THALES

‘eToken 5110+ FIPS’
FIPS 140-2 Cryptographic Module
Non-Proprietary Security Policy Level 2

Table of Contents

Table of Tables

Table of Figures

References

Acronym	Full Specification Name
[FIPS140-2]	NIST, *Security Requirements for Cryptographic Modules*, May 25, 2001
[GlobalPlatform]	*GlobalPlatform Consortium: GlobalPlatform Card Specification 2.2.1,* January 2011, http://www.globalplatform.org
[ISO 7816]	ISO/IEC 7816-1:1998 *Identification cards -- Integrated circuit(s) cards with contacts -- Part 1: Physical characteristics* ISO/IEC 7816-2:2007 *Identification cards -- Integrated circuit cards -- Part 2: Cards with contacts -- Dimensions and location of the contacts* ISO/IEC 7816-3:2006 *Identification cards -- Integrated circuit cards -- Part 3: Cards with contacts -- Electrical interface and transmission protocols* ISO/IEC 7816-4:2005 *Identification cards -- Integrated circuit cards -- Part 4: Organization, security and commands for interchange*
[ISO 14443]	*Identification cards – Contactless integrated circuit cards – Proximity cards* ISO/IEC 14443-1:2008 Part 1*: Physical characteristics* ISO/IEC 14443-2:2010 Part 2: *Radio frequency power and signal interface* ISO/IEC 14443-3:2011 Part 3: *Initialization and anticollision* ISO/IEC 14443-4:2008 Part 4: *Transmission protocol*
[JavaCard]	*Java Card 3.0.5 Runtime Environment (JCRE) Specification* *Java Card 3.0.5 Virtual Machine (JCVM) Specification* *Java Card 3.0.5 Application Programming Interface* Published by Sun Microsystems, October 2015.
[SP800-131A]	NIST Special Publication 800-131A revision 2, *Transitioning the Use of Cryptographic Algorithms and Key Lengths, March 2019*
[SP 800-133]	NIST Special Publication 800-133, revision 2, *Recommendation for Cryptographic Key Generation*, June 2020
[SP 800-38B]	NIST Special Publication 800-38B, *Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication*, May 2005
[SP 800-90A]	NIST Special Publication 800-90A revision 1, *Recommendation for the Random Number Generation Using Deterministic Random Bit Generators (Revised)*, June 2015
[SP 800-67]	NIST Special Publication 800-67 revision 2, *Recommendation for the Triple Data Encryption Algorithm (Triple-DES) Block Cipher*, November 2017
[FIPS113]	NIST, *Computer Data Authentication*, FIPS Publication 113, 30 May 1985.
[FIPS 197]	NIST, *Advanced Encryption Standard (AES)*, FIPS Publication 197, November 26, 2001.
[PKCS#1]	*PKCS #1 v2.1: RSA Cryptography Standard*, RSA Laboratories, June 14, 2002
[FIPS 186-4]	NIST, Digital Signature Standard (DSS), FIPS Publication 186-4, July, 2013
[SP 800-56A]	NIST Special Publication 800-56A revision 3, *Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography*, April 2018
[SP 800-56B]	NIST Special Publication 800-56B revision 2, *Recommendation for Pair-Wise Key-Establishment Schemes Using Integer Factorization Cryptography*, March 2019

Acronym	Full Specification Name
[FIPS 180-4]	NIST, *Secure Hash Standard*, FIPS Publication 180-4, August 2015
[SP 800-38F]	NIST Special Publication 800-38F, *Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping*, December 2012
[IG]	NIST, *Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program,* last updated Nov 5, 2021
[MD]	Microsoft, MD – Smart Card Mini Driver v7.07, April 20th , 2017

Table 1 – References

Acronyms and definitions

Acronym	Definition
FW	Firmware
GP	Global Platform
CVC	Card Verifiable Certificate
MMU	Memory Management Unit
OP	Open Platform
RMI	Remote Method Invocation

Table 2 – Acronyms and Definitions

1 Introduction

This document defines the Security Policy for the 'eToken 5110+ FIPS' which comprises the 5110 USB MCU FW, the IDCore3130 platform and the IDPrime930 (v4.5) and herein denoted as Cryptographic Module, or Module. The Cryptographic Module or CM, validated to FIPS 140-2 overall Level 2, is a USB token that contains a secure controller (SC) module implementing the Global Platform operational environment, with Card Manager, and the IDPrime applet (associated to MSPNP applet V1.2)

The CM is a limited operational environment under the FIPS 140-2 definitions. The CM includes a firmware load function to support necessary updates. New firmware versions within the scope of this validation must be validated through the FIPS 140-2 CMVP. Any other firmware loaded into this module is out of the scope of this validation and requires a separate FIPS 140-2 validation. The CM also includes the USB MCU FW firmware load service to support necessary updates of the USB controller FW.

The FIPS 140-2 security levels for the *Module* are as follows:

Security Requirement	Security Level
Cryptographic Module Specification	2
Cryptographic Module Ports and Interfaces	2
Roles, Services, and Authentication	3
Finite State Model	2
Physical Security	3
Operational Environment	N/A
Cryptographic Key Management	2
EMI/EMC	3
Self-Tests	2
Design Assurance	3
Mitigation of Other Attacks	2

Table 3 – Security Level of Security Requirements

The CM implementation is compliant with:

- [ISO 7816] Parts 1-4
- [JavaCard]
- [GlobalPlatform]
- [MD]

1.1 IDPrime Applet

IDPrime Applet (V4.5) is a Java applet that provides all the necessary functions to integrate a smart card in a public key infrastructure (PKI) system, suitable for identity and corporate security applications. It is also useful for storing information about the cardholder and any sensitive data. IDPrime Applet implements state–of–the–art security and conforms to the latest standards for smart cards and PKI applications. It is also fully compliant with digital signature law.

The IDPrime Applet, designed for use on JavaCard 3.0.5 and Global Platform 2.2.1 compliant smart cards.

The main features of IDPrime Applet are as follows:

- Digital signatures—these are used to ensure the integrity and authenticity of a message. (RSA, ECDSA)
- Storage of sensitive data based on security attributes
- PIN management.
- Secure messaging based on the AES algorithms.
- Public key cryptography, allowing for RSA keys and ECDSA keys
- Storage of digital certificates—these are issued by a trusted body known as a certification authority (CA) and are typically used in PKI authentication schemes.
- CVC verification
- Decryption RSA , ECDH
- On board key generation (RSA, ECDSA)
- Mutual authentication between IDPrime Applet and the terminal (ECDH)
- Support of integrity on data to be signed.
- Secure Key Injection according to Microsoft scheme.
- Touch Sense feature (not available on smart card, only on Token)
- PIN Single Sign On (PIN SSO)
- Reinit feature
- Extended APDU support

MSPNP applet is associated to IDPrime applet and offers:

- GUID tag reading, defined in Microsoft Mini Driver specification.

2 Cryptographic Module Specification

2.1 Hardware and Firmware Versions - mode of operation

The **Cryptographic Module (CM) 'eToken 5110+ FIPS'** is composed of:
- USB MCU Firmware,
- Smart card Firmware with the IDCore3130 platform, the IDPrime930 (v4.5) with MSPNP (v1.2) applets,

Hereafter is the list of firmware versions of the CM 'eToken 5110+ FIPS':

Firmware: 5110+ FIPS - FW ver-22-00-0000[1]**,** IDCore3130 - Build12G, IDPrime 930 Applet V4.5.0F, MSPNP Applet V1.2.

The module's hardware versions are the followings:

Reference	**SAFENET** ETOKEN 5110+FIPS (L2)	**SAFENET ETOKEN** 5110+FIPS (L2 NON MANAGED)
Part Numbers	**909-000156-001 - A3016629**	**909-000157-001 - A3016639**

Note: L2 NON MANAGED is relative to the IDPrime configuration requiring the use of the role ID IO for recycling/reinitializing the card.

The MCU Manufacturer is STMicroelectronics, which is not a configurable part of the module, and the following is provided for information purposes:

Hardware component	Manufacturer	Reference
USB MCU	STM	STM32F042K6U6TR

[1] To identify this version, please refer to Table 12 - Get Firmware version answer structure

Figure 1 depicts the Module at the cryptographic boundary:

Figure 1 – 'eToken 5110+ FIPS' Crypto Boundary

2.2 Cryptographic Functionality

The Module operating system implements the FIPS Approved and Non-FIPS Approved cryptographic function listed in Tables below.

Algorithm	Description	Cert #
AES	[FIPS 197] Advanced Encryption Standard algorithm. The Module supports 128-, 192- and 256-bit key lengths with ECB and CBC encrypt/ decrypt modes.	A1930
AES CMAC	[SP 800-38B] The Module supports generation and verification with 128-, 192- and 256-bit key lengths.	A1930
CKG	[SP 800-133] Section 6.1, Section 7.1: The Module generates symmetric keys and seeds to be used in asymmetric key generation directly from unmodified DRBG output.	Vendor Affirmed
DRBG	[SP 800-90A] Deterministic Random Bits Generator (256-bit security strength CTR-DRBG based on AES).	A1930
ECDSA	[FIPS 186-4] Elliptic Curve Digital Signature Algorithm using the NIST defined curves. – Key pair generation: P-224, P-256, P-384 and P-521 curves. – Signature generation: P-224, P-256, P-384 and P-521 curves with SHA-2. – Signature verification: P-224, P-256, P-384 and P-521 curves (approved SHA sizes of the CM).	A1930

KBKDF	[SP 800-108] The Module supports AES CMAC 128-, 192- and 256-bit key lengths.	A1930
KTS	Use of approved [FIPS 197] AES encryption method with the combination of approved Authentication method [SP 800-38B] AES CMAC The Module supports 128-, 192- and 256-bit key lengths. The Module supports 256-bit key length for Applet Secure Messaging. Provides between 112 and 150 bits of encryption strength.	A1930
SHA-1 SHA-2	[FIPS 180-4] Secure Hash Standard compliant one-way (hash) algorithms. The Module supports the SHA-1 (160 bits), SHA-2 (224- bit, 256-bit, 384-bit, 512-bit) variants.	A1930
Triple-DES	[SP 800-67] Triple Data Encryption Algorithm. The Module supports the 3-Key options; CBC and ECB decrypt modes. The CM restricts Triple-DES decryptions to 2^16 per key. After a counter for a given key reach 2^16, the key is blocked.	A1930
RSA	[FIPS 186-4] [PKCS#1 v1.5 and PSS] RSA algorithms. – Key pair generation using 2048-bit keys. – Signature generation using 2048-bit keys with SHA-2. – Signature verification using 1024, 2048-bit keys (approved SHA sizes of the CM). Note that RSA-1024 verification and the use of SHA-1 for any RSA verification is allowed for legacy-use only.	A1930
RSA CRT	[FIPS 186-4] [PKCS#1 v1.5 and PSS] RSA CRT algorithm. – Key pair generation using 2048-, 3072- and 4096-bit keys; – Signature generation using 2048-, 3072- and 4096-bit keys with SHA-2; – Signature verification using 1024-, 2048-, 3072- and 4096-bit keys (approved SHA sizes of the CM). Note that RSA-1024 verification and the use of SHA-1 for any RSA verification is allowed for legacy-use only.	A1930
KAS SSC ECC	[SP 800-56A] standalone Key Agreement Scheme SSC (section 5.7.1.2: ephemeral Unified) using the NIST defined curves: P-521.	A1930
KTS-RSA	[SP 800-56B] RSA key transport scheme (section 9.2.3 KTS-OAEP-basic) using 2048-, 3072- and 4096-bit keys. Provides between 112 and 150 bits of encryption strength.	A1930
DP RSA (CVL)	[SP 800-56B] RSA decryption primitive (section 9.2.3 KTS-OAEP-basic) using 2048-bit keys.	A1930
KDA	[SP 800-56C] Key Agreement Scheme Key Derivation function (section 4: One-step Key Derivation – Option 1 with approved hash function) using the NIST defined curve: P-521, and SHA-256.	A1930
KAS	Use of [SP 800-56A] KAS SSC ECC with the combination of key derivation function [SP 800-56C] KAS KDF. Provides 128 bits of encryption strength.	A1930

Table 4 – List of the algorithms/modes utilized by the module (Smart Card FW)

Note: Not all algorithms/modes that appear on the module's CAVP certificates are utilized by the module.

Algorithm	Description
RSA key wrap	Key unwrapping using 2048, 3072 or 4096 bit keys. Key establishment methodology provides between 112 and 150 bits of strength (for PKCS1 v1.5)

Table 5 – Non-FIPS Approved but Allowed Cryptographic Functions utilized by the module (Smart Card FW)

Algorithm	Description	Cert #
RSA Signature Verification (In the USB MCU FW)	[FIPS 186-4] RSA signature verification. The Module follows PKCS#1 and is CAVP validated for 2048 bit key length.	A1911
SHA256 (In the USB MCU FW)	[FIPS 180-4] Secure Hash Standard compliant one-way (hash) algorithms.	A1911

Table 6 – List of the algorithms/modes utilized by the module (USB MCU FW)

The CM includes an uncallable DES implementation. This algorithm is not used and no security claims are made for its presence in the Module.

FIPS approved security functions used specifically by the **IDPrime Applet** are:

- **DRBG**
- **AES CMAC**
- **AES**
- **Triple-DES**
- **RSA**
- **ECDSA**
- **SHA-1, SHA-224, SHA-256, SHA-384, SHA-512**
- **KAS SSC ECC**
- **KAS KDF one-step**

(Note: no security function is used in **MSPNP applet**)

2.3 Smart Card Firmware

2.3.1 Block diagram

Figure 2 below depicts the Smart Card operational environment and applets.

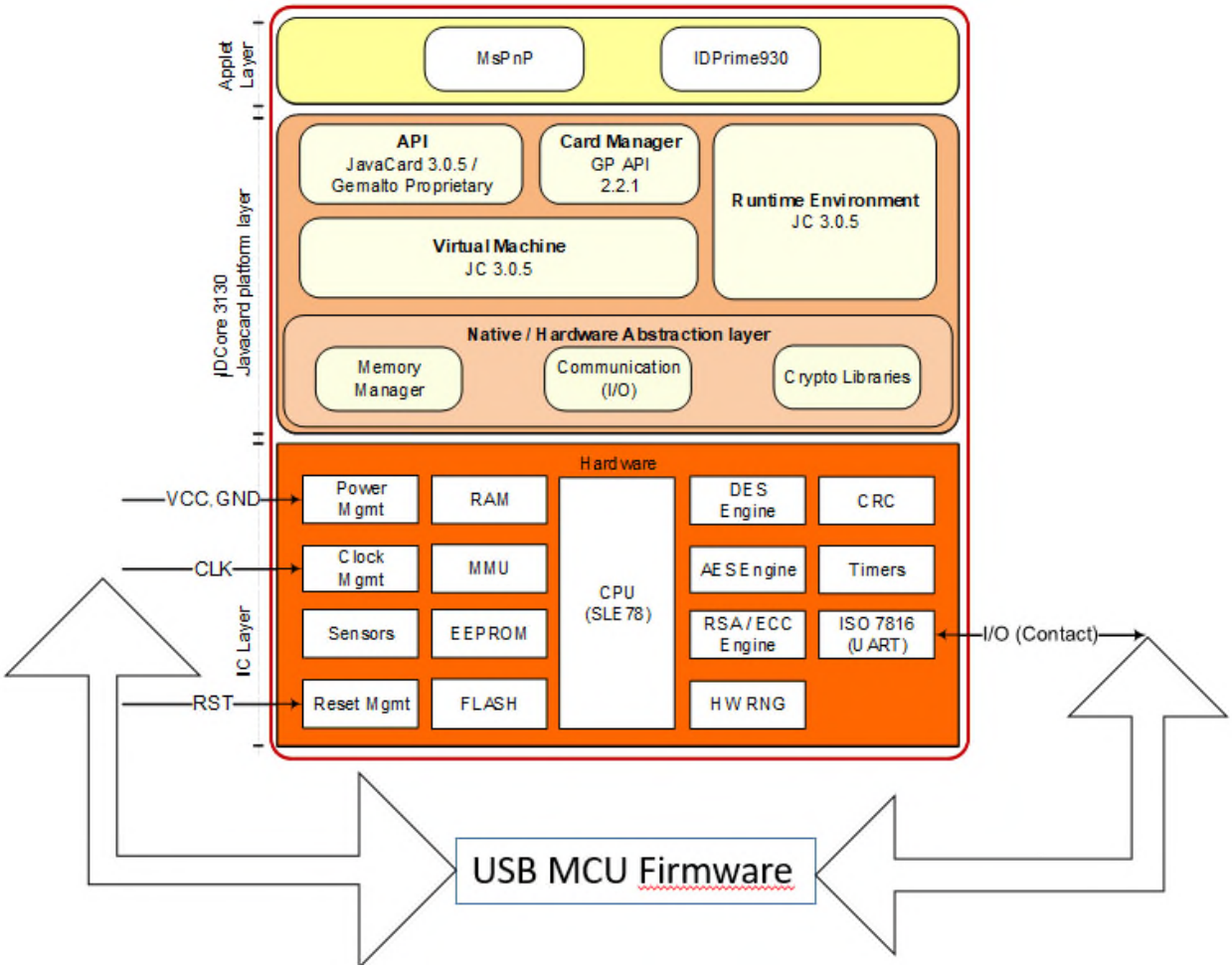

Figure 2 – Smart Card Firmware Block Diagram

The CM supports [ISO7816] T=0 and T=1, and also [ISO14443] T=CL communication protocols.
The CM provides services to both external devices and internal applets as the IDPrime, MsPnP.

Applets, as IDPrime accesses module functionalities via internal API entry points that are not exposed to external entities. External devices have access to CM services by sending APDU commands.

The CM provides an execution sandbox for the IDPrime applet and performs the requested services according to its roles and services security policy.

The CM inhibits all data output via the data output interface while the module is in error state and during self-tests.

The *JavaCard API* is an internal interface, available to applets. Only applet services are available at the card edge (the interfaces that cross the cryptographic boundary).

The *Javacard Runtime Environment* implements the dispatcher, registry, loader, logical channel and RMI functionalities.

The *Virtual Machine* implements the byte code interpreter, firewall, exception management and byte code optimizer functionalities.

The *Card Manager* is the card administration entity – allowing authorized users to manage the card content, keys, and life cycle states.

The *Memory Manager* implements services such as memory access, allocation, deletion, garbage collector.

The *Communication* handler deals with the implementation of ATR/ATS, PSS, T=0 T=1 and T=CL protocols.

The *Cryptography Libraries* implement the algorithms listed in Table 4 – List of the algorithms/modes utilized by the module (Smart Card FW)

2.3.2 Smart Card Firmware versions

The CM is always in the approved mode of operation. To verify that a CM is in the approved mode of operation, select the Card Manager and send the GET DATA commands shown below:

Field	CLA	INS	P1-P2 (Tag)	Le (Expected response length)	Purpose
Value	00	CA	9F-7F	2A	Get CPLC data
			01-03	1D	Identification information (proprietary tag)

The CM responds with the following information:

IDC3130 - CPLC data (tag 9F7F)			
Byte	**Description**	**Value**	**Value meaning**
1-2	IC fabricator	**4090h**	Infineon
3-4	IC type	**7861**	SLE78CFX400VPH
5-6	Operating system identifier	**1291**	Thales
7-8	Operating system release date (YDDD) – Y=Year, DDD=Day in the year	**7334**	Operating System release Date
9-10	Operating system release level	**0100h**	V1.0
11-12	IC fabrication date	xxxxh	Filled in during IC manufacturing
13-16	IC serial number	xxxxxxxxh	Filled in during IC manufacturing
17-18	IC batch identifier	xxxxh	Filled in during IC manufacturing
19-20	IC module fabricator	xxxxh	Filled in during module manufacturing
21-22	IC module packaging date	xxxxh	Filled in during module manufacturing
23-24	ICC manufacturer	xxxxh	Filled in during module embedding
25-26	IC embedding date	xxxxh	Filled in during module embedding
27-28	IC pre-personalizer	xxxxh	Filled in during smartcard preperso
29-30	IC pre-personalization date	xxxxh	Filled in during smartcard preperso
31-34	IC pre-personalization equipment identifier	xxxxxxxxh	Filled in during smartcard preperso
35-36	IC personalizer	xxxxh	Filled in during smartcard personalization
37-38	IC personalization date	xxxxh	Filled in during smartcard personalization
39-42	IC personalization equipment identifier	xxxxxxxxh	Filled in during smartcard personalization

IDC3130 - Identification data (tag 0103)			
Byte	**Description**	**Value**	**Value meaning**
1	Thales Family Name	**B0**	Javacard
2	Thales OS Name	**84**	IDCore family
3	Thales Mask Number	**65**	G286
4	Thales Product Name	**66**	IDCore3130 for IDPrime 930
5	Thales Flow Version	**XX**	**XX** is the version of the flow: ▪ 01h for flow version 01
6	Thales Filter Set	**00**	▪ Major nibble: filter family = 00h ▪ Lower nibble: version of the filter = 00h
7-8	Chip Manufacturer	**4090**	Infineon
9-10	Chip Version	**7861**	SLE78CLFX400VPH
11-12	FIPS configuration	**8F00**	MSByte: b8 : 1 = **conformity to FIPS certificate** b7 : 0 = not applicable b6 : 0 = not applicable b5 : 0 = not applicable b4 : 1 = ECC supported b3 : 1 = RSA CRT supported b2 : 1 = RSA STD supported b1 : 1 = AES supported LSByte: b8 .. b5 : 0 = not applicable b4 : 0 = not applicable (ECC in contactless) b3 : 0 = not applicable (RSA CRT in contactless) b2 : 0 = not applicable (RSA STD in contactless) b1 : 0 = not applicable (AES in contactless) For instance: **8F 00** = FIPS enable (CT only)–AES-RSA CRT/STD-ECC **(Full FIPS)** **8D 00** = FIPS enable (CT only)–AES-RSA CRT-ECC **(FIPS PK CRT) *** **85 00** = FIPS enable (CT only)–AES-RSA CRT **(FIPS RSA CRT)** **00 00** = FIPS disable (CT only)–No FIPS mode **(No FIPS)** **(* default configuration)**

13	FIPS Level for IDPrime product	**02**	02 = FIPS Level 2
14-15	Specific chip ID	**01 30**	01 30 = Contact (IDPrime 930 product)
16-29	RFU	**xx..xxh**	-

Table 7 – Versions and Mode of Operations Indicators

The **IDPrime 930** is identified with an applet version and a software version as follow:

Field	CLA	INS	P1-P2 (Tag)	Le (Expected response length)	Purpose
Value	**00**	**CA**	**DF-30**	**07**	Get Applet Version
			7F-30	**19**	Get Software Version

Table 8 – Applet Version and Software Version imput data

The Applet version is returned without any TLV format as follows:

IDPrime 930 – Applet Version Data (tag DF30)	
Value	**Value Meaning**
34 2E 35 2E 30 2E 46	Applet Version Display value = '4.5.0.F'

Table 9 – Applet Version returned value

The Software Version is returned in TLV format as follows:

IDPrime 930 – Software Version Data (tag 7F30)					
Tag	**Length**				
7F30	17				
		Tag	**Length**	**Value**	**Value meaning**
		C0	0E	49 41 53 20 43 6C 61 73 73 69 63 20 76 34	Applet Label Display value = 'IAS Classic v4'
		C1	07	34 2E 35 2E 30 2E 46	Software Version Display value = '4.5.0.F'

Table 10 – Software Version Returned Values

2.4 USB MCU Firmware

2.4.1 Block diagram

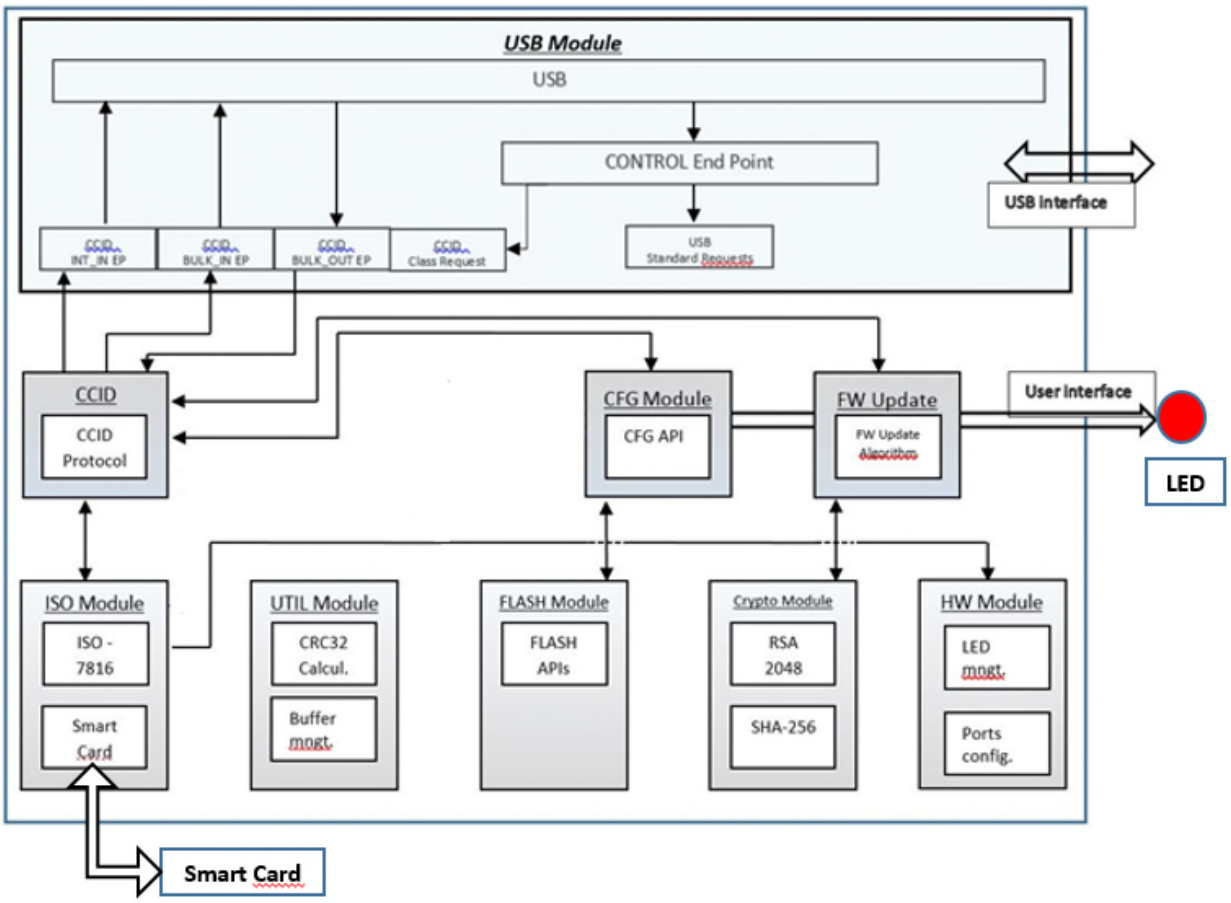

Figure 3 – USB MCU Block Diagram

The CM provides framework for the USB Standard and Class requests including the API dedicated to the CCID protocol. The CM defines an interface for USB MCU firmware update service secured with RSA-2048 PKCS#1 RSASSA-PKCS1-v1_5 signature. The USB MCU FW communicates with the SC OS using ISO-7816 T1 protocol.

The LED functions as status indicator with no connection to Critical Security Parameters, and thus cannot output any sensitive information.

2.4.2 USB MCU firmware Versions

2.4.2.1 Get Firmware version request

Offset	Field	Size (bytes)	Value	Field Description
0x00	Command Id	1	0x02	Get firmware version

Table 11 – Get Firmware version request structure

2.4.2.2 Get Firmware version answer

It is important that the answer starts with 'Gem' (for compatibility with the Thales PC/SC driver)

Offset	Field	Size (bytes)	Value	Field Description
0x00	String answer	20	GemP51-22.00.0000-X2	"GemP51-XX.YY.ZZZZ-X2", where: "P51" to indicate 'eToken'' XX : Firmware major version. ⇨ 22 for 'eToken 5110+ FIPS' YY : Firmware minor version (max 99). ⇨ 00 for 'eToken 5110+ FIPS' ZZZZ: Firmware build (max 9999). ⇨ 00 for 'eToken 5110+ FIPS'

Table 12 - Get Firmware version answer structure

Get Firmware Version command	
Cmd [00] = 02	= .
Resp [00] = 00 47 65 6D 50 35 31 2D 32 32 2E 30 30 2E 30 30	= .GemP51-22.00.00
[10] = 30 30 2D 58 32	= 00-X2

3 Cryptographic Module Ports and Interfaces

3.1 Physical and logical interfaces

'eToken 5110+ FIPS' is the composition of 2 single chips. Two (2) ICs are mounted on a PCB assembly with a connector and passive components, covered by epoxy on both sides, exposing only the LED and the USB connector. The Module is covered within a plastic enclosure. Physical inspection inside the Module boundary is not practical, as the epoxy layer is opaque.

The Module meets commercial-grade specifications for power, temperature, reliability, and shock/vibrations.

The only connector to the module is a contact interface that is fully compliant with USB 2.0. It functions as a slave device to process and respond to commands.

Interface	Description	Logical Mapping
USBDM	USB D- differential data	Data Input, Data Output. Status Output, Control Input
USBDP	USB D+ differential data	Data Input, Data Output, Status Output, Control Input
V_{Bus}	Power supply input	Power Input
GND	Ground (reference voltage)	N/A
LED	LED indicator	Status Output

Table 13 –Physical and Logical Interfaces

The I/O ports of the platform provide the following logical interfaces:

Interface	Physical Ports	APDU Command Fields
Data In	USBDM, USBDP	Command Data Field
Data Out	USBDM, USBDP	Response Data Field
Status Out	USBDM, USBDP, LED	SW1, SW2
Control In	USBDM, USBDP	CLA, INS, P1, P2, Lc, Le
Power	VBus, GND	

Table 14 – Logical Interfaces, Physical Ports and APDU Command Fields

4 Module Critical Security Parameters

All CSPs used by the CM are described in this section. All usages of these CSPs by the CM are described in the services detailed in Section 5. In addition, all keys stored in RAM are zeroized upon power-cycle of the CM.

4.1 Platform Critical Security Parameters

Key	Description / Usage
OS-DRBG-EI	1664-bit random drawn by an external entropy source populated during CM initialization and used as entropy input for the [SP800-90A] DRBG implementation. Provides at least 256 bits of entropy.
OS-DRBG-STATE	16-byte AES state V and 32-byte AES key (or Nonce) used in the [SP800-90A] CTR DRBG implementation.
OS-GLOBALPIN	4 to 16 byte Global PIN value managed by the ISD. Character space is not restricted by the OS. The PIN Policy is managed by applet.
OS-MKDK	AES-128 (SCP03) key used to encrypt OS-GLOBALPIN value.
SD-KENC	AES-128/192/256 (SCP03) master key used by the CO role to derive SD-SENC.
SD-KMAC	AES-128/192/256 (SCP03) master key used by the CO role to derive SD-SMAC.
SD-KDEK	AES-128/192/256 (SCP03) decryption key used by the CO role to decrypt secure channel data.
SD-SENC	AES-128/192/256 (SCP03) Session encryption key used by the CO role to encrypt / decrypt secure channel data.
SD-SMAC	AES-128/192/256 (SCP03) Session MAC key used by the CO role to verify secure channel data integrity.
DAP-SYM	AES-128 (DAP) key optionally loaded in the field and used to verify the MAC signature of packages loaded into the Module.
DAP-ASYM	2048-bit public part of RSA key pair used for Asymmetric Signature verification used to verify the signature of packages loaded into the Module.
DM-TOKEN-SYM	AES-128 Delegate Management Token Symmetric key.
DM-RECEIPT-SYM	AES-128 Delegate Management Receipt Symmetric key.
DM-TOKEN-ASYM	2048-bit public part of RSA key pair used for Delegated Management Token

Table 15 - Platform Critical Security Parameters

Keys with the "SD-" prefix pertain to a Global Platform Security Domain key set. The module supports the Issuer Security Domain at minimum, and can be configured to support Supplemental Security Domains.

4.2 IDPrime Applet Critical Security Parameters

Key	Description / Usage
IDP-SC-SMAC-AES	AES 256 Session key used for Secure Messaging (MAC)
IDP-SC-SENC-AES	AES 256 Session key used for Secure Messaging (Decryption)
IDP-AS-RSA	2048/3072/4096- private part of the RSA key pair used for Asymmetric Signature
IDP-AS-ECDSA	P-224, P-256, P-384, P-521 private part of the ECDSA key pair used for Asymmetric signature
IDP-AC-RSA	2048/3072/4096- private part of the RSA key pair used for Asymmetric Cipher (key wrap, key unwrap)
IDP-ECDH-ECC	P-224, P-256, P-384, P-521 private part of the ECDH key pair used for shared key mechanism
IDP-KG-AS-RSA	2048/3072/4096- private part of the RSA generated key pair used for Asymmetric signature
IDP-KG-AS-ECDSA	P-224, P-256, P-384, P-521 private part of the ECDSA generated key pair used for Asymmetric signature
IDP-KG-AC-RSA	2048/3072/4096- private part of the RSA generated key pair used for Asymmetric cipher (key unwrap)
IDP-KG-AC-ECDH	P-224, P-256, P-384, P-521 private part of the ECDSA generated key pair used for shared key mechanism
IDP-ECDSA-AUTH-ECC	P-224, P-256, P-384, P-521 private part of the ECDSA private key used to Authenticate the card
IDP-SC-DES3	192-bit Triple-DES key used for Admin (ICAA Role) authentication and provides 112 bits of security strength
IDP-SC-P-SKI-AES	AES 128/192/256 Session key used for Secure Key Injection
IDP-SC-T-SKI-AES	AES 128/192/256 Session key used for Secure Key Injection
IDP-SC-PIN-TDES	192-bit Triple-DES key used for PIN encryption (PIN History) and provides 112 bits of security strength
IDP-OWNERPIN	4 to 64 byte PIN value managed by the Applet.
IDP-INITK-AES	256-bits AES key used to authenticate in IO Role

Table 16 – IDPrime Applet Critical Security Parameters

4.3 IDPrime Applet Public Keys

Key	Description / Usage
IDP-KA-ECDH	P-224, P-256, P-384, P-521 ECDH key pair used for Key Agreement (Session Key computation)
IDP-AS-CA-ECDSA-PUB	P-224, P-256, P-384, P-521 CA ECDSA Asymmetric public key entered into the module used for CA Certificate Verification.
IDP-AS-IFD-ECDSA-PUB	P-224, P-256, P-384, P-521 IFD ECDSA Asymmetric public key entered into the module used for IFD Authentication.
IDP-AS-RSA-PUB	2048- public part of RSA key pair used for Asymmetric Signature
IDP-AS-ECDSA-PUB	P-224, P-256, P-384, P-521 public part of ECDSA key pair used for Asymmetric signature
IDP-AC-RSA-PUB	2048/3072/4096 public part of the RSA key pair used for Asymmetric Cipher (key wrap, key unwrap)
IDP-ECDH-ECC-PUB	P-224, P-256, P-384, P-521 public part of the ECDH key pair used for shared key mechanism
IDP-KG-AS-RSA-PUB	2048/3072/4096- public part of the RSA generated key pair used for Asymmetric signature
IDP-KG-AS-ECDSA-PUB	P-224, P-256, P-384, P-521 public part of the ECDSA generated key pair used for Asymmetric signature
IDP-KG-AC-RSA-PUB	2048/3072/4096- public part of the RSA generated key pair used for Asymmetric cipher
IDP-KG-AC-ECDH-PUB	P-224, P-256, P-384, P-521 public part of the ECDSA generated key pair used for shared key mechanism
IDP-ECDSA-AUTH-ECC-PUB	P-224, P-256, P-384, P-521 public part of the ECDSA key pair used to Authenticate the card

Table 17 – IDPrime Applet Public Keys

4.4 USB MCU FW Critical Security Parameters

Key	Description / Usage
ID_FW_DOWNLOAD_RSA_KEY_PUBLIC_CORE	2048 bit RSA Public key embedded in the USB MCU FW – used by the FW to validate the new FW signature during FW Download. It concerns the Core part of the FW.
ID_FW_DOWNLOAD_RSA_KEY_PUBLIC_KERNEL	2048 bit RSA Public key embedded in the USB MCU FW – used by the FW to validate the new FW signature during FW Download. It concerns the kernel low & high parts of the FW.

Table 18 – USB MCU FW Public Keys

5 Roles, Authentication and Services

The Module supports Identity-based authentication.

Table 19 lists all operator roles supported by the Module. This Module does not support a maintenance role. The Module clears previous authentications on power cycle. The Module supports GP logical channels, allowing multiple concurrent operators. Authentication of each operator and their access to roles and services is as described in this section, independent of logical channel usage. 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. Authentication data is encrypted during entry (by SD-SDEK), is stored encrypted (by OS-MKDK) and is only accessible by authenticated services.

Role ID	Role Description
CO	(Cryptographic Officer) This role is responsible for card issuance and management of card data via the Card Manager applet. Authenticated using the SCP authentication method with SD-SENC.
IUSR	(User) The IDPrime User, authenticated by the IDPrime applet – see below for authentication mechanism.
ICAA	(Card Application Administrator) The IDPrime Card Application Administrator authenticated by the IDPrime applet – see below for authentication mechanism.
IO	Initialization Officer. This role is responsible for recycling/reinitializing the card using Reinit Authentication - see below for authentication mechanism.
UA	Unauthenticated role

Table 19 - Role Description

5.1 Secure Channel Protocol (SCP) Authentication (CO)

The Open Platform Secure Channel Protocol authentication method is performed when the EXTERNAL AUTHENTICATE service is invoked after successful execution of the INITIALIZE UPDATE command. These two commands operate as described next.

The SD-KENC and SD-KMAC keys are used along with other information to derive the SD-SENC and SD-SMAC keys, respectively. The SD-SENC key is used to create a cryptogram; the external entity participating in the mutual authentication also creates this cryptogram. Each participant compares the received cryptogram to the calculated cryptogram and if this succeeds, the two participants are mutually authenticated (the external entity is authenticated to the Module in the CO role).

For SCP03, AES-128, AES-192 or AES-256 keys are used for Global Platform secure channel operations, in which the Module derives session keys from the master keys and a handshake process, performs mutual authentication, and decrypts data for internal use only. The Module encrypts a total of one block (the mutual authentication cryptogram) over the life of the session encryption key; no decrypted data is output by the Module. AES key establishment provides a minimum of 128 bits of security strength. The Module uses the

SD-KDEK key to decrypt critical security parameters, and does not perform encryption with this key or output data decrypted with this key.

The strength of GP mutual authentication relies on AES key length, and the probability that a random attempt at authentication will succeed is:

- $\left(\frac{1}{2^{128}}\right)$ for AES 16-byte-long keys;
- $\left(\frac{1}{2^{192}}\right)$ for AES 24-byte-long keys;
- $\left(\frac{1}{2^{256}}\right)$ for AES 32-byte-long keys;

Based on the maximum count value of the failed authentication blocking mechanism, the minimum probability that a random attempt will succeed over a one minute period is 255/2^128.

5.2 IDPrime User Authentication (IUSR)

This authentication method compares a PIN value sent to the Module to the stored PIN values if the two values are equal, the operator is authenticated. This method is used in the IDPrime Applet services to authenticate to the IUSR role. There can be several OWNER PIN and one GlobalPIN. Both kind are User PINs.

The module enforces string length of 4 bytes minimum (16 bytes maximum for Global PIN / 64 bytes maximum for OWNER PIN).

For the User PIN, an embedded PIN Policy allows at least a combination of Numeric value ('30' to '39') or alphabetic upper case ('A' to 'Z') or alphabetic lower case ('a' to z'), so the possible combination of value for the User PIN is at minimum 62^4 ,greater than 10^7. Consequently the strength of this authentication method is as follow:

- The probability that a random attempt at authentication will succeed is lower than 1/10^6
- Based on a maximum count of 15 for consecutive failed service authentication attempts, the probability that a random attempt will succeed over a one minute period is lower than 15/10^7

5.3 IDPrime Card Application Administrator Authentication (ICAA)

The 3-Key Triple-DES key establishment provides 112 bits of security strength. The Module uses the IDP-SC-DES3 to authenticate the ICAA role.

- The probability that a random attempt at authentication will succeed is 1/2^64 (based on challenge size)
- Based on the maximum count value of the failed authentication blocking mechanism, the probability that a random attempt will succeed over a one minute period is 255/2^64

5.4 IDPrime Init Key Authentication (Initialization Officer Role)

The AES-256 key provides 256 bits of security strength. The Module uses the IDP-INITK-AES to authenticate the IO role.

- The probability that a random attempt at authentication will succeed is 1/2^256 (based on challenge size)
- Based on the maximum count value of the failed authentication blocking mechanism, the probability that a random attempt will succeed over a one minute period is 15/2^256

5.5 Platform Services

All services implemented by the platform, part of the Module are listed in the tables below. Each service description also describes all usage of CSPs by the service.

Service	Description
Context	Select an applet or manage logical channels.
Module Info (Unauth)	Read unprivileged data objects, e.g., module configuration or status information.
Module Reset	Power cycle or reset the Module. Includes Power-On Self-Test if self-test flag is set.
Run Cryptographic KATs	Resets a flag so that cryptographic KATs of the Card platform may be performed on demand via Module Reset.

Table 20 - Unauthenticated Services

Service	Description	CO
Lifecycle	Modify the card or applet life cycle status.	X
Manage Content	Load and install application packages and associated keys and data.	X
Module Info (Auth)	Read module configuration or status information (privileged data objects).	X
Secure Channel	Establish and use a secure communications channel.	X

Table 21 – Authenticated Card Manager Services

All of the above commands use the SD-SENC and SD-SMAC keys for secure channel communications, and SD-SMAC for firmware load integrity.

The card life cycle state determines which modes are available for the secure channel. In the SECURED card life cycle state, all command data must be secured by at least a MAC. As specified in the GP specification, there exist earlier states (before card issuance) in which a MAC might not be necessary to send Issuer Security Domain commands. Note that the LOAD service enforces MAC usage.

CSPs / Service	OS-DRBG-SEI	OS-DRBG-STATE	OS-GLOBALPIN	OS-MKDK	SD-KENC	SD-KMAC	SD-KDEK	SD-SENC	SD-SMAC	DAP-SYM	DAP-ASYM	DM-TOKEN-SYM	DM-RECEIPT-SYM	DM-TOKEN-ASYM
Module Reset	ZE W	ZE G W	--	--	--	--	--	Z	Z	--	--	--	--	--
Run Cryptographic KATs	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Module Info (Unauth)	--	--	--	--	--	--	--	E[2]	E[1]	--	--	--	--	--
Context	--	--	--	--	--	--	--	Z	Z	--	--	--	--	--
Secure Channel	--	EW	--	E	E	E	E	GE[1]	GE[1]	--	--	--	--	--
Manage Content	--	--	W	E	W	W E	W E	E[1]	E[1]	E W	E W	E W	E W	E W
Lifecycle	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
Module Info (Auth)	--	--	--	--	--	--	--	E[1]	E[1]	--	--	--	--	--

Table 22 – Platform CSP Access by Service

- G = Generate: The *Module* generates the CSP.
- R = Read: The *Module* reads the CSP (read access to the CSP by an outside entity).
- E = Execute: The *Module* executes using the CSP.
- W = Write: The *Module* writes the CSP. The write access is typically performed after a CSP is imported into the *Module* or when the module overwrites an existing CSP.
- Z = Zeroize: The *Module* zeroizes the CSP. For the Context service, SD session keys are destroyed on applet deselect (channel closure)
- -- = Not accessed by the service.

5.6 IDPRIME Services

All services implemented by the IDPrime applet are listed in the table below.

[2] "E" for Secure Channel keys is included for situations where a Secure Channel has been established and all traffic is received encrypted. The Secure Channel establishment includes authentication to the module.

Service	Description	ICAA	IUSR	UA	IO
EXTERNAL AUTHENTICATE	Authenticates the external terminal to the card. Sets the secure channel mode.	X	X	X	X
INTERNAL AUTHENTICATE	Authenticates the card to the terminal	X	X	X	X
SELECT	Selects a DF or an EF by its file ID, path or name (in the case of DFs).	X	X	X	X
CHANGE REFERENCE DATA	Changes the value of a PIN. (Note : User Auth is always done within the command itself by providing previous PIN) Secure Messaging is enforced for this command.	X	X		
RESET RETRY COUNTER	Unblocks and changes the value of a PIN Secure Messaging is enforced for this command.	X	X		
CREATE FILE	Creates an EF under the root or the currently selected DF or creates a DF under the root.	X	X		X
DELETE FILE	Deletes the current DF or EF.	X	X		X
DELETE ASYMMETRIC KEY PAIR	Deletes an RSA or ECDSA Asymmetric Key Pair	X	X		X
ERASE ASYMMETRIC KEY	Erases an RSA or ELC Asymmetric Key Pair	X	X		X
GET DATA (IDPrime Applet Specific)	Retrieves the following information: ■ CPLC data ■ Applet version ■ Software version (includes applet version - BER-TLV format) ■ Available EEPROM memory ■ Additional applet parameters ■ PIN Policy Error ■ Applet install parameter (DF0Ah tag)	X	X	X	X
GET DATA OBJECT	Retrieves the following information: ■ Public key elements ■ KICC ■ The contents of a specified SE	X	X	X	X

Service	Description	ICAA	IUSR	UA	IO
	■ Information about a specified PIN ■ Key generation flag ■ Touch Sense flag				
PUT DATA (IDPrime Applet Specific)	Creates or updates a data object ■ Create container[3] ■ Update public/private keys(3)		X		X
PUT DATA (IDPrime Applet Specific)	Creates or updates a data object ■ Access Conditions ■ Applet Parameters (Admin Key, Card Read Only and Admin Key Try Limit) ■ PIN Info	X			X
PUT DATA (IDPrime Applet Specific)	Creates or updates a data object ■ Update DES or AES Secret keys(3)	X	X		X
READ BINARY	Reads part of a binary file.	X	X	X	X
ERASE BINARY	Erases part of a binary file.	X	X		X
UPDATE BINARY	Updates part of a binary file.	X	X		X
GENERATE AUTHENTICATE	Used to generate secure messaging session keys between both entities (IFD and ICC) as part of elliptic curve asymmetric key mutual authentication.	X	X	X	X
GENERATE KEY PAIR	Generates an RSA or ECDSA key pair and stores both keys in the card. It returns the public part as its response.		X		X
PSO – VERIFY CERTIFICATE	Sends the IFD certificate C_CV.IFD.AUT used in asymmetric key mutual authentication to the card for verification. No real reason to use it in the personalization phase, but it is allowed.	X	X	X	X
PSO - HASH	Entirely or partially hashes data prior to a PSO–Compute Digital Signature command or prepares the data if hashed externally		X		X
PSO - DECIPHER	(RSA) Deciphers an encrypted message using a decipher key stored in the card.		X		X

[3] Secure Messaging in Confidentiality is mandatory

Service	Description	ICAA	IUSR	UA	IO
	(ECDSA) Generates a shared symmetric key. Secure Messaging is enforced for this command.				
PSO – COMPUTE DIGITAL SIGNATURE	Computes a digital signature.		X		X
PUT SECURE KEY	Secure Key Injection Scheme from Microsoft Minidriver spec V7		X		
UNAUTHENTICATE EXT	Breaks a secure messaging session, or invalidates an MS3DES3 External Authentication.	X	X	X	X
CHECK RESET AND APPLET SELECTION	Tells the terminal if the card has been reset or the applet has been reselected since the previous time that the command was performed.	X	X	X	X
GET CHALLENGE	Generates an 8, 16 or 32-byte random number.	X	X	X	X
MANAGE SECURITY ENVIRONMENT	Supports two functions, Restore and Set. ■ Restore: replaces the current SE by an SE stored in the card. ■ Set: sets or replaces one component of the current SE.	X	X	X	X
VERIFY	Authenticates the user to the card by presenting the User PIN. The User Authenticated status is granted with a successful PIN verification. Secure Messaging is enforced for this command.		X		
EXTERNAL AUTHENTICATION (ADMIN)	Performs external authentication for ADMIN role (using Triple-DES challenge response)	X			
REINIT (Authenticate)	Command used to grand the IO role using a challenge based AES256 authentication.				X
REINIT (Key Update)	Updates the Init Key used for IO role authentication and its ratification counter.				X
REINIT (Reinit)	Process the reinit command, actions depends on options (in any cases, erase of all user keys). During reinit process IO can process all the commands for which he has rights.				X
REINIT (End Reinit)	End the reinit process	X	X	X	X

Service	Description	ICAA	IUSR	UA	IO
REINIT (Get Counters)	Get ratification and retry counters for Init Key	X	X	X	X
PUT DATA (PIN)	Creates PIN objects on card (only possible if the PIN was not existing, or erased during reinit process)				X

Table 23 – IDPrime Applet Services and CSP Usage

All services implemented by the MSPNP applet are listed in the table below.

Service	Description	ICAA	IUSR	UA
GET DATA (MSPNP applet specific)	Retrieves the following information: ■ GUID			X

Table 24 – MSPNP applet Services

	CSP																
Service	IDP-SC-SMAC-AES	IDP-SC-SENC-AES	IDP-AS-RSA	IDP-AS-ECDSA	IDP-AC-RSA	IDP-ECDH-ECC	IDP-KG-AS-RSA	IDP-KG-AS-ECDSA	IDP-KG-AC-RSA	IDP-KG-AC-ECDH	IDP-ECDSA-AUTH-ECC	IDP-SC-DES3	IDP-SC-P-SKI-AES	IDP-SC-T-SKI-AES	IDP-SC-PIN-TDES	IDP-OWNERPIN / OS-GLOBALPIN	IDP-INITK-AES
EXTERNAL AUTHENTICATE	E	E	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
INTERNAL AUTHENTICATE	E	E	--	--	--	--	--	--	--	--	E	--	--	--	--	--	--
SELECT	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
CHANGE REFERENCE DATA	E	E	--	--	--	--	--	--	--	--	--	--	--	--	E	E W Z	--
RESET RETRY COUNTER	E	E	--	--	--	--	--	--	--	--	--	E	--	--	E	E W Z	--
CREATE FILE	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

	CSP																
Service	IDP-SC-SMAC-AES	IDP-SC-SENC-AES	IDP-AS-RSA	IDP-AS-ECDSA	IDP-AC-RSA	IDP-ECDH-ECC	IDP-KG-AS-RSA	IDP-KG-AS-ECDSA	IDP-KG-AC-RSA	IDP-KG-AC-ECDH	IDP-ECDSA-AUTH-ECC	IDP-SC-DES3	IDP-SC-P-SKI-AES	IDP-SC-T-SKI-AES	IDP-SC-PIN-TDES	IDP-OWNERPIN / OS-GLOBALPIN	IDP-INITK-AES
DELETE FILE	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
DELETE ASYMMETRIC KEY PAIR	--	--	Z	Z	Z	Z	Z	Z	Z	--	Z	--	--	--	--	--	--
ERASE ASYMMETRIC KEY	--	--	Z	Z	Z	Z	Z	Z	Z	--	Z	--	--	--	--	--	--
GET DATA (IDPrime MD Applet Specific)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
GET DATA OBJECT	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
PUT DATA (IDPrime MD Applet Specific)	E	E	WZ	WZ	WZ	WZ	WZ	WZ	WZ	--	WZ	--	--	--	--	--	--
PUT DATA (IDPrime MD Applet Specific)	--	--	--	--	--	--	--	--	--	--	--	WZ	--	--	--	--	--
READ BINARY	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
ERASE BINARY	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
UPDATE BINARY	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
GENERATE AUTHENTICATE	G	G	--	--	--	E	--	--	--	GE	--	--	--	--	--	--	--
GENERATE KEY PAIR	E	E	--	--	--	--	G	G	G	--	--	--	--	--	--	--	--
PSO – VERIFY CERTIFICATE	E	E	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
PSO - HASH	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
PSO – DECIPHER	--	--	--	--	E	--	--	--	E	--	--	--	--	--	--	--	--
PSO – COMPUTE DIGITAL SIGNATURE	--	--	E	E	--	--	E	E	--	--	--	--	--	--	--	--	--
PUT SECURE KEY	--	--	WZ	WZ	WZ	WZ	WZ	WZ	WZ	--	WZ	--	E	EWZ	--	--	--
UNAUTHENTICATE EXT	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
CHECK RESET AND APPLET SELECTION	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
GET CHALLENGE	E	E	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

	CSP																
Service	IDP-SC-SMAC-AES	IDP-SC-SENC-AES	IDP-AS-RSA	IDP-AS-ECDSA	IDP-AC-RSA	IDP-ECDH-ECC	IDP-KG-AS-RSA	IDP-KG-AS-ECDSA	IDP-KG-AC-RSA	IDP-KG-AC-ECDH	IDP-ECDSA-AUTH-ECC	IDP-SC-DES3	IDP-SC-P-SKI-AES	IDP-SC-T-SKI-AES	IDP-SC-PIN-TDES	IDP-OWNERPIN / OS-GLOBALPIN	IDP-INITK-AES
MANAGE SECURITY ENVIRONMENT	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
VERIFY	E	E	--	--	--	--	--	--	--	--	--	--	--	--	--	E	--
EXTERNAL AUTHENTICATION (ADMIN)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	E	--
REINIT (Authenticate)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	E
REINIT (Key Update)	E	E	--	--	--	--	--	--	--	--	--	--	--	--	--	--	WZ
REINIT (Reinit)	E	E	Z	Z	Z	--	Z	Z	Z	--	--	--	WZ	--	--	--	--
REINIT (End Reinit)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
REINIT (Get Counters)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
PUT DATA (PIN)	E	E	--	--	--	--	--	--	--	--	--	--	--	--	E	WZ	--

Table 25 – IDPrime CSP Access by Service

5.7 USB MCU Services

All services implemented by the USB MCU FW are listed in the table below.

Service	Description	UA
USB SC	This module provides framework for the USB Standard and Class requests, CCID protocols, to allow ISO7816 communication with the SC.	X
USB MCU	This module provides framework for the USB commands which are directed to the FW such as FW get Info, etc.	X
FW Update	This module defines an interface for firmware update process; FW is protected by an RSA 2048 PKCS#1 v1.5 SHA256 signatures. The signature verification is for the purposes of authenticating the USB FW download. No	x

Service	Description	UA
	Thales signed FWs will be rejected by the USB MCU	

Table 26 – USB MCU FW Service

CSP		
Service	ID_FW_DOWNLOAD_RSA_KEY_PUBLIC_CORE	ID_FW_DOWNLOAD_RSA_KEY_PUBLIC_KERNEL
USB SC	--	--
USB MCU	--	--
FW Update	E	E

Table 27 – USB MCU FW CSP Access by Service

6 Physical Security Policy

'eToken 5110+ FIPS' is a multiple-Chip standalone cryptographic module, 2 ICs are mounted on a PCB assembly with a connector and passive components, covered by epoxy on both sides, exposing only the LED and USB connector. The Module is intended to be covered within a plastic enclosure. Physical inspection inside the Module boundary is not practical, as the epoxy layer is opaque.

The LED functions as status indicator and this is the reason it is kept non-covered with epoxy. The LED has no connection to Critical Security Parameters, and thus cannot output any sensitive information.

7 Operational Environment

This section does not apply to CM. No code modifying the behavior of the CM operating system can be added after its manufacturing process.

Only authorized applets can be loaded at post-issuance under control of the Cryptographic Officer. Their execution is controlled by the CM operating system following its security policy rules.

Only authorized (Signed by Thales private keys) USB MCU FW can be loaded at post-issuance while none Thales signed FWs will be rejected by the USB MCU.

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 a separate FIPS 140-2 validation.

8 Electromagnetic Interference and Compatibility (EMI/EMC)

The Module conforms to the EMI/EMC requirements specified by part 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, Digital Devices, Class B.

9 Self-test

9.1 USB MCU Self-test

The LED is used to indicate token activity or FW/HW failure.

In particular, the LED blinks once if USB MCU integrity test fails, twice if USB MCU SHA-256/RSA tests fail. These checks are performed systematically at the boot of the token.

9.2 Card Self-test

The card or platform tests result has no impact on the LED activity.

9.3 Power-on Self-test

On power-on or reset, the Module performs self-tests described in table below. All KATs must be completed successfully prior to any other use of cryptography by the *Module*. If one of the KATs fails, the *Module* enters the *Card Is Mute* error state or *Card is Killed* error state, depending on number of failures.

Test Target	Description
Firmware Integrity (USB MCU)	32 bit CRC performed over all code located in USB MCU Flash memory.
Firmware Integrity (card)	16 bit CRC performed over all code located in FLASH and EEPROM memory (for OS, Applets).
AES (card)	Performs decrypt KAT using an AES 128-bit key in ECB mode. AES encrypt is self-tested as an embedded algorithm of AES-CMAC.
DRBG (card)	Performs DRBG SP 800-90A Section 11.3 instantiate and generate health test KAT with fixed inputs (derivation function and no reseeding supported).
KAS SSC ECC (card)	Performs a KAS SSC ECC KAT using an ECC P-224 key.
ECDSA (card)	Performs separate ECDSA signature and verification KATs using an ECC P-224 key.
KBKDF AES-CMAC (card)	Performs a KDF AES-CMAC KAT using an AES 128 key and 32-byte derivation data. The KAT computes session keys and verifies the result. Note that KDF KAT is identical to an AES-CMAC KAT; the only difference is the size of input data.
RSA (card)	Performs separate RSA PKCS#1 v1.5 signature and verification KATs using an RSA 2048 bit key, and a RSA PKCS#1 v1.5 signature KAT using the RSA CRT implementation with a 2048 bit key. RSA CRT signature verification is tested as part of the RSA signature verification KAT as described above. RSA PKCS#1 v1.5 decryption KAT with a 2048 bit key is also performed.
SHA-1, SHA-2 (card)	Performs separate KATs for SHA-1 and SHA-512.

Test Target	Description
Triple-DES (card)	Performs separate encrypt and decrypt KATs using 3-Key TDEA in ECB mode.
KAS KDF (card)	Performs a KAS KDF KAT using "One-step key derivation" scheme as in SP800-56C rev2. The KAT uses a 4-byte counter, a 32-byte shared secret and 1-byte fixedInfo, and computes a SHA-256 of the input data, then compared the result with the expected one.
SHA-256 (USB MCU)	Performs a SHA-256 on a constant message stored in memory. Result given by the operation is compared with the expected result. If they are the same, self-test is OK.
RSA SIGN Verification (USB MCU)	Performs a RSA PKCS#1 signature verification - 2048 bit key on a constant key (modulus & exponent) and message and the expected result stored in memory. If the signature verification is successful, self-test is OK.

Table 28 – Power-On Self-Test

9.4 Conditional Self-tests

On every call to the [SP 800-90A] DRBG, the Module performs the FIPS 140-2 Continuous RNG test (CRNGT) to assure that the output is different than the previous value. Note that the DRBG is seeded only once per power cycle and therefore a CRNGT is not required to be performed on the NDRNG per IG 9.8.

When any asymmetric key pair is generated (for RSA or ECC keys) the Module performs a pairwise consistency test.

When new firmware is loaded into the Module using the Manage content service, the CO verifies the integrity and authenticity of the new firmware (applet) using the SD-SMAC key for MAC process.

Optionally, the CO may also verify a MAC or a signature of the new firmware (applet) using the DAP-SYM key or DAP-ASYM key respectively. The signature or MAC block in this scenario is generated by an external entity using the key corresponding to the asymmetric key DAP-ASYM or the secret key DAP-SYM.

The Module also performs the required assurances from [SP800-56A-rev3] (public Key Validation).

When a new FW is downloaded to the USB MCU, the existing MCU FW validates the integrity of the new FW by verifying the new FW signatures using 2048 bit RSA Public keys embedded in the USB MCU existing FW.

Number of blinks	Error meaning	Self-tests coverage
1	Code integrity check of the firmware failure: The check is done systematically at the boot of the token	Yes
2	Cryptographic self-test has failure The check is done systematically at the boot of the token	Yes

3	Non-register interrupt has occurred Example: try to write in the Config area (CFG) without having unlock the flash of the microcontroller	Not Covered by Security Policy
4	CPU FLASH read protection is not set to **JTAG**	Not Covered by Security Policy
5	CCID OUT or Control transfer buffer payload is more than internal allocated buffer length	Not Covered by Security Policy

Table 29 – Self-Tests output mechanism

This table reports how FW interacts with the User depending on one of the Self-tests reporting an error.

9.5 Reducing the number of Known Answer Tests (Card only)

The card (part of the CM) implements latest IG **r**educing the number of Known Answer tests (KAT) described at chapter 9.11.

On the 1st reset of the card, it performs "Firmware Integrity" test and all Cryptographic KATs.

On each next reset of the card, it performs only "Firmware Integrity test" as permitted by IG document.

The cryptographic KATs are also available on demand and can be played by any operator with the Run Cryptographic KATs service (see Section 5.5– Platform Services).

10 Design Assurance

The Module meets the Level 3 Design Assurance section requirements.

10.1 Configuration Management

An additional document (Configuration Management Plan document) defines the methods, mechanisms and tools that allow to identify and place under control all the data and information concerning the specification, design, implementation, generation, test and validation of the card software throughout the development and validation cycle.

10.2 Delivery and Operation

Some additional documents ('Delivery and Operation', 'Reference Manual', 'Card Initialization Specification' documents) define and describe the steps necessary to deliver and operate the Module securely.

10.3 Guidance Documents

The Guidance document provided with Module is intended to be the 'Reference Manual'. This document includes guidance for secure operation of the Module by its users as defined in the section: Roles, Authentication and Services.

10.4 Language Level

The Module operational environment is implemented using a high level language. A limited number of software modules have been written in assembler to optimize speed or size.

The IDPrime Applet is a Java applet designed for the Java Card environment.

11 Mitigation of Other Attacks Policy

The Module implements defenses against:

- Fault attacks
- Side channel analysis (Timing Analysis, SPA/DPA, Simple/Differential Electromagnetic Analysis)
- Probing attacks
- Card tearing

The following table outlines the protection mechanisms used to mitigate each of the attacks:

		MITIGATED ATTACKS			
		A1 - Fault attacks	A2 - Side channel analysis	A3 - Probing attacks	A4 - Card tearing
COUNTER MEASURES	C1 - Sensor activation	X	X	X	
	C2 - Resident countermeasures	X		X	
	C3 - Memory ciphering	X	X	X	
	C4 - BUS protection system	X		X	
	C5 - Secret data masking	X	X	X	
	C6 - Secret data cleaning	X		X	
	C7 - Sensitive data and processes management	X	X		X
	C8 - Dummy code execution		X		
	C9 - Redundancy techniques	X			
	C10 - Data integrity check	X			X
	C11 - Life cycle management	X			
	C12 – Fault detection counter	X			

Table 30 – Cross mapping table between mitigated attacks and counter measures

12 Security Rules and Guidance

The Module implementation also enforces the following security rules:

- No additional interface or service is implemented by the Module which would provide access to CSPs.
- Data output is inhibited during key generation, self-tests, zeroization, and error states.
- There are no restrictions on which keys or CSPs are zeroized by the zeroization service.
- The Module does not support manual key entry, output plaintext CSPs or output intermediate key values.
- Status information does not contain CSPs or sensitive data that if misused could lead to a compromise of the Module.

At the time the card is issued, **IDPrime Applet** shall be personalized with the appropriate data in order to be initialized into the Approved mode. Personalization includes IDPrime keys and PIN values, as listed below:

- IDP-AS-RSA: RSA key pair used for Asymmetric Signature
- IDP-AS-ECDSA: ECDSA key pair used for Asymmetric signature
- IDP-AC-RSA: RSA key pair used for Asymmetric Cipher (key wrap, key unwrap)
- IDP-ECDH-ECC: ECDH key pair used for shared key mechanism
- IDP-ECDSA-AUTH-ECC: ECDSA key used to Authenticate the card
- IDP-SC-DES3: 3-Key Triple-DES key used for authentication.
- IDP-SC-P-SKI-AES: AES session key used for Secure Key Injection
- IDP-OWNERPIN: PIN value managed by the Applet.
- IDP-INITK-AES: AES key used to authenticate IO Role

END OF DOCUMENT