MultiApp ID V2.1 Platform
FIPS 140-2 Cryptographic Module Security Policy

Table of Contents

Table of Tables

Table of Figures

References

The following standards are referred to in this Security Policy.

Acronym	Full Specification Name
[FIPS140-2]	NIST, *Security Requirements for Cryptographic Modules*, May 25, 2001
[GlobalPlatform]	*GlobalPlatform Consortium: GlobalPlatform Card Specification 2.1.1,* March 2003, http://www.globalplatform.org *GlobalPlatform Consortium: GlobalPlatform Card Specification 2.1.1* Amendment A, March 2004
[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*
[JavaCard]	*Java Card 2.2.2 Runtime Environment (JCRE) Specification* *Java Card 2.2.2 Virtual Machine (JCVM) Specification* *Java Card 2.2.2 Application Programming Interface* Published by Sun Microsystems, March 2006
[SP800-131A]	*Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths*, January 2011
[ANS X9.31]	American Bankers Association, *Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA)*, ANSI X9.31-1998 - Appendix A.2.4.
[SP 800-67]	NIST Special Publication 800-67, *Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher*, version 1.2, July 2011
[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 186-2]	NIST, *Digital Signature Standard (DSS)*, FIPS Publication 186-2, January, 2000 with Change Notice 1. (DSA, RSA and ECDSA)
[SP 800-56A]	NIST Special Publication 800-56A, *Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography*, March 2007
[FIPS 180-3]	NIST, *Secure Hash Standard*, FIPS Publication 180-3, October 2008
[AESKeyWrap]	NIST, *AES Key Wrap Specification*, 16 November 2001. This document defines symmetric key wrapping, Use of 2-Key TDEA in lieue of AES is described in [IG] D.2.
[IG]	NIST, *Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program,* last updated 15 July 2011.

Table 1 – References

Acronyms and definitions

Acronym	Definition
GP	Global Platform
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 Gemalto MultiApp ID V2.1 cryptographic module, hereafter denoted ***the Module***. The Module, validated to FIPS 140-2 overall Level 3, is a "contact-only" secure controller implementing the Global Platform operational environment, with Card Manager and a Demonstration Applet. The Demonstration Applet is available only to demonstrate the complete cryptographic capabilities of the Module for FIPS 140-2 validation, and is not intended for general use.

The Module is designated as a limited operational environment under the FIPS 140-2 definitions. The Module includes a firmware load service to support necessary updates. The loading of non-validated firmware within the validated cryptographic module invalidates the module's validation; 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 FIPS 140-2 security levels for the Module are as follows:

Security Requirement	Security Level
Cryptographic Module Specification	3
Cryptographic Module Ports and Interfaces	3
Roles, Services, and Authentication	3
Finite State Model	3
Physical Security	4
Operational Environment	N/A
Cryptographic Key Management	3
EMI/EMC	3
Self-Tests	3
Design Assurance	3
Mitigation of Other Attacks	3

Table 3 – Security Level of Security Requirements

The Module implementation is compliant with:

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

1.1 Hardware and Physical Cryptographic Boundary

The Module is designed to be embedded into plastic card bodies, with a contact plate connection. The physical form of the Module is depicted in Figure 1 (to scale); the red outline depicts the physical cryptographic boundary, representing the surface of the chip and the bond pads. The cross-hatching indicates the presence of the hard, opaque outer layer shielding. In production use, the Module is wire-bonded to a frame connected to a contact plate, enclosed in epoxy and mounted in a card body. The Module relies on [ISO7816] card readers as input/output devices.

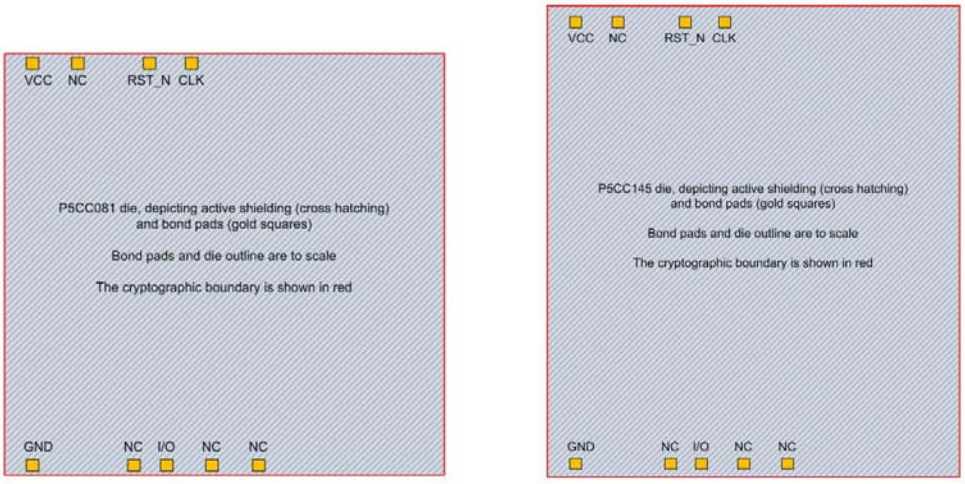

Figure 1 –Physical Form and Cryptographic Boundary (P5CC081 left; P5CC145 right)

Pad	Description	Logical interface type
VCC, GND	ISO 7816: Power and ground	Power
CLK	ISO 7816: Clock	Control in
RST	ISO 7816: Reset	Control in
I/O	ISO 7816: Serial interface	Data in, data out, control in, status out
NC	No connect	Not used

Table 4 – Ports and Interfaces

1.2 Firmware and Logical Cryptographic Boundary

Figure 2 depicts the Module operational environment and applets.

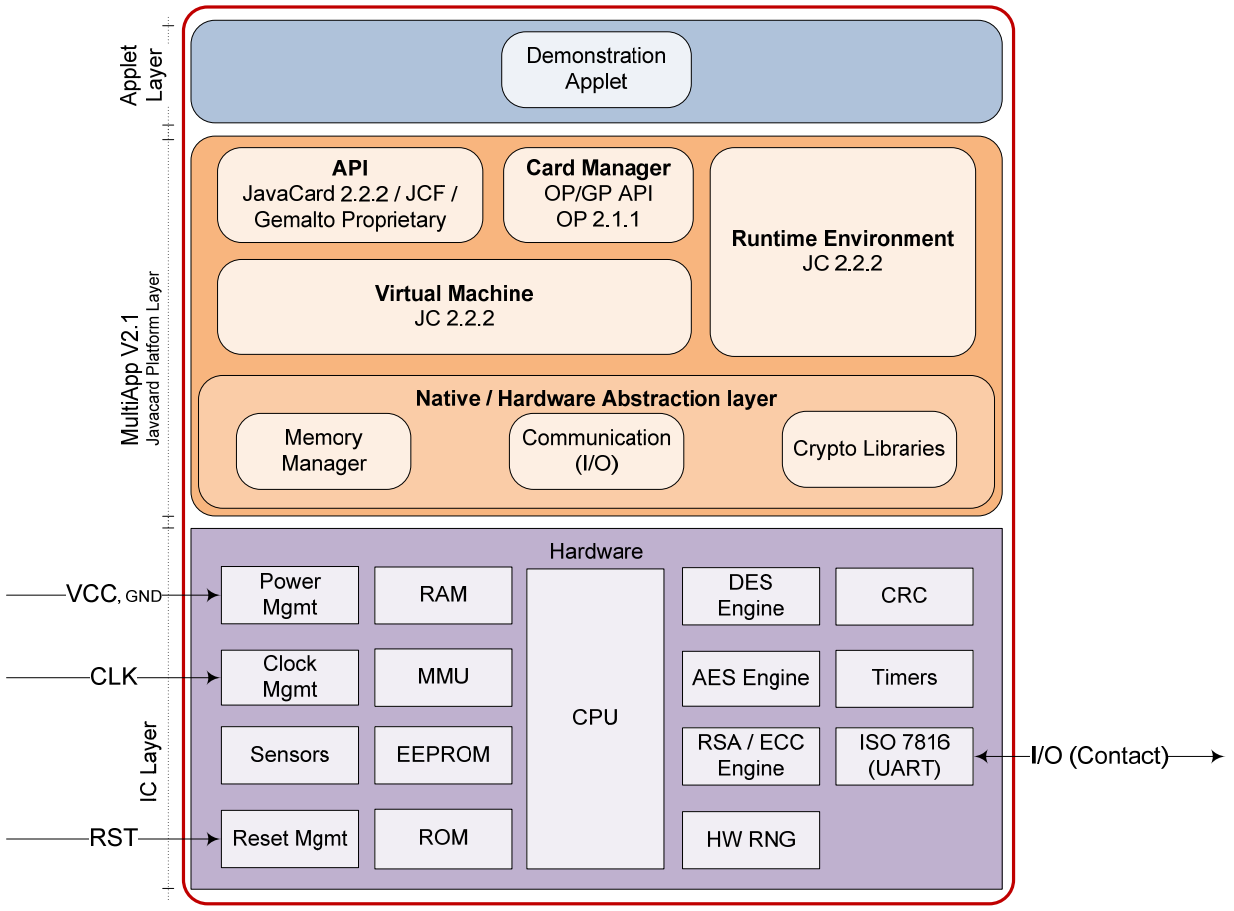

Figure 2 - Module Block Diagram

The Module supports [ISO7816] T=0 and T=1 communications protocol variations and two memory configurations:

- M1 (utilizing the NXP P5CC081): 80 Kbyte EEPROM, 264 Kbyte ROM
- M2 (utilizing the NXP P5CC145): 144 Kbyte EEPROM, 264 Kbyte ROM

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 bytecode interpreter, firewall, exception management and bytecode 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, PSS, T=0 and T=1 protocols.

The *Cryptography Libraries* implement the algorithms listed in section 2.

1.3 Versions and mode of operation

Hardware: P5CC081, P5CC145
Firmware: MultiApp ID V2.1, patch V2.2 (for P5CC081 implementation) and V2.4 (for P5CC145 implementation), Demonstration Applet version V1.1

The Module is always in the approved mode of operation. To verify that a module 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 Module responds with the following information:

MPH117 Mask - CPLC data (tag 9F7F)			
Byte	**Description**	**Value**	**Value meaning**
1-2	IC fabricator	**4790h**	NXP
3-4	IC type	**5081h**	P5CC081
5-6	Operating system identifier	**1291h**	Gemalto
7-8	Operating system release date (YDDD) – Y=Year, DDD=Day in the year	**1102h**	April 12th 2011
9-10	Operating system release level	**0201h**	V2.1
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

MPH119 Mask - CPLC data (tag 9F7F)			
Byte	**Description**	**Value**	**Value meaning**
0-1	IC fabricator	**4790h**	NXP
2-3	IC type	**5145h**	P5CC145
4-5	Operating system identifier	**1291h**	Gemalto
6-7	Operating system release date (YDDD) – Y=Year, DDD=Day in the year	**1157h**	June 6th 2011
8-9	Operating system release level	**0201h**	V2.1
10-11	IC fabrication date	xxxxh	Filled in during IC manufacturing
12-15	IC serial number	xxxxxxxxh	Filled in during IC manufacturing
16-17	IC batch identifier	xxxxh	Filled in during IC manufacturing
18-19	IC module fabricator	xxxxh	Filled in during module manufacturing
20-21	IC module packaging date	xxxxh	Filled in during module manufacturing
22-23	ICC manufacturer	xxxxh	Filled in during module embedding
24-25	IC embedding date	xxxxh	Filled in during module embedding

26-27	IC pre-personalizer	xxxxh	Filled in during smartcard preperso
28-29	IC pre-personalization date	xxxxh	Filled in during smartcard preperso
30-33	IC pre-personalization equipment identifier	xxxxxxxxh	Filled in during smartcard preperso
34-35	IC personalizer	xxxxh	Filled in during smartcard personalization
36-37	IC personalization date	xxxxh	Filled in during smartcard personalization
38-41	IC personalization equipment identifier	xxxxxxxxh	Filled in during smartcard personalization

MPH117 Mask - Identification data (tag 0103)			
Byte	**Description**	**Value**	**Value meaning**
0	Gemalto Family Name	**B0h**	Javacard
1	Gemalto OS Name	**85h**	MultiApp ID
2	Gemalto Mask Number	**36h**	MPH117
3	Gemalto Product Name	**38h**	Generic MPH117 product
4	Gemalto Flow Version	**08h**	▪ Major nibble: flow version = 0h ▪ Minor nibble: conformity to the security certificates *(1b in case of conformity claim, otherwise 0b)* b3 (leftmost bit): conformity to FIPS certificate = 1b b2: conformity to PPSUN certificate = 0b b1: conformity to IAS Classic V3 PPSSCD certificate = 0b b0: conformity to IAS XL PPSSCD certificate = 0b
5	Gemalto Filter Set	**22h**	▪ Major nibble: filter family = 2h ▪ Lower nibble: version of the filter = 2h
6-7	Chip Manufacturer	**4790h**	NXP
8-9	Chip Version	**5081h**	P5CC081
10-11	RFU	**0000h**	-
12-17	RFU	**xx..xxh**	-
18-28	RFU	**xx..xxh**	-

MPH119 Mask - Identification data (tag 0103)			
Byte	**Description**	**Value**	**Value meaning**
0	Gemalto Family Name	**B0h**	Javacard
1	Gemalto OS Name	**85h**	MultiApp ID
2	Gemalto Mask Number	**37h**	MPH119
3	Gemalto Product Name	**39h**	Generic MPH119 product
4	Gemalto Flow Version	**08h**	▪ Major nibble: flow version = 0h ▪ Minor nibble: conformity to the security certificates *(1b in case of conformity claim, otherwise 0b)* b3 (leftmost bit): conformity to FIPS certificate = 1b b2: conformity to PPSUN certificate = 0b b1: conformity to IAS Classic V3 PPSSCD certificate = 0b b0: conformity to IAS XL PPSSCD certificate = 0b
5	Gemalto Filter Set	**24h**	▪ Major nibble: filter family = 2h ▪ Lower nibble: version of the filter = 4h
6-7	Chip Manufacturer	**4790h**	NXP
8-9	Chip Version	**5145h**	P5CC145
10-11	RFU	**0000h**	-
12-17	RFU	**xx..xxh**	-
18-28	RFU	**xx..xxh**	-

Table 5 –Versions and Mode of Operations Indicators

The Demonstration Applet AID (application identifier) value is 464950535F546573744170706C657401. It can be retrieved using the GET STATUS command - available after a successful Card Manager authentication – which provides the AIDs of all the packages loaded in the card.

Field	**CLA**	**INS**	**P1-P2 (Tag)**	**Lc-Le**	**Purpose**
Value	**80**	**F2**	**20-00**	**02-00**	Get AID list – first command
Value	**80**	**F2**	**20-01**	**02-00**	Get AID list, continued (to get the end of the list, if previous command returned '6310 SW)

2 Cryptographic functionality

The Module operating system implements the FIPS Approved and Non-FIPS Approved but Allowed cryptographic function listed in Tables 6 and 7 below.

Algorithm	Description	Cert #
RNG	[ANS X9.31] Random number generator	1023
Triple-DES	[SP 800-67] Triple Data Encryption Algorithm. The Module supports the 2-Key and 3-Key options; CBC and ECB modes. Note that the Module does not support a mechanism that would allow collection of plaintext / ciphertext pairs aside from authentication, limited in use by a counter.	1264
Triple-DES MAC	[FIPS113] TDEA Message Authentication Code. Vendor affirmed, based on validated TDEA.	1264
AES	[FIPS 197] Advanced Encryption Standard algorithm. The Module supports 128-, 192- and 256-bit key lengths with ECB and CBC modes.	1943
RSA	[PKCS#1] RSA signature generation, verification, and key pair generation. The Module follows PKCS#1 and supports 1024 to 2048-bit RSA keys (by step of 32 bits).	1006
RSA CRT	[PKCS#1] RSA signature generation, verification, CRT key pair generation. The Module follows PKCS#1 and supports 1024 to 2048-bit RSA keys (by step of 32 bits).	1010
ECDSA	[FIPS 186-2] Elliptic Curve Digital Signature Algorithm. The Module supports the NIST defined P-192, P-224, P-256, P-384 and P-521 curves.	280
ECC CDH	[SP 800-56A] The Section 5.7.1.2 ECC CDH Primitive. The Module supports the NIST defined P-192, P-224, P-256, P-384 and P-521 curves.	17
SHA-1, SHA-224, SHA-256	[FIPS 180-3] Secure Hash Standard compliant one-way (hash) algorithms.	1706
SHA-384, SHA-512	[FIPS 180-3] Secure Hash Standard compliant one-way (hash) algorithms.	1707

Table 6 – FIPS Approved Cryptographic Functions

Algorithm	Description
Symmetric Key Wrap	[AESKeyWrap] The Module supports symmetric key unwrapping using 2-Key TDEA. This key establishment methodology provides 112 bits of encryption strength.
EC DH	Non-SP 800-56A EC DH. The Module supports the NIST defined P-192, P-224, P-256, P-384 and P-521 curves.

Table 7 – Non-FIPS Approved But Allowed Cryptographic Functions

2.1 Platform Critical Security Parameters

All CSPs used by the Module are described in this section. All usage of these CSPs by the Module, including all CSP lifecycle states, are described in the services detailed in Section 4.

Key	Description / Usage
OS-SEED-KEY	2-Key TDEA random loaded into the card during pre-personalization of the card used as a seed key for the [ANS X9.31]
OS-RNG-STATE	64 bit random value and 64 bit counter value used in the [ANS X9.31] RNG implementation. Proprietary information describes seeding and persistence of the RNG state.
OS-GLOBALPIN	Global PIN value managed by the ISD.
OS-MKDK	2-Key TDEA key used to encrypt OS-GLOBALPIN value
ISD-KENC	2-Key TDEA Master key used by the CO role to generate ISD-SENC
ISD-KMAC	2-Key TDEA Master key used by the CO role to generate ISD-SMAC
ISD-KDEK	2-Key TDEA Sensitive data decryption key used by the CO role to decrypt CSPs for SCP01, and used to generate ISD-SDEK in case of SCP02.
ISD-SENC	2-Key TDEA Session encryption key used by the CO role to encrypt / decrypt secure channel data.
ISD-SMAC	2-Key TDEA Session MAC key used by the CO role to verify inbound secure channel data integrity and authenticity.
ISD-SDEK	2-Key TDEA Session DEK key used by the CO role to decrypt CSPs for SCP02.
DAP-DES	An optional 2-Key Triple-DES key used to verify integrity and authenticity of packages loaded into the module.

Table 8 - Platform Critical Security Parameters

2.2 Platform Public key

Key	Description / Usage
DAP-SVK	RSA 1024 Global Platform Data Authentication Public Key. Optionally used to verify the signature of packages loaded into the Module.

Table 9 – Platform Public Keys

2.3 Demonstration Applet Critical Security Parameters

Key	Description / Usage
DSC-AES	AES 128/192/256 key used by Demonstrate Symmetric Cipher
DSC-TDEA	3-Key TDEA key used by Demonstrate Symmetric Cipher
DSS-TDEA	3-Key TDEA key used by Demonstrate Symmetric Signature (MAC generation and verify)
DAS-RSA	1024-, 1536-, 2048- RSA private key used by Demonstrate Asymmetric Signature (signature generation and verify)
DAS-ECDSA	P-192, P-224, P-256, P-384, P-521 ECDSA private key used by Demonstrate Asymmetric Signature (signature generation and verify)
ECDH-ECC	P-192, P-224, P-256, P-384, P-521 ECDSA private key used by Demonstrate ECC CDH (shared secret primitive)
DKG-RSA	1024-, 1536-, 2048- RSA private key generated by Demonstrate Asymmetric Key Generation
DKG-ECDSA	P-192, P-224, P-256, P-384, P-521 ECDSA private key generated by Demonstrate Asymmetric Key Generation
DMK	Demonstration master key, 3-Key TDEA key used to encrypt or decrypt keys exported out of or imported into the module for use by the demonstration applet.

Table 10 – Demonstration Applet Critical Security Parameters

2.4 Demonstration Applet Public Keys

Key	Description / Usage
DAS-RSA-SVK	1024-, 1536-, 2048- RSA public key used by Demonstrate Asymmetric Signature (signature generation and verify)
DAS-ECDSA-SVK	P-192, P-224, P-256, P-384, P-521 ECDSA public key used by Demonstrate Asymmetric Signature (signature generation and verify)
DKG-RSA-PUB	1024-, 1536-, 2048- RSA public key generated by Demonstrate Asymmetric Key Generation
DKG-ECDSA-PUB	P-192, P-224, P-256, P-384, P-521 ECDSA public key generated by Demonstrate Asymmetric Key Generation

Table 11 – Demonstration Applet Public Keys

3 Roles, authentication and services

Table 12 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 deselection (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 ISD-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 ISD-SENC.
USR	(User) This role has the privilege to use the cryptographic services provided by the demonstration applet. Authenticated using the GLOBAL PIN verification.

Table 12 - Roles description

3.1 Secure Channel Protocol (SCP) Authentication

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 ISD-KENC and ISD-KMAC keys are used along with other information to derive the ISD-SENC and ISD-SMAC keys, respectively. The ISD-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 CM role).

[SP 800-131A] Section A.1 provides the NIST rationale for 2-Key TDEA security strength. 2-Key TDEA is 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. The Module claims 112-bit security strength for its 2-Key TDEA operations, as the meet-in-the-middle attack rationale described in [SP 800-131A] does not apply unless the attacker has access to encrypt/decrypt pairs.

2-Key TDEA key establishment provides 112 bits of security strength. The Module uses the ISD-KDEK key to decrypt critical security parameters, and does not perform encryption with this key or output data decrypted with this key.

- The probability that a random attempt at authentication will succeed is 1/2^64 (the block size), meeting the FIPS 140-2 one in 1,000,000 requirement.
- 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 255/2^112, meeting the FIPS 140-2 one in 100,000 requirement.

3.2 USR role authentication

This authentication method compares a PIN value sent to the Module to the stored OS-GLOBALPIN values if the two values are equal, the operator is authenticated. This method is used in the Demonstration Applet services to authenticate to the USR role.

The Module enforces a minimum character length of 6 characters, allowing all characters, so the strength of this authentication method is as follows:

- The probability that a random attempt at authentication will succeed is 1/256^6, meeting the FIPS 140-2 one in 1,000,000 requirement.
- Based on a maximum count of 15 for failed service authentication attempts, the probability that a random attempt will succeed over a one minute period is 15/256^6, meeting the FIPS 140-2 one in 100,000 requirement.

3.3 Services

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

The ISD-SENC and ISD-SMAC keys are used by every Card Manager service when a secure channel has been established, for decryption and MAC verification (packet integrity and authenticity), respectively. This is noted below as "Optionally uses ISD-SENC, ISD-SMAC (SCP)". Unauthenticated commands listed below function whether or not a secure channel has been established.

Service	Description
Card Reset (Self-test)	Power cycle the Module by removing and reinserting it into the contact reader slot, or by reader assertion of the RST signal. The *Card Reset* service will invoke the power on self-tests described in Section 4. Moreover, on any card reset, the Module overwrites with zeros the RAM copy of, OS-RNG-STATE, ISD-SENC, ISD-SMAC and ISD-SDEK. The Module can also write the values of all CSPs stored in EEPROM as a consequence of restoring values in the event of card tearing or a similar event.
EXTERNAL AUTHENTICATE	Authenticates the operator and establishes a secure channel. Must be preceded by a successful INITIALIZE UPDATE. Uses ISD-SENC and ISD-SMAC.
INITIALIZE UPDATE	Initializes the Secure Channel; to be followed by EXTERNAL AUTHENTICATE. Uses the ISD-KENC, ISD-KMAC and ISD-KDEK master keys to generate the ISD-SENC, ISD-SMAC and ISD-SDEK session keys, respectively.
GET DATA	Retrieve a single data object. Optionally uses ISD-SENC, ISD-SMAC (SCP).
MANAGE CHANNEL	Open and close supplementary logical channels. Optionally uses ISD-SENC, ISD-SMAC (SCP).
SELECT	Select an applet. Optionally uses ISD-SENC, ISD-SMAC (SCP).

Table 13 - Unauthenticated Services and CSP Usage

Receipt of the first command generates the RAM copy of OS-RNG-STATE and updates the EEPROM copy of OS-RNG-STATE.

Service	Description	CO
DELETE	Delete an applet from EEPROM. Uses ISD-SENC, ISD-SMAC (SCP). This service is provided for use when an applet is loaded on the card, and does not impact platform CSPs	X
GET STATUS	Retrieve information about the card. Optionally uses ISD-SENC, ISD-SMAC (SCP).	X
INSTALL	Perform Card Content management. Optionally uses ISD-SENC, ISD-SMAC (SCP). Optionally, the Module uses the DAP key (either DAP-DES or DAP-SVK) to verify the package signature.	X
LOAD	Load a load file (e.g. an applet). Optionally uses ISD-SENC, ISD-SMAC (SCP).	X
PUT DATA	Transfer data to an application during command processing. Optionally uses ISD-SENC, ISD-SMAC (SCP).	X
PUT KEY	Load Card Manager keys The Module uses the ISD-SDEK session key to decrypt the keys to be loaded. Optionally uses ISD-SENC, ISD-SMAC (SCP).	X
SET STATUS	Modify the card or applet life cycle status. Optionally uses ISD-SENC, ISD-SMAC (SCP).	X
STORE DATA	Transfer data to an application or the security domain (card manager) processing the command. Optionally, updates OS-GLOBALPIN. Optionally uses ISD-SENC, ISD-SMAC (SCP).	X
GET MEMORY SPACE	Monitor the memory space available on the card. Optionally uses ISD-SENC, ISD-SMAC (SCP).	X
SET ATR	Change the card ATR. Optionally uses ISD-SENC, ISD-SMAC (SCP).	X

Table 14 – Authenticated Card Manager Services and CSP Usage

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.

Service	Description	USR
Demonstrate RNG	Generates a random value. Does not use CSPs.	X
Demonstrate Hash	Hashes a provided value using SHA-1, SHA-224, SHA-256, SHA-384, SHA-512. Does not use CSPs.	X
Demonstrate Symmetric Cipher	Encrypts or decrypts a provided value using DSC-AES or DSC-TDEA provided in encrypted form with the service.	X
Demonstrate Symmetric Signature	Generates or verifies a TDEA MAC using DSS-TDEA provided in encrypted form during service invocation.	X
Demonstrate Asymmetric Signature	Generates or verifies a signature using DAS-RSA or DAS-ECDSA provided to the module in encrypted form during service invocation.	X
Demonstrate EC DH	Generates a shared secret value in accordance with SP 800-56A Section 5.7.1.2, and as well with non-SP 800-56A EC DH, using ECDH-ECC.	X
Demonstrate Asymmetric Key Generation	Demonstrates RSA, RSA CRT and ECC key generation, generating DKG-RSA and DKG-ECDSA.	X

Table 15 – Demonstration Applet Services and CSP Usage

4 Self-test

4.1 Power-on self-test

Each time the Module is powered up it tests that the cryptographic algorithms still operate correctly and that sensitive data have not been damaged. Power-on self–tests are available on demand by power cycling the Module.

On power on or reset, the Module performs the self-tests described in Table 14 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.

Test Target	Description
Firmware Integrity	16 bit CRC performed over all code located in EEPROM. This integrity test is not required or performed for code stored in masked ROM code memory.
RNG	Performs ANSI X9.31 KAT with fixed inputs
TDEA	Performs separate encrypt and decrypt KATs using 2-Key TDEA in ECB mode.
AES	Performs separate encrypt and decrypt KATs using an AES 128 key in ECB mode.
RSA	Performs separate RSA PKCS#1 signature and verification KATs using an RSA 1024 bit key.
ECDSA	Performs a pairwise consistency test using an ECC P-192 key pair.
ECC CDH	Performs an ECC CDH KAT using an ECC P-192 key pair.
SHA-1	Performs a SHA-1 KAT.
SHA-224	Performs a SHA-224 KAT.
SHA-256	Performs a SHA-256 KAT.
SHA-384	Performs a SHA-384 KAT.
SHA-512	Performs a SHA-512 KAT.

Table 16 – Power-On Self-Test

4.2 Conditional self-tests

On every call to the [ANS X9.31] RNG, the Module performs the FIPS 140-2 Continuous RNG test as described in AS09.42 to assure that the output is different than the previous value.

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 LOAD command, the Module verifies the integrity and authenticity of the new firmware using a TDEA MAC process and the ISD-SMAC key. Optionally, the Module may also verify a signature of the new firmware using the DAP-SVK public key or the DAP-DES key; the signature block in this scenario is signed by an external entity using the private key corresponding to DAP-SVK or the symmetric DAP-DES key.

5 Physical security policy

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 and is protected by passive shielding (metal layer coverings opaque to the circuitry below) and active shielding (a grid of top metal layer wires with tamper response). A tamper event detected by the active shield places the Module permanently into the Card Is Killed error state.

The Module is intended to be mounted in a plastic smartcard; physical inspection of the card boundaries is not practical after mounting. Physical inspection of modules for tamper evidence is performed using a lot sampling technique during the card assembly process. The Module also provides a key to protect the Module from tamper during transport and the additional physical protections listed in Section 7 below.

6 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.

7 Mitigation of other attacks policy

The Module implements defenses against:

- Light attacks
- Invasive fault attacks
- Side-channel attacks (SPA/DPA/EMA)

8 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.

END OF DOCUMENT