ST Engineering Urban Solutions Ltd. and i-Engine Pte Ltd.

ST Engineering & i-Engine SAM Applet on NXP P71D600

FIPS 140-3 Non-Proprietary Security Policy

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

Federal Information Processing Standards Publication 140-3 — Security Requirements for Cryptographic Modules specifies requirements for cryptographic modules to be deployed in a sensitive but unclassified (SBU) environment. The National Institute of Standards and Technology (NIST) and Canadian Centre for Cyber Security (CCCS) Cryptographic Module Validation Program (CMVP) run the FIPS 140-3 program. The National Voluntary Laboratory Accreditation Program (NVLAP) provides accreditation to independent testing labs performing FIPS 140-3 testing; the CMVP validates modules meeting FIPS 140-3 validation. Validated is the term given to a module that is documented and tested against the FIPS 140-3 criteria. More information is available on the CMVP website at: https://csrc.nist.gov/projects/cryptographic-module-validation-program.

This non-proprietary Cryptographic Module Security Policy for the ST Engineering & i-Engine SAM Applet on NXP P71D600 provides an overview of the product and a high-level description of how it meets the overall Security Level 3 requirements of FIPS 140-3.

The ST Engineering & i-Engine SAM Applet on NXP P71D600 may also be referred to as the “module” in this document.

1.2 Security Levels

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	4
8	Non-invasive security	3
9	Sensitive security parameter management	3
10	Self-tests	3
11	Life-cycle assurance	3
12	Mitigation of other attacks	3
	Overall Level	3

Table 1: Security Levels

1.3 Additional Information

The Module, validated to FIPS 140-3 overall Level 3, is a single chip module (P71D600 so known as “P71”) implementing the Global Platform operational environment, with a Card Manager (ISD/SSD) and an applet, SAM applet (hereafter referred to as the applet).

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design, and manufacturing. ST Engineering Urban Solutions Ltd. and i-Engine Pte Ltd. shall have no liability for any error or damages of any kind resulting from the use of this document.

Notices
This document may be freely reproduced and distributed in its entirety without modification.

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The module is designed to be used as a part of a larger system. It works as an auxiliary security device attached to a host controller. The physical form of the module is depicted in Figures 1 and 2 (to scale); the outline depicts the cryptographic boundary, representing the surface of the chip and the bond pads. The red outline in Figure 3 also depicts the cryptographic boundary. In production use, the module is delivered to either vendors or end user customers either on film frame carrier (FFC) or various packages such as PDM1.1, NXD6.2, MOB6/10 or HVQFN20 package. The package is outside the cryptographic boundary and thus excluded from the FIPS 140-3/ISO/IEC 19790 security testing.

The contactless ports of the module require connection to an antenna. The module relies on [ISO 7816] and [ISO 14443] card readers as input/output devices, or a [NXP I2C] connection to a host controller.

The Module is composed of a GlobalPlatform operational environment and a single Java Card applet – the SAM, running on the NXP P71D600 chip.

The Module has a limited operational environment under the FIPS 140-3 definitions. Firmware loading is limited to loading of the SAM applet on the NXP P71D600 chip by the vendor as part of the manufacturing process pre-shipment. New firmware versions within the scope of this validation must be validated through the CMVP. Any other firmware loaded into this Module is out of the scope of this validation and requires separate FIPS 140-3 validation.

The GlobalPlatform operational environment is identified by the following elements:

- The ROM ID
- The Platform ID (a data string that allows the identification of the P71D600 Card Manager component)
- The Patch ID
- Other information describing the content in ROM, NVM and loaded patches.

The SAM Applet is identified by the following elements:

- The Applet Name
- The Applet Version

The P71D600 GlobalPlatform operational environment component can be identified by using the IDENTIFY APDU command (Info service). This command returns the card identification data, which includes a Platform ID, a Patch ID and other information that allows the identification of

the content in ROM, NVM and loaded patches. The Platform ID is a data string that allows the identification of the P71D600 Card Manager component.

Module Type: Hardware

Module Embodiment: Single Chip

Cryptographic Boundary:

The JavaCard and Global Platform APIs are internal interfaces available to the applet. In the Approved mode only the SAM applet and Card Manager (ISD/SSD) services are available at the card edge (the interfaces that cross the cryptographic boundary).

Tested Operational Environment's Physical Perimeter (TOEPP):

The physical form of the module is depicted in Figures 1 and 2 (to scale); the outline depicts the cryptographic boundary, representing the surface of the chip and the bond pads. The red outline in Figure 3 also depicts the cryptographic boundary. The Tested Operational Environment's Physical Perimeter (TOEPP) is the entirety of the single-chip.

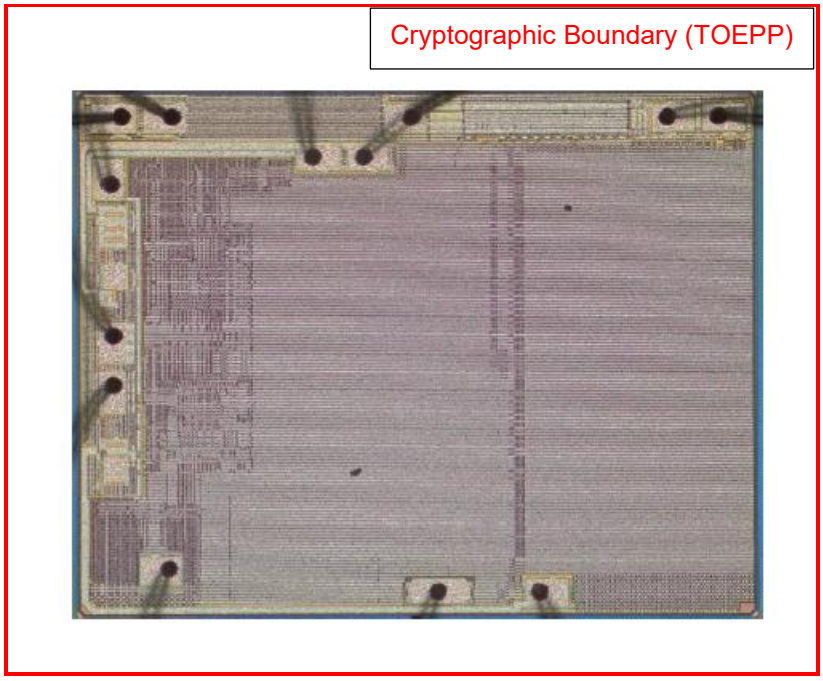

Figure 1: P71D600

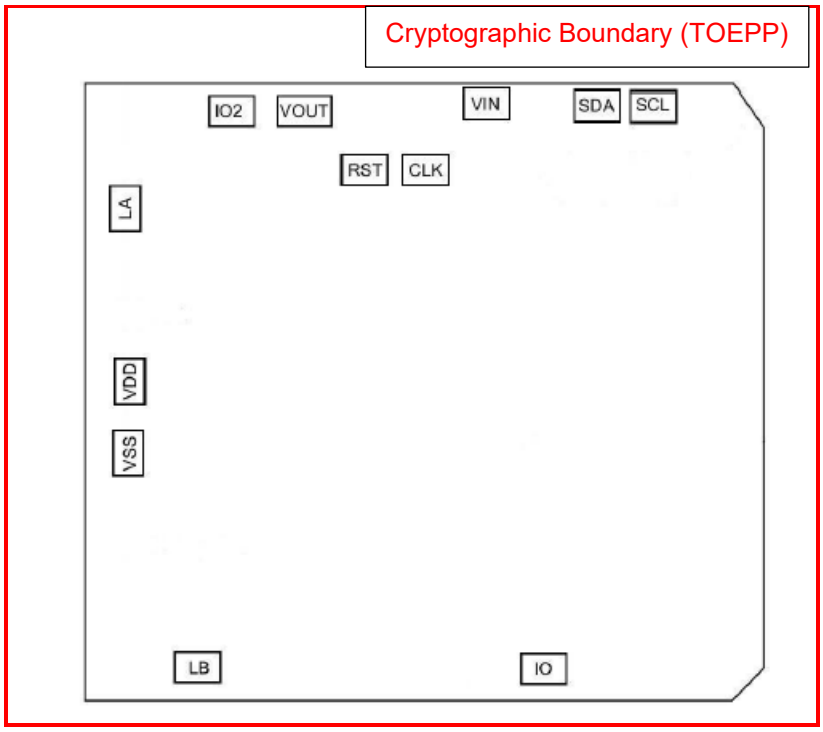

Figure 2: P71D600 Physical form (Schematic)

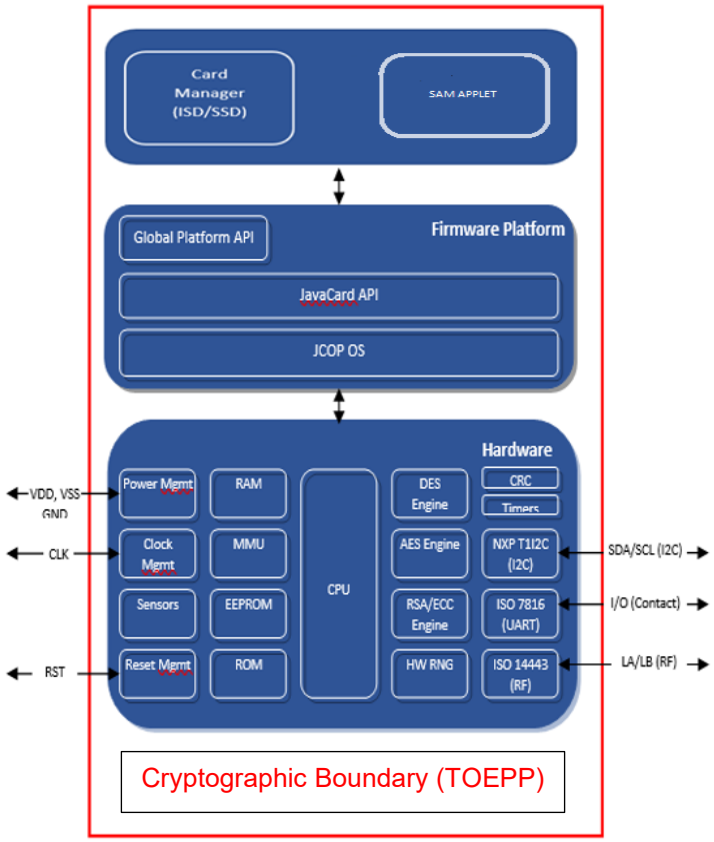

Figure 3: Module Block Diagram

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
P71D600	N7122 A1	Platform ID: J3R6000373181200 ROM ID: B3375FE9B5508BC4; Patch ID: 00000000 00000000; SAM v3.0.0.1.1	MRK3-SC 16/32-bit RISC CPU	The GlobalPlatform operational environment is identified with the Platform ID, the ROM ID, the Patch ID, and other information, describing the content in ROM, NVM and loaded patches. The Platform ID is a data string that allows the identification of the P71D600 Card

Table 2: Tested Module Identification – Hardware

2.3 Excluded Components

In production use, the Module is delivered to either vendors or end user customers either on film frame carrier (FFC) or various packages such as PDM1.1, NXD6.2, MOB6/10 or HVQFN20 package. The package is excluded from the FIPS140-3 security testing.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved mode	The module supports the Approved mode of operation after following the initialization instructions outlined in Section 11.1	Approved	To verify that the GlobalPlatform operational environment runs in the Approved mode of operation, use the IDENTIFY APDU. The DF28 file tag ‘05’ contains the status of the Approved mode compliancy, where ‘00’ identifies Approved mode not active and ‘01’ - Approved mode active. The Lifecycle service (GET STATUS APDU/command) can be used to verify that the SAM applet is in the Approved mode of operation (it is in this mode by default post initialization): Command: 00 CA DF 66 00; Output Data: 01; Status: 90 00

Table 3: Modes List and Description

Mode Change Instructions and Status:

The module only supports an Approved mode of operation by default.

The P71D600 GlobalPlatform operational environment component can be identified by using the IDENTIFY APDU command (Info service). This command returns the card identification data, which includes a Platform ID, a Patch ID and other information that allows the identification of the content in ROM, NVM and loaded patches. The Platform ID is a data string that allows the identification of the P71D600 Card Manager component.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A2713	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A2713	Key Length - 128, 192, 256	SP 800-38C
AES-CMAC	A2713	Direction - Generation, Verification Key Length - 128, 192, 256	SP 800-38B
AES-CTR	A2713	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A2713	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
Counter DRBG	A2713	Prediction Resistance - No, Yes Mode - AES-256 Derivation Function Enabled - Yes	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-5)	A6534	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A6534	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A6534	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A2713	Key Length - Key Length: 112-2048 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A2713	Domain Parameter Generation Methods - P-256 Scheme - onePassDh - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA HKDF Sp800-56Cr1	A2713	Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-4096 Increment 8 HMAC Algorithm - SHA-1, SHA2-256, SHA2-384, SHA2-512	SP 800-56C Rev. 2
KDF SP800-108	A2713	KDF Mode - Counter, Feedback Supported Lengths - Supported Lengths: 112-4096 Increment 8, Supported Lengths: 128	SP 800-108 Rev. 1
RSA Decryption Primitive Sp800-56Br2 (CVL)	A5911	Modulo - 2048, 3072, 4096	SP 800-56B Rev. 2
RSA KeyGen (FIPS186-5)	A6535	Key Generation Mode - probable Modulo - 2048, 3072, 4096 Primality Tests - 2pow100 Private Key Format - standard	FIPS 186-5
RSA KeyGen (FIPS186-5)	A6536	Key Generation Mode - probable Modulo - 2048, 3072, 4096	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
		Primality Tests - 2pow100 Private Key Format - crt	
RSA SigGen (FIPS186-5)	A6535	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-5)	A6535	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
SHA-1	A2713	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A2713	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A2713	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 4: Approved Algorithms

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG SP800-133r2 Section 4	Key Type:Symmetric and Asymmetric	N/A	NIST SP800-133r2 Section 4: Symmetric keys and seeds for asymmetric key generation are generated using methods described in Section 4 of SP800-133r2 (example 1). The module supports the following per NIST SP 800-133r2: 1. Section 5.1: Key Pairs for Digital Signature Schemes 2. Section 6.2.1: Symmetric Keys Generated Using Key-Agreement Schemes 3. Section 6.2.2: Symmetric Keys Derived from a Pre-existing Key 4. Section 6.4: Distributing the Generated Symmetric Key
CKG SP 800-133r2 Section 6.3	Key Type :Symmetric	N/A	NIST SP 800-133r2 Section 6.3 Symmetric Keys Produced by Combining (Multiple) Keys and Other Data (approach/method 2. Exclusive-Oring one or more symmetric keys and possibly one or more other items of data)

Table 5: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

The module does not implement any Non-Approved, Allowed Algorithms.

Non-Approved, Allowed Algorithms with No Security Claimed:

Name	Caveat	Use and Function
RSA Encryption using PKCS1.5 padding	no security claimed	Used to encrypt and protect keys; Per IG 2.4.A example 1.

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

Non-Approved, Not Allowed Algorithms:

The module does not implement any Non-Approved, Not Allowed Algorithms.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Counter DRBG	DRBG	Random bit generation		Counter DRBG: (A2713)
KAS-1	KAS-Full	NIST SP 800-56Arev3 KAS-ECC-SCC per IG D.F Scenario 2 path (2); Used in the context of ECDH Key Agreement	IG :IG D.F Scenario 2 path (2), split Key confirmation:no Key derivation :KDA (separately tested) Caveat:Key establishment methodology provides 128 bits of security strength	KAS-ECC-SSC Sp800-56Ar3: (A2713) CKG SP800-133r2 Section 4 : () Key Type: Symmetric and Asymmetric KDA HKDF Sp800-56Cr1: (A2713)
Perform Self-Tests	BC-Auth BC-UnAuth DigSig-SigGen DigSig-SigVer DRBG KAS-56CKDF KAS-SSC KBKDF MAC SHA	Perform self-tests on demand		AES-CBC: (A2713) AES-CMAC: (A2713) Counter DRBG: (A2713) ECDSA SigGen (FIPS186-5): (A6534) ECDSA SigVer (FIPS186-5): (A6534) KAS-ECC-SSC Sp800-56Ar3: (A2713) KDA HKDF Sp800-56Cr1:

Name	Type	Description	Properties	Algorithms
				(A2713) KDF SP800-108: (A2713) RSA SigGen (FIPS186-5): (A6535) RSA SigVer (FIPS186-5): (A6535) SHA-1: (A2713) SHA2-256: (A2713) SHA2-512: (A2713)
Asymmetric Key Generation	AsymKeyPair-KeyGen	Aymmetric keypair generation		RSA KeyGen (FIPS186-5): (A6535, A6536) ECDSA KeyGen (FIPS186-5): (A6534) CKG SP800-133r2 Section 4 : () Key Type: Symmetric and Asymmetric
SCP03/secure channel	KBKDF	Sessions keys are used with AES-CBC and AES-CMAC to provide an end-to-end confidential and authenticated protected channel (Approved KTS) between the external entity (User) and the module. KDF SP800-108 Counter is used for deriving the secure channel keys		KDF SP800-108: (A2713) CKG SP800-133r2 Section 4 : () Key Type: Symmetric and Asymmetric HMAC-SHA-1: (A2713)
EC Crypto Operations	DigSig-SigGen DigSig-SigVer	ECDSA Signature		ECDSA SigGen (FIPS186-5): (A6534)

Name	Type	Description	Properties	Algorithms
		Generation and Verification		ECDSA SigVer (FIPS186-5): (A6534)
RSA Crypto Operations	DigSig-SigGen DigSig-SigVer	RSA Signature Generation and Verification, RSA Decryption		RSA SigGen (FIPS186-5): (A6535) RSA SigVer (FIPS186-5): (A6535) RSA Decryption Primitive Sp800-56Br2: (A5911)
Symmetric Cipher Crypto Operations	BC-Auth BC-UnAuth	Encryption and Decryption		AES-CBC: (A2713) AES-CMAC: (A2713) AES-CCM: (A2713) AES-CTR: (A2713) AES-ECB: (A2713)
KTS-1	KTS-Unwrap KTS-Wrap	SP 800-38F KTS (key wrapping and unwrapping) per IG D.G	Standard:SP 800-38F IG D.G:approved combination method from IG D.G Key confirmation:no Caveat:Key establishment methodology provides between 128 and 256 bits of security strength	AES-CBC: (A2713) AES-CMAC: (A2713) CKG SP800-133r2 Section 4 : () Key Type: Symmetric and Asymmetric
DAP	DigSig-SigVer SHA	Firmware Load Test		ECDSA SigVer (FIPS186-5): (A6534)
Entropy Source	ENT-ESV	Entropy Input provided by the entropy source within the module to seed the approved NIST SP 800-		

Name	Type	Description	Properties	Algorithms
		90Ar1 CTR_DRBG		
OS-MKEK Generation	CKG	OS-SKEK permutation (xor between OS-SKEK and a constant value)		CKG SP 800-133r2 Section 6.3: () Key Type : Symmetric

Table 7: Security Function Implementations

2.7 Algorithm Specific Information

IG 2.4.B
RSADP (standard)
Usage Restriction: The RSA Decryption Primitive (CVL) shall only be used within the context of a SP 800-56Brev2 KTS.

IG D.H
Symmetric keys and seeds used for generating the asymmetric keys are generated using methods described in Section 4, Section 5.1, Section 6.2.1, Section 6.2.2, Section 6.3 and Section 6.4 of SP 800-133r2.

IG D.L
In the case of the CTR_DRBG, the test report shall indicate if a derivation function is used during the instantiation and reseeding: The CTR_DRBG implementation of the module does use a derivation function (CAVP Cert. #A2713).

In accordance with the Resolution in the IG, the V and Key values for the CTR_DRBG have been defined as CSPs as can be verified from the Section 9.4 SSP Table 1 Table 20.

IG D.M
Specific requirements for generating symmetric keys using SP 800-108 are found in Sec. 6.4 of SP 800-133rev1, “Symmetric Keys Derived from a Preshared Key.” SP 800-108 KDFs may not be used to generate asymmetric keys:

As can be verified from the Section 9.4 SSP Table 2 Table 21, the module only uses KBKDF to derive symmetric keys.

IG D.F:

KAS-1: Key Agreement i.e. KAS-ECC-SSC per NIST SP 800-56Arev3 combined with an HKDF per NIST SP 800-56Cr1 (KDA) used in the context of ECDH Key Agreement service (Scenario 2 path (2) per IG D.F – KAS-ECC-SSC and KDA self-tested separately):

KAS (KAS-ECC-SSC CAVP Cert. #A2713 and KDA HKDF NIST SP 800-56Cr1 CAVP Cert. #A2713; P-256 curve providing 128 bits of encryption strength)

IG D.G:

KTS-1: Key wrapping (Using the approved AES modes CMAC and CBC) used to provide an end-to-end confidential and authenticated protected channel between the external entity (User) and the module (i.e. used in the context of the Secure Channel service). This is per Scenario 2 in IG D.G Approved methods for key transport i.e., a “combination” method: use any approved symmetric encryption mode, such as AES ECB, AES CBC, Triple-DES ECB, etc. together with an approved authentication method (for example, HMAC or AES CMAC, or KMAC):

KTS (AES-CMAC CAVP Cert. #A2713 and AES-CBC CAVP Cert. #A2713; 128, 192, and 256-bit keys providing 128, 192, or 256 bits of encryption strength)

IG 9.6.A
An AES or a Triple-DES encryption using any approved mode of AES or the Triple-DES as defined in SP 800-140C CMVP Approved Security Functions: SSPs are stored encrypted using AES-CBC.

Additional Comment #1: The approved algorithm implementations used to protect stored SSPs shall be tested by the CAVP (or vendor affirmed if allowed by an IG): The AES-CBC has been tested per CAVP Cert. #A2713.

IG C.E:
The module generates RSA signature keys using an approved key generation procedure per RSA KeyGen validated for conformance to FIPS 186-5 Certs. #A6535 and #A6536.

IG C.F:
The RSA KeyGen, SigGen and SigVer implementations have been tested for all implemented RSA modulus lengths (moduli 2048, 3072 and 4096 bits). No untested moduli apply.

2.8 RBG and Entropy

Cert Number	Vendor Name
E148	NXP Semiconductors

Table 8: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
JCOP 4.5 on P71D600	Physical	JCOP 4.5 on P71D600	8 bits	7.30359 bits	N/A

Table 9: Entropy Sources

The Public Use Document (PUD) for the entropy source validation (ESV Cert. #E148) can be found at: https://csrc.nist.gov/CSRC/media/projects/cryptographic-module-validation-program/documents/entropy/E148_PublicUse.pdf. The min-entropy per byte provided by the entropy source is 7.30359 bits/byte and the DRBG is seeded with 384 bits which is sufficient for generation of keys with a security strength of up to 256 bits. Given that 256 bits is the maximum

key length supported by the module, the entropy provided is deemed sufficient and a caveat per IG 9.3.A thus does not apply.

2.9 Key Generation

The module uses an approved NIST SP 800-90Ar1 DRBG for the generation of keys/SSPs.

2.10 Key Establishment

The module supports approved key establishment methods as specified in the Security Functions Implementations tables (KAS-1 and KTS-1).

2.11 Industry Protocols

The module does not support any industry protocols.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
VSS, VDD	Power	These interfaces are used to supply power to the module in contact mode; The module starts when interface is powered
VIN, VOUT	Power	These interfaces are used to supply power to the module in contact, contactless and I2C mode
RST_N	Control Input	If a signal is sent on this interface on contact mode, the module will reboot (active low)
CLK	Control Input	The interface is used by an external device (ex: smartcard reader) to provide a clock signal to the IC in contact mode; The IC will derive its own clock from this signal
IO1	Data Input Data Output Control Input Status Output	The interface is used to communicate with an external entity (ex: SmartCard reader) in contact mode; It also functions as I2C master SDA in I2C mode
IO2	Data Input Data Output Control Input Status Output	The interface is used to communicate with an external entity (ex: SmartCard reader) in contact mode; It also functions as I2C master SCL in I2C mode or as SPI interface
LA, LB	Data Input Data Output Control Input	The interface is used to communicate with an external entity (ex: smartcard reader) in contactless mode; This interface is also used to set the internal clock and to supply power to the module

Physical Port	Logical Interface(s)	Data That Passes
	Status Output Power	
SDA	Data Input Data Output Control Input Status Output	The interface is used to communicate with an external entity such as a host controller
SCL	Control Input	The interface is used by an external device (ex: host controller) to provide a clock signal to the I2C HW

Table 10: Ports and Interfaces

The module does not support control output.

4 Roles, Services, and Authentication

4.1 Authentication Methods

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
Platform Authentication (Secure Channel Protocol 03 Authentication Method)	The Secure Channel Protocol authentication method is provided by the Secure Channel service; The SD-KENC and SD-KMAC keys are used to derive the SD-SENC, SD-SMAC, and SD-	The external entity participating in the mutual authentication sends a 64-bit challenge to the Secure Element; The Secure Element generates its own challenge and computes a 64-bit cryptogra	1/(2^128) = 2.9E-39 (MAC||cryptogram) using a 128-bit block for authentication; This authentication method includes a counter of failed authentication called “velocity checking” by GlobalPlatform; The counter is decremented prior to any attempt to authenticate and is only reset to its threshold (maximum value) upon successful authentication	The module enforces a maximum of 60 failed Global Platform SCP03 authentication attempts before permanently blocking the card; The probability that a random attempt will succeed over a one-minute interval is (with the assumption here that one attempt is possible per second): 60/(2^128) = 1.7E-37 (MAC||cryptogra

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
	RMAC session keys; These sessions keys are used with AES-CBC and AES-CMAC to provide an end-to-end confidential and authenticated protected channel (Approved KTS) between the external entity (User) and the module	m with SD-SMAC key and both challenges; The Secure Element cryptogram and challenge are sent to the external entity which checks the Secure Element cryptogram and creates its own 64-bit cryptogram with both challenges; A 64-bit message authentication code (MAC) is also computed on the command containing the external entity cryptogram with AES-CMAC and SD-SMAC key; The		m), using a 128-bit block for authentication

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
		MAC is concatenated to the command, and the command is sent to the Secure Element; The Secure Element checks the message authentication code and compares the received cryptogram to the calculated cryptogram; If all of this succeeds, the two participants are mutually authenticated (the external entity is authenticated to the Module)		
Owner PIN	8 bytes PIN used for authentication	-	The probability that a random-access attempt will succeed when using a PIN of size 8 bytes depends on the total number of possible PIN combinations. 1. PIN Size: 8 bytes means there are 8 * 8 = 64 bits (since 1 byte = 8 bits). 2. Possible Combinations: The	Probability of Success: The probability of success for any single random-access attempt with a 8-byte PIN, is ≈ 5.4 x 10^-19

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
			number of possible PIN combinations is 2^64, as each bit can be either 0 or 1. Therefore, the total number of combinations is: 2^64=18,446,744,073,709,551,616 If a random-access attempt means choosing one of these combinations, and assuming all combinations are equally likely, the probability of a successful attempt is: Probability of success=1/2^64 This is an extremely small probability, roughly: 1/118,446,744,073,709,551,616≈5.4×10^−19 So, the probability of a successful random-access attempt with an 8-byte PIN is extraordinarily low	Probability of Failure for One Attempt: The probability of a failure on any single random access attempt is: Pfailure=1−Psuccess≈1 (Since the success probability is so small, the failure probability is practically 1.) Probability of Failure for All Attempts in One Minute: If 60N attempts are made in one minute, the probability of failing on all attempts is: Pfailure (all)=(Pfailure)^60N ≈1^(60N) =1 (Again, this approximation assumes the success probability is negligible.) Probability of At Least One Success in One Minute: The probability of having at least one success is the complement of the probability of failing on all attempts: Psuccess (at least one)=1−Pfailure (all)=1−(Pfailure)^

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
				(60N) Since Pfailure is extremely close to 1, we can estimate this probability as: Psuccess (at least one)≈ 60N×Psuccess So, the probability of success within one minute, given N attempts per second, is: Psuccess (at least one)≈ 60N×6.8×10^−49 Assuming 1 million attempts can be made per second = 10^6, the number of attemper per minute is: 60x10^6 = 60, 000, 000 attempts The probability of at least one success in one minute is: Psuccess (at least one)≈ 60, 000, 000 attempts x 5.4×10^−19 ~ 3.24 x 10 ^-11
Marriage code	20 bytes PIN used for authentication	-	The probability that a random-access attempt will succeed when using a PIN of size 8 bytes depends on the total number of possible PIN combinations. 1. PIN Size: 20 bytes means there are 20 * 8 = 160 bits (since 1 byte = 8 bits). 2. Possible Combinations: The number of possible PIN combinations is 2^160, as each bit can be either 0 or 1. Therefore, the total number of combinations is: 2^160=	Probability of Success: The probability of success for any single random-access attempt with a 20-byte PIN, is ≈6.8×10^−49 Probability of Failure for One Attempt: The probability of a

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
			1,461,501,637,330,901,217,467,999,448 If a random-access attempt means choosing one of these combinations, and assuming all combinations are equally likely, the probability of a successful attempt is: Probability of success=1/2^160 This is an extremely small probability, roughly: 1/ 1,461,501,637,330,901,217,467,999,448 ≈ 6.8×10^−49 So, the probability of a successful random-access attempt with an 20-byte PIN is extraordinarily low	failure on any single random access attempt is: Pfailure=1−Psuccess≈1 (Since the success probability is so small, the failure probability is practically 1.) Probability of Failure for All Attempts in One Minute: If 60N attempts are made in one minute, the probability of failing on all attempts is: Pfailure (all)=(Pfailure)^60N ≈1^(60N) =1 (Again, this approximation assumes the success probability is negligible.) Probability of At Least One Success in One Minute: The probability of having at least one success is the complement of the probability of failing on all attempts: Psuccess (at least one)=1−Pfailure (all)=1−(Pfailure)^(60N) Since Pfailure is extremely close to 1, we can

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
				estimate this probability as: Psuccess (at least one)≈ 60N×Psuccess So, the probability of success within one minute, given N attempts per second, is: Psuccess (at least one)≈ 60N×6.8×10^−49 Assuming 1 million attempts can be made per second = 10^6, the number of attemper per minute is: 60x10^6 = 60, 000, 000 attempts The probability of at least one success in one minute is: Psuccess (at least one)≈ 60, 000, 000 attempts x 6.8×10^−49 ~ 4.08 x 10^-41
Passive Authentication	ECDSA P-384 SHA2-384	EC Crypto Operations	The private key is a random number in the range { 0 , 1 , 2 , … , n − 1 } {0,1,2,…,n−1}, where n n is the order of the elliptic curve (approximately 2^384 for P-384); Since the key space is approximately 2^384, the probability of successfully forging a valid signature (i.e., a random-access attempt succeeding) is around: 1/(2^384) = ~ 3.5×10^(-116)	The probability of a single attempt succeeding is 3.5×10^(-116); The probability of at least one successful attempt per second can be approximated as: 10^12 x 1/ (2^384) = ~3.5 x 10 ^(-104); Since there are 60 seconds in a minute, multiply the probability per

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
				second by 60 to get the probability per minute: 60×3.5x10^(-104) = 2.1 x 10^(-102)
Active Authentication	AES 256-bit key used (mode: ECB) to send random nonces to the SSID (an external applet). The SAM encrypts the nonce using this key, if the SSID can decrypt it, mutual authentication succeeds	Symmetric Cipher Crypto Operations	To successfully decrypt the ciphertext and authenticate, the attacker would need to know the AES-256 key used for encryption; AES-256 bit key has a space of 2^256, so the probability of randomly guessing the correct key i.e. random access is = 1/(2^256) = ~8.63×10^(-78)	The probability of a successful random access is: 8.63×10^(-78); The probability of success per minute is: 60×8.63x10^(-78) = 5.18 x 10^(-76)

Table 11: Authentication Methods

The module employs (and enforces i.e. the method is required) identity-based authentication mechanisms.

Authentication of each operator and their access to roles and services is as described below, independent of logical channel usage (identity-based authentication methods implemented).

- The SCP03 authentication method is the only method available to the operator on first use. This permits the Cryptographic Officer (CO) to configure the module.
- Only one operator at a time is permitted on a channel.
- Applet de-selection (including Card Manager), card reset, or power down terminates the current authentication. Re-authentication is required after any of these events for access to authenticated services.
- CO authentication method does not exchange plaintext CSPs.
- User authentication data is encrypted and authenticated during entry with GlobalPlatform SCP03, is stored encrypted with OS-MKEK and is only accessible by authenticated services.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Cryptographic Officer	Identity	Cryptographic Officer	Platform Authentication (Secure Channel Protocol 03 Authentication Method) Owner PIN Marriage code Passive Authentication Active Authentication
User	Identity	User	Platform Authentication (Secure Channel Protocol 03 Authentication Method) Owner PIN Marriage code Passive Authentication Active Authentication

Table 12: Roles

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Manage Content (ISD Services)	Load keys and data; APDU(s) used: DELETE, LOAD, INSTALL, MANAGE CHANNEL	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY APDU can be used where the DF28 file tag ‘05’ contains the status of the Approved mode compliancy (‘01’	Command parameters (data objects, SSPs)	Status Word (Response APDU 9000)	KTS-1 DAP	Cryptographic Officer - OS-SKEK: W,E,Z - SD-KENC: W,E,Z - SD-KMAC: W,E,Z - SD-KDEK: W,E,Z - DAP-DAPK: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		signifies that Approved mode is active))				
Lifecycle (Show status and Perform zeroisation) (SSD Services)	Get or modify the card or applet life cycle status; APDU(s) used: SET STATUS, GET STATUS, TERMINATE (Zeroise)	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY APDU can be used where the DF28 file tag '05' contains the status of the Approved mode compliancy ('01' signifies that Approved mode is active))	Target status	Status Word (Response APDU 9000)	KTS-1 DAP	Cryptographic Officer - OS-DRBG-EI: E,Z - OS- DRBG-KEY: E,Z - OS-DRBG-V: E,Z - OS-SKEK: E,Z - OS-MKEK: E,Z - SD-KENC: E,Z - SD-KMAC: E,Z - SD-KDEK: E,Z - SD-SENC: E,Z - SD-SMAC: E,Z - SD-RMAC: E,Z - DAP-DAPK: E,Z - RSA-PRIV-KEY: E,Z - EC-PRIV-KEY: E,Z - EC-KAS-Z: E,Z - RSA-PUB-KEY: E,Z - EC-PUB-KEY: E,Z - EC-KAS-PUB: E,Z
Manage Content (SSD Services)	Load keys and data; APDU(s) used: PUT	Status Word (Response APDU 9000);	Command parameters (data	Status Word (Response APDU 9000)	KTS-1 DAP	Cryptographic Officer - OS-SKEK: W,E,Z - SD-KENC:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	KEY, STORE DATA	Approved mode indicator (IDENTIFY APDU can be used where the DF28 file tag '05' contains the status of the Approved mode compliancy ('01' signifies that Approved mode is active))	objects, SSPs)			W,E,Z - SD-KMAC: W,E,Z - SD-KDEK: W,E,Z - DAP-DAPK: W,E
Privileged Info (Show module's versioning information) (SSD Services)	Read Module data (privileged data objects, but no CSPs); APDU(s) used: GET DATA	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY APDU can be used where the DF28 file tag '05' contains the status of the Approved mode	Command: IDENTIFY APDU	Requested information; Status Word (Response APDU 9000)	KTS-1	Cryptographic Officer - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		compliancy (‘01’ signifies that Approved mode is active))				
Secure Channel (SSD Services)	Establish and use a secure communication channel; APDU(s) used: INITIALIZE, UPDATE, EXTERNAL AUTHENTICATE	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY APDU can be used where the DF28 file tag ‘05’ contains the status of the Approved mode compliancy (‘01’ signifies that Approved mode is active))	Command parameters (data objects, SSPs)	Status Word (Response APDU 9000)	Counter DRBG SCP03/secure channel KTS-1 Entropy Source OS-MKEK Generation	Cryptographic Officer - OS-DRBG-EI: G,E,Z - OS- DRBG-KEY: G,E,Z - OS-DRBG-V: G,E,Z - OS-MKEK: G,E,Z - SD-KENC: G,E,Z - SD-KMAC: G,E,Z - SD-SENC: G,E,Z - SD-SMAC: G,E,Z - SD-RMAC: G,E,Z - OS-DRBG-SEED: G,E,Z
Card Reset	Power cycle or reset the module; APDU(s) used: N/A	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY	Power cycle or reset the module	Status Word (Response APDU 9000)	KTS-1	Cryptographic Officer - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		APDU can be used where the DF28 file tag ‘05’ contains the status of the Approved mode compliancy (‘01’ signifies that Approved mode is active))				User - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E Unauthenticated
Context	Select an applet or manage logical channels; APDU(s) used: SELECT, MANAGE CHANNEL	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY APDU can be used where the DF28 file tag ‘05’ contains the status of the Approved mode compliancy (‘01’ signifies that Approve	Command parameters (data objects, SSPs)	Status Word (Response APDU 9000)	KTS-1	Cryptographic Officer - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E User - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E Unauthenticated

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		d mode is active))				
Info (Show status and Perform self-tests)	Read unprivileged data objects, e.g., module configuration or status information (Show Status). This service includes the Pre-operational Self-Test on-demand; APDU(s) used: GET DATA	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY APDU can be used where the DF28 file tag '05' contains the status of the Approved mode compliancy ('01' signifies that Approved mode is active))	Command parameters (data objects, SSPs)	Status Word (Response APDU 9000)	Perform Self-Tests	Cryptographic Officer - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E User - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E Unauthenticated
Get Applet Version	Retrieve firmware version of the applet	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY APDU can be used where the DF28	Command: Read Applet Version APDU (00 CA DF 65 00)	Status Word (Response APDU 9000)	None	Cryptographic Officer - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E - Marriage Code: W,E - Owner PIN: W,E - Passive

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		file tag ‘05’ contains the status of the Approved mode compliancy (‘01’ signifies that Approved mode is active))				Authentication Key: E - Active Authentication: E User - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E - Marriage Code: W,E - Owner PIN: W,E - Passive Authentication Key: E - Active Authentication: E
Create data group	Inject keys (AES, ECDSA, RSA)	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY APDU can be used where the DF28 file tag ‘05’ contains the status of the Approved mode compliancy (‘01’	Command parameters (data objects, SSPs)	Status Word (Response APDU 9000)	KTS-1	Cryptographic Officer - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E - RSA-PRIV-KEY: W - RSA-PUB-KEY: W - EC-PUB-KEY: W - EC-PRIV-KEY: W - Application AES Communication : W - Marriage Code: W,E - Owner PIN:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		signifies that Approved mode is active))				W,E - Passive Authentication Key: E - Active Authentication: E User - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E - RSA-PRIV-KEY: W - RSA-PUB-KEY: W - EC-PRIV-KEY: W - EC-PUB-KEY: W - Application AES Communication : W - Marriage Code: W,E - Owner PIN: W,E - Passive Authentication Key: E - Active Authentication: E
Perform Cryptography	Perform AES Encrypt, Decrypt, RSA Sign, Verify, Decrypt, ECDSA Verify	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY APDU	Command parameters (data objects, SSPs)	Status Word (Response APDU 9000)	Asymmetric Key Generation EC Crypto Operations RSA Crypto Operations Symmetric Cipher Crypto Operations	Cryptographic Officer - RSA-PRIV-KEY: G,E - RSA-PUB-KEY: G,E - EC-PRIV-KEY: G,E - EC-PUB-KEY: G,E - Application

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		can be used where the DF28 file tag ‘05’ contains the status of the Approved mode compliancy (‘01’ signifies that Approved mode is active))				AES Communication : E - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E - Marriage Code: W,E - Owner PIN: W,E - Passive Authentication Key: E - Active Authentication: E User - RSA-PRIV-KEY: G,E - EC-PRIV-KEY: G,E - RSA-PUB-KEY: G,E - EC-PUB-KEY: G,E - Application AES Communication : E - OS-SKEK: E - SD-KENC: E - SD-KMAC: E - SD-KDEK: E - Marriage Code: W,E - Owner PIN: W,E - Passive Authentication Key: E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- Active Authentication: E
ECDH Key Agreement	Key Agreement	Status Word (Response APDU 9000); Approved mode indicator (IDENTIFY APDU can be used where the DF28 file tag '05' contains the status of the Approved mode compliancy ('01' signifies that Approved mode is active))	Key identifier, Mechanism identifier, Host ECC public key	Status Word (Response APDU 9000)	KAS-1	User - EC-KAS-Z: G,E,Z - EC-KAS-PUB: W,E,Z - EC-PRIV-KEY: G,E,Z

Table 13: Approved Services

The modes of access shown in the table above are defined as:
• G = Generate: The service generates or derives the CSP/Public Key.
• W = Write: The service inputs the CSP/Public Key.
• E = Execute: The Module executes using the CSP/Public Key.
• R = Read: The service outputs the CSP/Public Key. CSP are always protected with the approved KTS.
• Z = Zeroise: The Module zeroises the CSP/Public Key after usage. A zeroised CSP is not retrievable or reusable.

4.4 Non-Approved Services

The module does not support any Non-Approved Services.

4.5 External Software/Firmware Loaded

The module contains a limited operational environment. The only possible loading is that of the SAM applet onto the NXP P71D600 chip/hardware platform and this is performed by the vendor at manufacture time i.e. pre-shipment. The module implements a firmware load test in support of this loading though not required given as the loading occurs at manufacture time per the Security Policy Section 10.2, Conditional Self-Tests Table (using ECDSA P-256 SHA2-256).

5 Software/Firmware Security

5.1 Integrity Techniques

The cryptographic module is considered a hardware module with firmware components. An error detection code (32-bit CRC performed over all code located in Flash) is applied to all firmware components within the module. If the integrity test fails, the module enters the hard error (MUTE) state. The module is a single-chip and the executable code of the module (firmware), is Register Transfer Language (RTL). All data and control inputs, and data, control and status outputs of the cryptographic module and services are directed through the module's defined interfaces.

5.2 Initiate on Demand

An operator of the module can perform the integrity test on demand with the GET DATA APDU command.

5.3 Additional Information

ROM endurance has been proven to be more than 10 years after the date of manufacture. Therefore, per FIPS 140-3 IG 5.A, no pre-operational ROM integrity self-test has been implemented. The module's end-of-life procedures must be applied prior to the degradation of the ROM by setting the module to the TERMINATE state.

All data and control inputs, and data and status outputs of the cryptographic module and services are directed through the module's defined interfaces.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Limited

How Requirements are Satisfied:

The module claims to meet Physical Security Level 4 and thus the requirements per this section do not apply.

7 Physical Security

7.1 Mechanisms and Actions Required

Mechanism	Inspection Frequency	Inspection Guidance
Hard, tamper evident coating	N/A	N/A

Table 14: Mechanisms and Actions Required

The module is a single-chip implementation that meets commercial-grade specifications for power, temperature, reliability, and shock/vibrations. The module uses standard passivation techniques. The module includes Environmental Failure Protection features such as temperature and voltage sensors. Fault Induction mitigation techniques are light sensors and spike sensors on the supply voltage lines.

Identification of internal features such as sensitive components or interconnections is impeded by a fine mesh of metal shield lines that reside at the outermost layers of the chip.

Delivery forms of the module are QFN package, contactless chip card module, or sawn wafer. Therefore, the module does not rely on any physical security based on a package.

7.2 Fault Induction Mitigation

As specified in Section 12 of this document, the module implements fault induction mitigations such as light sensors, voltage glitch sensors and an active shield, and detection of illegal address or instruction.

7.3 EFP/EFT Information

Temp/Voltage Type	Temperature or Voltage	EFP or EFT	Result
LowTemperature	-40C	EFP	Shutdown
HighTemperature	+105C	EFP	Shutdown
LowVoltage	1.62V	EFP	Shutdown
HighVoltage	6.0V	EFP	Shutdown

Table 15: EFP/EFT Information

7.4 Hardness Testing Temperature Ranges

Temperature Type	Temperature
LowTemperature	-45C
HighTemperature	+125C

Table 16: Hardness Testing Temperatures

8 Non-Invasive Security

8.1 Mitigation Techniques

In accordance with FIPS 140-3 IG 12.A Additional Comment 1, the non-invasive security mitigations have been specified in Section 12 of this document.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
Temporarily stored in RAM	plaintext (does not persist beyond a power cycle); object identifier to entity association	Dynamic
Stored in NVM	plaintext; object identifier to entity association	Static
Stored in NVM - Encrypted	encrypted with Approved AES CBC with OSMKEK; key version to entity association	Static
Plaintext: Persistent	plaintext (persists beyond a power cycle); identity-based authentication data	Static

Table 17: Storage Areas

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Imported in secure channel specified by GP-Amd-I	External endpoint	Stored in NVM - Encrypted	Encrypted	Automated	Electronic	KTS-1
Encrypted using AES-CBC (using SD-KDEK) (RFC 3394 method) and transported using platform SCP03	External endpoint	Stored in NVM encrypted with Approved AES CBC with OS-MKEK	Encrypted	Automated	Electronic	KTS-1
Entered encrypted with	External endpoint	Stored in NVM -	Encrypted	Automated	Electronic	KTS-1

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
the previous SD-KDEK		Encrypted				
Not Imported	N/A	N/A	Plaintext	N/A	N/A	
Not Exported	N/A	N/A	Plaintext	N/A	N/A	
Exported using Approved KTS	Stored in NVM - Encrypted	External Endpoint	Encrypted	Automated	Electronic	KTS-1
Entered during manufacturing/pre-personalization	Vendor generated at manufacture/during pre-personalization	Stored in NVM - Encrypted	Encrypted	N/A	N/A	
Entered during manufacturing/pre-personalization - plaintext storage	Vendor generated at manufacture/during pre-personalization	Stored in NVM	Plaintext	N/A	N/A	

Table 18: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Power-off	Ephemeral parameter	Cleared upon power cycling the module	Operator initiation
Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	Overwritten with zeroes	Setting the module to the TERMINATE state triggers zeroisation and renders the module unusable	Operator initiation
Destroyed because of OS-MKEK zeroisation	-	Zeroising the root of trust causes zeroisation of SSPs derived from it	Module initiation
Cleared immediately following return to the caller	Ephemeral shared secret	Shared secret computed is zeroised post being returned to caller (for use internal to the module)	Module initiation

Table 19: SSP Zeroization Methods

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
OS-DRBG-EI	Random value from ENT (P) used to seed the AES-256 CTR_DRBG	384 bits - 384 bits	Entropy Input - CSP	Entropy Source		Counter DRBG
OS-DRBG-SEED	Seed provided to the CTR_DRBG	384 bits - 384 bits	DRBG seed - CSP	Counter DRBG		Counter DRBG
OS- DRBG-KEY	Current DRBG state value (per IG D.L Resolution 3.)	256 bits - 256 bits	DRBG State Value - CSP	Counter DRBG		Counter DRBG
OS-DRBG-V	Current DRBG state value (per IG D.L Resolution 3.)	256 bits - 256 bits	DRBG State Value - CSP	Counter DRBG		Counter DRBG
OS-SKEK	Used to build OS-MKEK	128 bits - 128 bits	Symmetric key - CSP			Symmetric Cipher Crypto Operations
OS-MKEK	Used to encrypt all secret and private key data stored in NVM	128 bits - 128 bits	Symmetric key - CSP	OS-MKEK Generation		Symmetric Cipher Crypto Operations
SD-KENC	Used to derive SD-SENC	128 bits - 128 bits	Symmetic key - CSP			SCP03/secure channel
SD-KMAC	Used to derive SD-SMAC	128 bits - 128 bits	Symmetric key - CSP			Symmetric Cipher Crypto Operations
SD-KDEK	Sensitive data decryption key used to decrypt CSPs	128 bits - 128 bits	Symmetric key - CSP			Symmetric Cipher Crypto Operations KTS-1

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
SD-SENC	Session encryption key used to secure channel data	128 bits - 128 bits	Symmetric key - CSP		SCP03/secure channel	Symmetric Cipher Crypto Operations
SD-SMAC	Session MAC key used to verify inbound secure channel data integrity	128 bits - 128 bits	Symmetric key - CSP		KDF SP800-108 (A2713)	Symmetric Cipher Crypto Operations
SD-RMAC	Session MAC key used to verify outbound secure channel data integrity	128 bits - 128 bits	Symmetric key - CSP		KDF SP800-108 (A2713)	Symmetric Cipher Crypto Operations
DAP-DAPK	ECC public key used for Mandated DAP	P-256 - 256 bits	Public key - Neither			EC Crypto Operations
RSA-PRIV-KEY	Used for performing RSA cryptographic operations (digital signature, key transport)	2048, 3072, 4096 bits - 112, 128, 152 bits	Private key - CSP	Asymmetric Key Generation		RSA Crypto Operations
EC-PRIV-KEY	Static EC key for performing ECDSA and ECDH KAS operations	P-256, P-384 - 128, 192 bits	Private key - CSP	Asymmetric Key Generation		EC Crypto Operations
EC-KAS-Z	Ephemereal EC CDH shared secret (Z) value generated as an input to a	P-256, P-384 - 128, 192 bits	Shared Secret - CSP		KAS-1	KAS-1

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	key agreement (KAS) operation					
RSA-PUB-KEY	Used for performing RSA cryptographic operations (digital signature, key transport)	2048, 3072, 4096 bits - 112, 128, 152 bits	Public key - PSP	Asymmetric Key Generation		RSA Crypto Operations
EC-PUB-KEY	Static EC key for performing ECDSA and ECDH KAS operations.	P-256, P-384 - 128, 192 bits	Public key - PSP	Asymmetric Key Generation		EC Crypto Operations
EC-KAS-PUB	Ephemeral EC public key used in key establishment (KAS) operation.	P-256, P-384 - 128, 192 bits	Public key - PSP			KAS-1
Marriage Code	Used for authenticating to the module	20-byte - 20-byte	Authentication data - CSP			
Application AES Communication	Encrypt/Decrypt of data passing through the module	256 bits - 256 bits	Symmetric key - CSP			Symmetric Cipher Crypto Operations
Owner PIN	Used for authenticating to the module	8-byte - 8-byte	Authentication data - CSP			
Passive Authentication Key	Used for authenticating to the module	P-384 - 192 bits	Public key - CSP			ECDSA SigVer (FIPS186-5) (A6534)
Active Authentication	Used for mutual authenticatio	256 bits - 256 bits	Symmetric key - CSP			AES-ECB (A2713)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	n to/from the module					

Table 20: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
OS-DRBG-EI	Not Imported Not Exported	Temporarily stored in RAM :Plaintext	Until power-cycle	Power-off Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
OS-DRBG-SEED	Not Imported Not Exported	Temporarily stored in RAM :Plaintext	Until power-cycle	Power-off Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
OS- DRBG-KEY	Not Imported Not Exported	Stored in NVM :Plaintext		Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
OS-DRBG-V	Not Imported Not Exported	Stored in NVM :Plaintext		Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
OS-SKEK	Entered during manufacturing/pre-personalization - plaintext storage	Stored in NVM :Plaintext		Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
OS-MKEK	Not Imported Not Exported	Stored in NVM :Plaintext		Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
SD-KENC	Encrypted using AES-CBC (using SD-KDEK) (RFC 3394 method) and transported using platform SCP03 Exported using Approved KTS Entered during manufacturing/pre-personalization	Stored in NVM - Encrypted:Encrypted	Until the OS-MKEK is zeroised	Destroyed by termination of the module (LifeCycle/Perform Zeroisation service) Destroyed because of OS-MKEK zeroisation	
SD-KMAC	Encrypted using AES-CBC (using SD-KDEK) (RFC 3394 method) and transported using platform SCP03 Exported using Approved KTS Entered during manufacturing/pre-personalization	Stored in NVM - Encrypted:Encrypted	Until the OS-MKEK is zeroised	Destroyed by termination of the module (LifeCycle/Perform Zeroisation service) Destroyed because of OS-MKEK zeroisation	
SD-KDEK	Entered encrypted with the previous SD-KDEK Entered during manufacturing/pre-personalization	Stored in NVM - Encrypted:Encrypted	Until the OS-MKEK is zeroised	Destroyed by termination of the module (LifeCycle/Perform Zeroisation service) Destroyed because of OS-MKEK zeroisation	
SD-SENC	Not Imported Not Exported	Temporarily stored in RAM :Plaintext	Until power-cycle	Power-off Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
SD-SMAC	Not Imported Not Exported	Temporarily stored in RAM :Plaintext	Until power-cycle	Power-off Destroyed by termination of	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				the module (LifeCycle/Perform Zeroisation service)	
SD-RMAC	Not Imported Not Exported	Temporarily stored in RAM :Plaintext	Until power-cycle	Power-off Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
DAP-DAPK	Entered during manufacturing/pre-personalization - plaintext storage	Stored in NVM :Encrypted		Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
RSA-PRIV-KEY	Imported in secure channel specified by GP-Amd-I	Stored in NVM - Encrypted:Encrypted	Until the OS-MKEK is zeroised	Destroyed by termination of the module (LifeCycle/Perform Zeroisation service) Destroyed because of OS-MKEK zeroisation	
EC-PRIV-KEY	Imported in secure channel specified by GP-Amd-I	Stored in NVM - Encrypted:Encrypted	Until the OS-MKEK is zeroised	Destroyed by termination of the module (LifeCycle/Perform Zeroisation service) Destroyed because of OS-MKEK zeroisation	
EC-KAS-Z		Temporarily stored in RAM :Plaintext	While in use	Destroyed by termination of the module (LifeCycle/Perform Zeroisation service) Cleared immediately	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				following return to the caller	
RSA-PUB-KEY	Imported in secure channel specified by GP-Amd-I	Stored in NVM - Encrypted:Encrypted	Until the OS-MKEK is zeroised	Destroyed by termination of the module (LifeCycle/Perform Zeroisation service) Destroyed because of OS-MKEK zeroisation	
EC-PUB-KEY	Imported in secure channel specified by GP-Amd-I	Stored in NVM - Encrypted:Encrypted	Until the OS-MKEK is zeroised	Destroyed by termination of the module (LifeCycle/Perform Zeroisation service) Destroyed because of OS-MKEK zeroisation	
EC-KAS-PUB	Imported in secure channel specified by GP-Amd-I Not Exported	Temporarily stored in RAM :Plaintext	Until power-cycle	Power-off Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
Marriage Code	Imported in secure channel specified by GP-Amd-I	Stored in NVM - Encrypted:Encrypted		Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
Application AES Communication	Imported in secure channel specified by GP-Amd-I	Stored in NVM - Encrypted:Encrypted		Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
Owner PIN	Imported in secure channel specified by GP-Amd-I	Stored in NVM - Encrypted:Encrypted		Destroyed by termination of the module (LifeCycle/Perfor	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				m Zeroisation service)	
Passive Authentication Key	Imported in secure channel specified by GP-Amd-I	Stored in NVM - Encrypted:Encrypted		Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	
Active Authentication	Imported in secure channel specified by GP-Amd-I	Stored in NVM - Encrypted:Encrypted		Destroyed by termination of the module (LifeCycle/Perform Zeroisation service)	

Table 21: SSP Table 2

9.5 Transitions

- The usage of SHA-1 is deprecated through December 31, 2030, for non-digital signature applications and disallowed i.e., non-approved and not allowed thereafter. The module uses SHA-1 for non-digital signature applications, specifically, as the underlying hash function (PRF) for HMAC and NIST SP 800-56Cr1 HKDF.
- Usage of keys with a minimum-security strength of 112 bits i.e. RSA mod 2048 bits and ECDSA P-224 curve is deprecated through December 31, 2030, and disallowed thereafter. January 1, 2031, and beyond a minimum of 128 bits of security strength will be required.

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
Firmware Integrity	32-bit CRC performed over all code located in Flash	32-bit CRC	SW/FW Integrity	9000	-

Table 22: Pre-Operational Self-Tests

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CBC (A2713) - Encrypt	128-bit	KAT	CAST	The APDU code 9000 signifies success	Encrypt	Performed automatically on every boot
AES-CBC (A2713) - Decrypt	128-bit	KAT	CAST	The APDU code 9000 signifies success	Decrypt	Performed automatically on every boot
AES-CMAC (A2713) - Encrypt	128-bit	KAT	CAST	The APDU code 9000 signifies success	Encrypt	Performed automatically on every boot
AES-CMAC (A2713) - Decrypt	128-bit	KAT	CAST	The APDU code 9000 signifies success	Decrypt	Performed automatically on every boot
Counter DRBG (A2713)	256-bit	KAT	CAST	The APDU code 9000 signifies success	Health Tests: Generate, Reseed, Instantiate functions per Section 11 in NIST SP800-90Ar1	Performed automatically on every boot
ECDSA SigGen (FIPS186-5) (A6534)	P-521 SHA-256	KAT	CAST	The APDU code 9000 signifies success	Signature Generation	Performed automatically on every boot
ECDSA SigVer (FIPS186-5) (A6534)	P-521 SHA-256	KAT	CAST	The APDU code 9000 signifies success	Signature Verification	Performed automatically on every boot
KAS-ECC-SSC Sp800-	P-256 Scheme: onePassDh: KAS Role:	KAT	CAST	The APDU code 9000	Shared Secret Computation (Z)	Performed automatically on every boot

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
56Ar3 (A2713)	initiator, responder			signifies success		
KDA HKDF Sp800-56Cr1 (A2713)	SHA-1, SHA2-256, SHA2-384, SHA2-512	KAT	CAST	The APDU code 9000 signifies success	Key Derivation Function per NIST SP 800-56Cr1	Performed automatically on every boot
KDF SP800-108 (A2713) - AES-128	Counter mode with AES-128	KAT	CAST	The APDU code 9000 signifies success	Key Derivation Function per NIST SP 800-108r1 (Counter mode)	Performed automatically on every boot
KDF SP800-108 (A2713) - HMAC-SHA1	Feedback Mode with HMAC-SHA1	KAT	CAST	The APDU code 9000 signifies success	Key Derivation Function per NIST SP 800-108r1 (Counter mode)	Performed automatically on every boot
RSA SigGen (FIPS186-5) (A6535)	2048-bit SHA2-256	KAT	CAST	The APDU code 9000 signifies success	Signature Generation	Performed automatically on every boot
RSA SigVer (FIPS186-5) (A6535)	2048-bit SHA2-256	KAT	CAST	The APDU code 9000 signifies success	Signature Verification	Performed automatically on every boot
SHA-1 (A2713)	SHA-1	KAT	CAST	The APDU code 9000 signifies success	Hash generation	Performed automatically on every boot
SHA2-256 (A2713)	SHA2-256	KAT	CAST	The APDU code 9000 signifies success	Hash generation	Performed automatically on every boot

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA2-512 (A2713)	SHA2-512	KAT	CAST	The APDU code 9000 signifies success	Hash generation	Performed automatically on every boot
Generate PCT	Pairwise consistency test performed when an asymmetric key pair is generated for RSA or ECC.	PCT	PCT	The APDU code 9000 signifies success	-	Performed automatically on generation of keypairs
Signature PCT	Pairwise consistency test performed when a signature is generated for RSA or ECDSA	PCT	PCT	The APDU code 9000 signifies success	-	Performed automatically on generation of keypairs
Firmware Load Test	ECDSA P-256 with SHA2-256	Load Test	SW/FW Load	The APDU code 9000 signifies success	Signature Verification based on ECDSA P-256 with SHA2-256	Upon loading of firmware from an external source (e.g. the applet).
NIST SP800-90B ENT (P) Repetition Count Test (RCT)	-	NIST SP 800-90B Health Test	CAST	Implicit, based on output of entropy	-	Performed continuously by the entropy source
NIST SP800-90B ENT (P) Developer Defined Heath Test Transition Count Test	-	NIST SP 800-90B Health Test	CAST	Implicit, based on output of entropy	-	Performed continuously by the entropy source
NIST SP800-90B ENT (P) Developer	-	NIST SP 800-90B	CAST	Implicit, based on output of entropy	-	Performed continuously by the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Defined Heath Test Chi-Square Test		Health Test				entropy source
NIST SP800-90B ENT (P) Developer Defined Heath Test Amplitude Limiter Analog Test	-	NIST SP 800-90B Health Test	CAST	Implicit, based on output of entropy	-	Performed continuously by the entropy source

Table 23: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Firmware Integrity	32-bit CRC	SW/FW Integrity	Repeated after every 500,000 CAPDUs/commands	Automatic execution per module design (the test is repeated after every 500,000 CAPDUs/commands)

Table 24: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CBC (A2713) - Encrypt	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CBC (A2713) - Decrypt	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
AES-CMAC (A2713) - Encrypt	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
AES-CMAC (A2713) - Decrypt	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Counter DRBG (A2713)	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
ECDSA SigGen (FIPS186-5) (A6534)	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
ECDSA SigVer (FIPS186-5) (A6534)	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KAS-ECC-SSC Sp800-56Ar3 (A2713)	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
KDA HKDF Sp800-56Cr1 (A2713)	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
KDF SP800-108 (A2713) - AES-128	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KDF SP800-108 (A2713) - HMAC-SHA1	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
RSA SigGen (FIPS186-5) (A6535)	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
RSA SigVer (FIPS186-5) (A6535)	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
SHA-1 (A2713)	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
SHA2-256 (A2713)	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
SHA2-512 (A2713)	KAT	CAST	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Generate PCT	PCT	PCT	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
Signature PCT	PCT	PCT	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.
Firmware Load Test	Load Test	SW/FW Load	On Demand	Manually, all the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
NIST SP800-90B ENT (P) Repetition Count Test (RCT)	NIST SP 800-90B Health Test	CAST	On Demand	Manually, by rebooting the module/resetting the entropy source
NIST SP800-90B ENT (P) Developer Defined Heath Test Transition Count Test	NIST SP 800-90B Health Test	CAST	On Demand	Manually, by rebooting the module/resetting the entropy source
NIST SP800-90B ENT (P) Developer Defined Heath Test Chi-Square Test	NIST SP 800-90B Health Test	CAST	On Demand	Manually, by rebooting the module/resetting the entropy source
NIST SP800-90B ENT (P) Developer Defined Heath Test Amplitude Limiter Analog Test	NIST SP 800-90B Health Test	CAST	On Demand	Manually, by rebooting the module/resetting the entropy source

Table 25: Conditional Periodic Information

The JCOP OS is intended to execute pre-operational self-tests (rather than conditional self-tests) periodically, since these tests are pre-defined to be executed post resets. The periodic time will thus always start from zero. RAM can be used to manage the periodic time interval which is the number of execution events e.g., number of APDUs/commands received (the tests are repeated after every 500,000 CAPDUs/commands).

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Hard Error (MUTE) state	All the Self-Tests can be performed on-demand with the GET DATA APDU command (Info service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00. The expected result is FE04DF4B0120. The return code 9000 signifies success. In case of a failure, the module	In case of failure of a pre-operational, conditional self-test or NIST SP 800-90B compliant entropy source health test.	A reset of the module can be attempted but in the event that the error persists, the module must be returned to the vendor	66A7

Name	Description	Conditions	Recovery Method	Indicator
	enters a hard error (MUTE) state and returns a code/status indicator			

Table 26: Error States

10.5 Operator Initiation of Self-Tests

The pre-operational firmware integrity self-test must be completed successfully prior to any other use of cryptography by the module. The Cryptographic Algorithm Self-Tests are performed at boot. The conditional self-tests are performed when the corresponding conditions occur. If one of the self-tests fails, the system is halted and will start again after a reset.

All the Self-Tests can be performed on-demand with the GP GET DATA APDU command (*Info* service) with the following parameters: CLA = 80, INS = CA, P1 = 00, P2 = FE, Lc = 04, Incoming Data = DF4B0120, and Le = 00. The expected result is FE04DF4B0120.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The Module will be delivered with the Approved mode enabled by default at manufacturing by the vendor. The vendor signs and loads the SAM applet to run in the Approved mode of operation using a vendor-created DAP script and creates data structures inside using pre-personalization scripts also provided by them. The applet only supports the approved mode of operation. Upon shipment, the Crypto Officer (CO) can do the following:

a. Verify that the operating system (JCOP OS) is running in the Approved mode of operation by using instructions provided in Section 2.4 of this document.
b. Perform personalization i.e. enter commands (APUDs) for entering biometric data (not SSPs) into the module. This is done by authenticating to the OS (using the Active Authentication mechanism) using software provided by the vendor (Perso software) and another applet (SSID). The data is encrypted first and provided to the SAM via APDU commands. The biometric data stored in the SAM applet are extracted by a vendor device (called IPC/Triton/Vega-1) with/alongside which the module is intended to be used. This device captures the biometric data from the device's fingerprint reader at the point of use and matches it against the biometric data stored in the SAM applet.

11.2 Administrator Guidance

The aforementioned information can also be found in the *JCOP 4.5 User guidance and administrator manual*.

11.3 Non-Administrator Guidance

The aforementioned information can also be found in the *JCOP 4.5 User guidance and administrator manual*.

11.4 Design and Rules

All configuration management items are managed using an automated configuration management system. The module is designed to allow the testing of all provided security-related services. All firmware is implemented using a high-level language and is designed in a manner that avoids the use of code, parameters, or symbols not necessary for the module’s functionality and execution. The firmware is designed and implemented in a manner that avoids the use of code, parameters or symbols not necessary for the module's functionality and execution.

11.5 Maintenance Requirements

No specific maintenance requirements apply to the module.

11.6 End of Life

The module must be zeroised (using the Perform zeroisation service per Section 4.3 of this document) in order to perform secure sanitization of the module.

12 Mitigation of Other Attacks

12.1 Attack List

The module is protected against the following non-invasive attacks: SPA, DPA, Timing Analysis and Fault Induction using a combination of firmware and hardware countermeasures. Protection features include detection of out-of-range supply voltages, frequencies or temperatures, fault induction mitigations like light sensors, voltage glitch sensors and an active shield, and detection of illegal address or instruction.

12.2 Mitigation Effectiveness

All cryptographic computations and sensitive operations such as critical data comparison provided by the module are designed to be resistant to timing and power analysis. Sensitive operations are performed in constant time, regardless of the execution context (parameters, keys, etc.), owing to a combination of hardware and firmware features. In addition to the non-invasive attacks, the module also uses standard passivation techniques and is protected by active shielding (a grid of top metal layer wires with tamper response) which qualifies for classification under mitigation of other attacks.