute since the account is locked after 3 failed login attempts
Key transport authentication		RSA-OAEP Encryption Decryption	Probability of 1/2^112 in a single random attempt.	Probability of 60,000,000/2^112 in multiple random attempts in a minute.

Table 10: Authentication Methods

4.2 Roles

Name	Type	Operator Type	Authentication Methods
CO	MultiFactorIdentity	Crypto Officer	Two-factor identity-based authentication (Smart Card) Two-factor identity-based authentication (Software)
User	MultiFactorIdentity	User	Two-factor identity-based authentication (Smart Card) Two-factor identity-based authentication (Software)
Host	Identity	Host application	Key transport authentication

Table 11: Roles

4.3 Approved Services

The approved services are described below:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Verify audit log	Verify audit log	API returns 0	Audit log	OK / Fail	MsgAuth	CO - Audit log key: E
Integrity test on demand	Integrity test on demand	API returns 0	N/A	Success / Error	RSASignVer	CO
Self-test on demand	Self-test on demand	API returns 0	N/A	Success / Error	None	CO

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Zeroize slot	Delete all object on slot	API returns 0	Slot ID	OK / Fail	None	CO - Client OAEP key: Z - Clone key: Z - Session key: Z - Application key RSA: Z - Application key ECDSA: Z - Application key AES: Z - Security officer's public key: Z - User's public key: Z - P11 slot SO password salt hash: Z - P11 slot User password salt hash: Z - Entropy Input String: Z - DRBG Seed, internal state: V value and Key: Z
Export CloneWhole HSM	Export whole HSM	API returns 0	None	Encryption file	KeyWrap	CO - Clone key: R
Import CloneWhole HSM	Import whole HSM	API returns 0	Encryption file	OK / Fail	KeyWrap	CO - Clone key: W
Export CloneSlot	Clone and export a slot	API returns 0	Slot ID	Encryption file	KeyWrap	CO - Clone key: R
Import CloneSlot	Import a clone slot	API returns 0	Encryption file	OK / Fail	KeyWrap	CO - Clone key: W
Switch to Initialization state & Erase ALL keys	Set module to uninitialized state	API returns 0	N/A	OK / Fail	None	CO - Master key: Z
Generate CO Quorum Token	Generate CO authentication quorum token	API returns 0	CO ID	CO Quorum Token	GenRand	CO
Sign CO Token (with Smartcard)	CO send the signature for the CO Quorum Token to the module	API returns 0	CO ID, signature	OK / Fail	ECDSASignVer RSASignVer	CO - Application key RSA: R - Application key ECDSA: R
Generate key	Generate a cryptographic key (AES, RSA, ECDSA)	API returns 0	Slot ID, key type, key ID	OK / Fail	GenAES GenRSA GenECDSA	User - Client OAEP key: W - Session key: W

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Wrap key	Export an Assigned key using a cryptographic key (AES)	API returns 0	Slot ID, key type ,key ID, KEK key type , KEK key ID, encryption key data	Wrapped key data	KeyWrap	User - Session key: R
Unwrap key	Import an Assigned key from a cryptographic key (AES)	API returns 0	Slot ID, key type, key ID, KEK key type, KEK key ID, encryption key data	OK / Fail	KeyWrap	User - Session key: R
Encrypt	Perform encrypt operation on user supplied data object	API returns 0	Slot ID, key type, key ID, plain data	Encrypted data	None	User - Application key AES: E
Decrypt	Perform decrypt operation on user supplied data object	API returns 0	Slot ID, key type, key ID, encrypted data	Plain data	None	User - Application key AES: E
Sign	Generate signature over user supplied data	API returns 0	Slot ID, key type, key ID, data	Signature	ECDSASignVer RSASignVer	User - Application key RSA: R - Application key ECDSA: R
Verify	Validate signature over user supplied data	API returns 0	Slot ID, key type, key ID, signature	OK / Fail	ECDSASignVer RSASignVer	User - Application key RSA: R - Application key ECDSA: R
Generate Key Quorum Token	Generate a quorum token if cryptographic key need quorum-controlled operation	API returns 0	Slot ID, key type, key ID	Key Quorum Token	GenRand	User
Sign Key Token (with Smartcard)	User sends the signature for the key Quorum Token to the module	API returns 0	Slot ID, key type, key ID, signature	OK / Fail	ECDSASignVer RSASignVer	User - Application key RSA: R - Application key ECDSA: R
View Hardware and firmware version	Show hardware/firmware version and status of the module	API returns 0	None	Hardware / firmware version and status of the module	None	Unauthenticated
Zeroization	Zeroize all unprotected SSPs	OLED displays	None	None	None	CO - Master key: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		"Zeroization + time()"				
Software update	Load module firmware	OLED displays "Delete all keys + Initial Mode"	Update image	OK / Fail	RSASignVer	CO
Get handle	Generate trusted channel	API returns 0	Server IP, N, D, E value of the OAEP Client key, Name and ID	Handle	KeyTransport	Host
Release handle	Delete trusted channel	API returns 0	Handle	OK / Fail	None	Host
CO login	CO login into the system	API returns 0	CO ID, signature, and password	OK / Fail	RSASignVer	CO - P11 slot SO password salt hash: R
User login	User login into the system	API returns 0	User ID, signature, and password	OK / Fail	RSASignVer	User - P11 slot User password salt hash: R
Generate Clone key	Generate Clone key to encrypt the whole HSM or a slot information	API returns 0	N/A	OK / Fail	GenRand	CO - Clone key: W
Create user	Add a user account	API returns 0	User ID, User password, User's public key	OK / Fail	RSASignVer	CO - P11 slot User password salt hash: W

Table 12: Approved Services

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 = Zeroize: The module zeroizes the SSP.

4.4 Non-Approved Services

N/A for this module.

4.5 External Software/Firmware Loaded

The only firmware to be loaded to the module is HiPKI SafGuard 2000 service firmware. To verify the firmware integrity, an RSA 2048 Manufacture key pair is generated outside the boundary. Manufacture public key is imported to the module in advance before shipping from the factory. The firmware file to be loaded is signed with Manufacture private key, the signed data and the firmware version information are attached to the firmware file.
The module supports two COs authentication to operate HiPKI SafGuard 2000 service firmware loaded (update) process. At first, the module imported the whole firmware file. After verifying the signed data by Manufacture public key and checking the firmware version information is not older than the previous version, the module updates the firmware version and store the firmware file and the signed data. The newest version of the firmware will be loaded after re-staring up the module. Then the module switch to Initialization state and the OLED will display “Initial Mode.” The module enters the error state if any self-test fails.

5 Software/Firmware Security

5.1 Integrity Techniques

At start up, the HiPKI SafGuard 2000 HSM firmware executable code is verified by an RSA 2048 Manufacture public key and compared to a firmware signed data stored in the module. The test fails if the calculated value does not equal the stored value.

5.2 Initiate on Demand

The integrity test can be initiated by COs who invoke function to verify the integrity of the firmware code.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Non-Modifiable

7 Physical Security

7.1 Mechanisms and Actions Required

The HiPKI SafGuard 2000 HSM is defined as a multi-chip standalone module. The module consists of production grade components, which include standard passivation techniques. The HiPKI SafGuard 2000 HSM is being validated against FIPS 140-3 level 3. The module’s enclosure is made of an aluminum alloy and is opaque to the visible spectrum. The HiPKI SafGuard 2000 HSM has a mechanism for tamper detection and response, which zeroizes Master key stored internally to the module in eMMC if an attempt is made to open the enclosure. The tamper detection and response circuit are backed up by battery housed

internally in the HiPKI SafGuard 2000 HSM in case of power failure to the module. If the top casing is restored after an attempt is made to remove it, the OLED will continue to display “detect tamper” and the module will return to the factory settings when the power is restarted. The module’s operating temperature range is from 0˚C to 68˚C.

Mechanism	Inspection Frequency	Inspection Guidance
Tamper evident seals	Weekly	The tamper-evident seals are installed only by the module manufacturer. After delivering to the customer, a system administrator is required to check serial number of each tamper-evident seal and inspect the tamper-evident seals if the module has been interfered with. Upon viewing any signs of tampering, the administrator is required to zeroize Master key and return the module to the factory.

Table 13: Mechanisms and Actions Required

7.2 User Placed Tamper Seals

Number: 3 tamper evident seals

Placement:

The case has a removable cover on the top. The cover is fixed to the case through eight screws. Three screws locate on the left and the other three place on the right of the chassis, respectively. The rest of the two screws place on the rear top of the chassis. The junctures between the case and the cover are protected with three tamper-evidence seals. One seal is located on the front top; another two seals cover the two screws placed on the rear top. The photo of tamper-evidence seal is illustrated, there is a serial number on it which is uniquely identifiable. The picture below shows the position of the seals on HiPKI SafGuard 2000 HSM.

Surface Preparation: None

Operator Responsible for Securing Unused Seals: None

Part Numbers: FS-4120

Figure 3: Tamper-evidence seal

Figure 4: Tamper-evident seals placement

7.5 EFP/EFT Information

There is a temperature sensor module installed in the HiPKI SafGuard 2000. While the operating temperature drops below specified limits (0˚C), the Master key stored internally to the module in eMMC will be zeroized. When the temperature reaches 68˚C, the HiPKI SafGuard 2000 will shut down.
HiPKI SafGuard 2000 HSM system will shut down after the voltage signal being pulled HIGH to 264V or being pulled LOW to 90V.
The EFP information is listed below.

Temp/Voltage Type	Temperature or Voltage	EFP or EFT	Result
LowTemperature	0°C	EFP	Zeroization
HighTemperature	68°C	EFP	Shutdown
LowVoltage	90V	EFP	Shutdown
HighVoltage	264V	EFP	Shutdown

Table 14: EFP/EFT Information

7.6 Hardness Testing Temperature Ranges

The hardness testing temperature ranges is displayed by the table below.

Temperature Type	Temperature
LowTemperature	-0.08°C
HighTemperature	68.9°C

Table 15: Hardness Testing Temperatures

8 Non-Invasive Security

The module claims no non-invasive security techniques.

9 Sensitive Security Parameters Management

The following section indicates the key generation method, usage, and storage.

Master key stored in eMMC will be zeroized under the following circumstances:

1. if the tamper response switch is activated
2. if the CO returns the module to the “initialization” state as it is referred to by HiPKI SafGuard 2000 Administrator Guidance
3. when the firmware loaded(update) process is executed

The HiPKI SafGuard 2000 HSM returns to the same state (initialization state) as it was when shipped from the factory and must be reconfigured to continue operation. Two internal independent actions, which means two CO authentication, are always required to output keys or CSPs in cipher text. Keys are not output in plaintext.
The cryptographic module could distinguish between data, control information and power for input, and data, control information and status information for output. All input commands and control data (including function calls and manual controls such as switches and keypads) used to control the operation of a cryptographic module shall only enter via the "control input" interface. For each SSPs operations, the user authentication is needed.

9.1 Storage Areas

The SSP storage techniques are displayed in the following table.

Storage Area Name	Description	Persistence Type
eMMC	None	Static
Flash	None	Static
SDRAM	None	Dynamic

Table 16: Storage Areas

9.2 SSP Input-Output Methods

The SSPs are entered and output in different methods, the following table lists the SSP input-output methods.

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Master key split Knowledge (Import)	Smart card	Cryptographic boundary	Plaintext	Manual	Electronic	AES-ECB (A2892)
Master key split Knowledge (Export)	Cryptographic boundary	Smart card	Plaintext	Automated	Direct	AES-ECB (A2892)
Split Knowledge (Import)	Smart card	Cryptographic boundary	Encrypted	Automated	Direct	AES-ECB (A2892)
Split Knowledge (Export)	Cryptographic boundary	Smart card	Encrypted	Automated	Direct	AES-ECB (A2892)
Wrapping key service	Cryptographic boundary	Outside of the boundary	Encrypted	Automated	Electronic	KeyWrap

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Unwrapping key service	Outside of the boundary	Cryptographic boundary	Encrypted	Automated	Electronic	KeyWrap

Table 17: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
MK_free	SSPs are encrypted with Master key and stored in eMMC.Zeroization of Mater key protects all other SSPs.	The Master key stored internally to the module in eMMC will overwrite with zeros within 1ms.	Triggered by tamper detection and response circuit
Key_free	Zeroization of Mater key can be done by API called.	The Master key stored internally to the module in eMMC will overwrite with zeros within 1ms. The time is not enough for the attacker to compromise the Master Key.	Call the function in application

Table 18: SSP Zeroization Methods

Since SSPs are encrypted with Master key and stored in eMMC in encrypted form, the Master key stored internally to the module in eMMC will overwrite with zeros within 1ms when the zeroization function is invoked. The time is not enough for the attacker to compromise the Master Key. Unprotected SSPs including session key, intermediate keygen values, entropy input string and DRBG seed, internal state: V value and Key are also zeroized when the corresponding crypto services is finished.

9.4 SSPs

The following two tables summarize the module's SSPs:

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Master key	Encrypt Application key	256 - 256	Symmetric key - CSP - CSP	HMAC DRBG (A2894) CKG		AES-ECB (A2892)
Client OAEP key	Wraps Session key	2048 - 112	Public key - PSP - PSP	Generated by host externally	KeyWrap	AES-KW (A2894) AES-KWP (A2894)
Audit log key	Verification of audit event log	256 - 256	Symmetric key - CSP - CSP	HMAC DRBG (A2894)		HMAC-SHA2-256 (A2892)
Clone key	Encrypt the whole HSM or a slot information	256 - 256	Symmetric key - CSP - CSP	HMAC DRBG (A2894)		AES-ECB (A2892)
Session key	Wrapped by Client OAEP key to encrypt data between host and HSM	128 - 128	Symmetric key - CSP - CSP	HMAC DRBG (A2894)		AES-ECB (A2892)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Application key RSA	Application purpose	2048, 3072, 4096 - 112, 128, >128	Private key - CSP / Public key - PSP - CSP	RSA (internal), the random value used in RSA is generated using SP800-90Arev1 DRBG		RSA SigGen (FIPS186-4) (A2894) RSA SigVer (FIPS186-4) (A2894)
Application key ECDSA	Application purpose	224, 256, 384, 521 - 112, 128, 192, 256	Public key - PSP/ Private key - CSP - CSP	ECDSA (internal), the random value used in ECDSA is generated using SP800-90Arev1 DRBG		ECDSA SigGen (FIPS186-4) (A2894) ECDSA SigVer (FIPS186-4) (A2894)
Application key AES	Application purpose	128, 192, 256 - 128, 192, 256	Symmetric key - CSP - CSP	HMAC DRBG (A2894)		AES-CBC (A2892) AES-ECB (A2892) AES-KW (A2894) AES-KWP (A2894)
Security officer's public key	Authenticate a role	2048 - 112	Public key - PSP - PSP	Generated by Smart card or host externally		RSA SigVer (FIPS186-4) (A2894)
User's public key	Authenticate a role	2048 - 112	Public key - PSP - PSP	Generated by Smart card or host externally		RSA SigVer (FIPS186-4) (A2894)
P11 slot SO password salt hash	Hash password	128 - 128	Password - CSP - CSP	Salt by HMAC (internal)		HMAC-SHA2-256 (A2892)
P11 slot User password salt hash	Hash password	128 - 128	Password - CSP - CSP	Salt by HMAC (internal)		HMAC-SHA2-256 (A2892)
Entropy Input String	Random number	256 - 256	Entropy Input - CSP - CSP	Obtained from entropy source		HMAC DRBG (A2894)
DRBG Seed, internal state: V value and Key	Random number	256 - 256	Internal state - CSP - CSP	HMAC DRBG (A2894)	CKG	HMAC DRBG (A2894)

Table 19: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Master key	Master key split Knowledge (Import) Master key split Knowledge (Export)	eMMC:Encrypted	While in use	MK_free	
Client OAEP key	Wrapping key service	Flash:Encrypted	While in use	Key_free	Session key:wraps

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	Unwrapping key service				
Audit log key	Wrapping key service Unwrapping key service	Flash:Encrypted	While in use	Key_free	
Clone key	Split Knowledge (Import) Split Knowledge (Export)	eMMC:Encrypted	While in use	Key_free	
Session key	Wrapping key service Unwrapping key service	SDRAM:Encrypted	While in use	Key_free	
Application key RSA	Wrapping key service Unwrapping key service	eMMC:Encrypted	While in use	Key_free	
Application key ECDSA	Wrapping key service Unwrapping key service	eMMC:Encrypted	While in use	Key_free	
Application key AES	Wrapping key service Unwrapping key service	eMMC:Encrypted	While in use	Key_free	
Security officer's public key	Wrapping key service Unwrapping key service	eMMC:Encrypted	While in use	Key_free	
User's public key	Wrapping key service Unwrapping key service	eMMC:Encrypted	While in use	Key_free	
P11 slot SO password salt hash	Wrapping key service Unwrapping key service	eMMC:Encrypted	While in use	Key_free	
P11 slot User password salt hash	Wrapping key service Unwrapping key service	eMMC:Encrypted	While in use	Key_free	
Entropy Input String		eMMC:Encrypted	From generation until DRBG seed is created	Key_free	DRBG Seed, internal state: V value and Key:Derives
DRBG Seed, internal state: V value and Key		eMMC:Encrypted	Until cipher handled is freed or module powered off	Key_free	DRBG Seed, internal state: V value and Key:Derived From

Table 20: SSP Table 2

9.5 Transitions

In compliance with NIST SP 800-131A Rev. 2, the module supports algorithms and key lengths that provide a minimum of 112 bits of security strength for applying cryptographic protection. Starting January 1, 2031, the minimum security strength for applying cryptographic protection will be 128 bits, and security strengths between 112 bits and 128 bits will be allowed for legacy use only to process information that is already protected.
Please see the latest revision of SP 800-131A and CMVP Programmatic Transitions page for transitions that may affect this module.

10 Self-Tests

The module performs pre-operational, conditional, and periodic self -tests to verify the integrity and correctness of the cryptographic functionality. If the self-tests all pass, a status message, “Self-tests OK” is displayed on the OLED. If any of self-tests fail, the module transitions to error state and must be rebooted.

10.1 Pre-Operational Self-Tests

The module implements the following pre-operational self-tests:

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
RSA SigVer (FIPS186-4) (A2894)	RSA 2048	Verification	SW/FW Integrity	Return code	Signature Verification

Table 21: Pre-Operational Self-Tests

10.2 Conditional Self-Tests

Conditional self-tests are running during operation of the module. The module performs the following conditional self-tests:

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-ECB (A2892)-Encrypt	AES-128	KAT	CAST	Return code	Encrypt	First use
AES-ECB (A2892)-Decrypt	AES-128	KAT	CAST	Return code	Decrypt	First use
ECDSA SigGen (FIPS186-4) (A2894)	P-256	KAT	CAST	Return code	Sign	First use
ECDSA SigVer (FIPS186-4) (A2894)	P-256	KAT	CAST	Return code	Verify	First use
ECDSA KeyGen (FIPS186-4) (A2894)	P-224, P-256, P-384, P-521	PCT	PCT	Return code	Sign and verify	Key pair generation

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-256 (A2892)	SHA2-256	KAT	CAST	Return code	Message Authentication	First use
HMAC DRBG (A2894)	SHA2-256	KAT	CAST	Return code	Instantiate, reseed and generate	First use
RSA SigGen (FIPS186-4) (A2894)	RSA-2048	KAT	CAST	Return code	Sign	First use
RSA SigVer (FIPS186-4) (A2894)	RSA-2048	KAT	CAST	Return code	Verify	First use
RSA KeyGen (FIPS186-4) (A2894)	2048, 3072, 4096	PCT	PCT	Return code	Sign and verify	Key pair generation
SHA2-256 (A2892)	Data length=1	KAT	CAST	Return code	Hash	First use
AES-KW (A2894)	AES-128	KAT	CAST	Return code	Key Wrap	First use

Table 22: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigVer (FIPS186-4) (A2894)	Verification	SW/FW Integrity	24 hours	Automatically

Table 23: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-ECB (A2892)-Encrypt	KAT	CAST	24 hours	Automatically
AES-ECB (A2892)-Decrypt	KAT	CAST	24 hours	Automatically
ECDSA SigGen (FIPS186-4) (A2894)	KAT	CAST	24 hours	Automatically
ECDSA SigVer (FIPS186-4) (A2894)	KAT	CAST	24 hours	Automatically
ECDSA KeyGen (FIPS186-4) (A2894)	PCT	PCT	24 hours	Automatically
HMAC-SHA2-256 (A2892)	KAT	CAST	24 hours	Automatically
HMAC DRBG (A2894)	KAT	CAST	24 hours	Automatically
RSA SigGen (FIPS186-4) (A2894)	KAT	CAST	24 hours	Automatically
RSA SigVer (FIPS186-4) (A2894)	KAT	CAST	24 hours	Automatically
RSA KeyGen (FIPS186-4) (A2894)	PCT	PCT	24 hours	Automatically
SHA2-256 (A2892)	KAT	CAST	24 hours	Automatically
AES-KW (A2894)	KAT	CAST	24 hours	Automatically

Table 24: Conditional Periodic Information

10.4 Error States

The module enters the error state If any of self-tests fail, the error state list is shown below.

Name	Description	Conditions	Recovery Method	Indicator
Pre-Operational Self-Tests Error	N/A	Pre-Operational tests failure	Restart the module	OLED displays error message
Conditional Self-Tests Error	N/A	Conditional tests failure	Restart the module	OLED displays error message and API returns error code

Name	Description	Conditions	Recovery Method	Indicator
Periodic Self-Tests Error	N/A	Periodic tests failure	Restart the module	OLED displays error message

Table 25: Error States

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

To initialize HiPKI SafGuard 2000, the administrator shall use a computer directly linked to the module and executes HiPKI SafGuard 2000 Management Tool to set up the module from initialization state to FIPS mode. See Administrator Guidance for more information about initialization process.

11.2 Administrator Guidance

See Administrator Guidance for more information.

11.3 Non-Administrator Guidance

See User Manual and Non-Administrator Guidance for more information.

11.4 Design and Rules

A Configuration Management and Security Delivery of Operation Document is provided by Chunghwa Telecom which offers System Administrators with the secure destruction procedures to ensure FIPS 140-3 Compliance of HiPKI SafGuard 2000.

12 Mitigation of Other Attacks

The module does not claim to mitigate any other attacks beyond the scope of FIPS 140-3 requirements.

Glossary and abbreviations

AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
CMVP	Cryptographic Module Validation Program
CO	Crypto Officer
CSP	Critical Security Parameter

DRBG	Deterministic Random Bits Generator
ECB	Electronic Codebook Book
ECDSA	Elliptic Curve Digital Signature Algorithm
ENT (NP)	Non-Physical Entropy Source
FIPS	Federal Information Processing Standard
HMAC	Keyed-Hash Message Authentication Code
HSM	Hardware Security Module
KTS	Key Transportation Schemes
RSA	Rivest-Shamir-Adleman Public Key Algorithm
SHA	Secure Hash Algorithm
SHS	Secure Hash
SSP	Sensitive Security Parameter