JCOP 4.7 SE051

Security Target Lite

Rev. 2.12 — 18 December 2025

Evaluation document

NSCIB-2500026-01

Document information

Information	Content
Keywords	ASE, JCOP, Common Criteria, EAL6 augmented
Abstract	This document contains information to fulfill the requirements of the Common Criteria component ASE (Security Target) for the Evaluation of the JCOP 4.7 SE051 developed and provided by NXP Semiconductors, Business Unit Security & Connectivity, according to the Common Criteria for Information Technology Security Evaluation Version 3.1 at EAL6 augmented

Revision history

Rev	Date	Description
2.12	18 December 2025	Updated reference to HW platform ST and UGM.
2.11	22nd May 2025	Updated UGM to v2.4
2.10	23 January 2025	Updated reference to HW platform ST
2.9	1 November 2024	Updated reference to HW platform ST, certificate ID and UGM.
2.8	24 March 2023	Updated UGM to v2.1.
2.7	13 March 2023	Corrected datasheet version
2.6	26 January 2023	Added back statement w.r.t PACE
2.5	20 January 2023	Added missing SIG_CIPHER_ECDSA_PLAIN to FCS_COP.1[ECSignature]. Updated UGM to v1.9. Removed statement w.r.t PACE.
2.4	3 January 2023	No update but sync the version with full ST's version.
2.3	1 November 2022	Updated datasheet and UGM references.
2.2	29 August 2022	Added a new configuration JCOP 4 SE051 v4.7 R3.02.11 built on top of HW configuration R4 with Crypto Library v0.7.7. Updated reference to HW platform ST and certificate ID. Updated UGM reference. Added missing SIG_CIPHER_ECDSA_PLAIN to FCS_COP.1[ECSignature] Updated datasheet reference.
2.1	29 June 2022	Clarified out-of-scope by adding PACE key agreement (GM) and PACE CAM support to Section 1.2.1 "Usage and Major Security Features of the TOE", item 10.
2.0	16 June 2022	Updated reference to UGM. Updated HW platform name and Cert ID. Removed OT.RNG from Section 2.4.3 "Security Objectives Statement".

1 ST Introduction (ASE_INT)

1.1 ST Reference and TOE Reference

Table 1. ST Reference and TOE reference

Title	JCOP 4.7 SE051 Security Target Lite
Version	Revision 2.12
Date	18 December 2025
Product Type	Java Card
TOE name	JCOP 4.7 SE051
Certification ID	NSCIB-2500026-01
CC version	Common Criteria for Information Technology Security Evaluation Version 3.1, Revision 5, April 2017 (Part 1 [1], Part 2 [2] and Part 3 [3])

1.2 TOE Overview

The TOE consists of the Micro Controller and a software stack which is stored on the Micro Controller and which can be executed by the Micro Controller. The software stack can be further split into the following components:

- Firmware for booting and low level functionality of the Micro Controller (MC FW) like writing to flash memory. This includes software for implementing cryptographic operations, called Crypto Library.
- Software for implementing a Java Card Virtual Machine [15], a Java Card Runtime Environment [16] and a Java Card Application Programming Interface [14], called JCVM, JCRE and JCAPI.
- Software for implementing content management according to GlobalPlatform [17], called GlobalPlatform (GP) Framework.
- Software for executing native libraries, called Secure Box.

The TOE is referred to as JCOP 4.7 SE051. The JCOP 4 Operating System (JCOP 4 OS) consists of the software stack without the Crypto Library (Crypto Lib) and without the Micro Controller Firmware (MC FW). The TOE uses one or more communication interfaces to communicate with its environment.

The complete TOE is depicted in Figure 1. The elements are described in more detail in Section 1.3 "TOE Description".

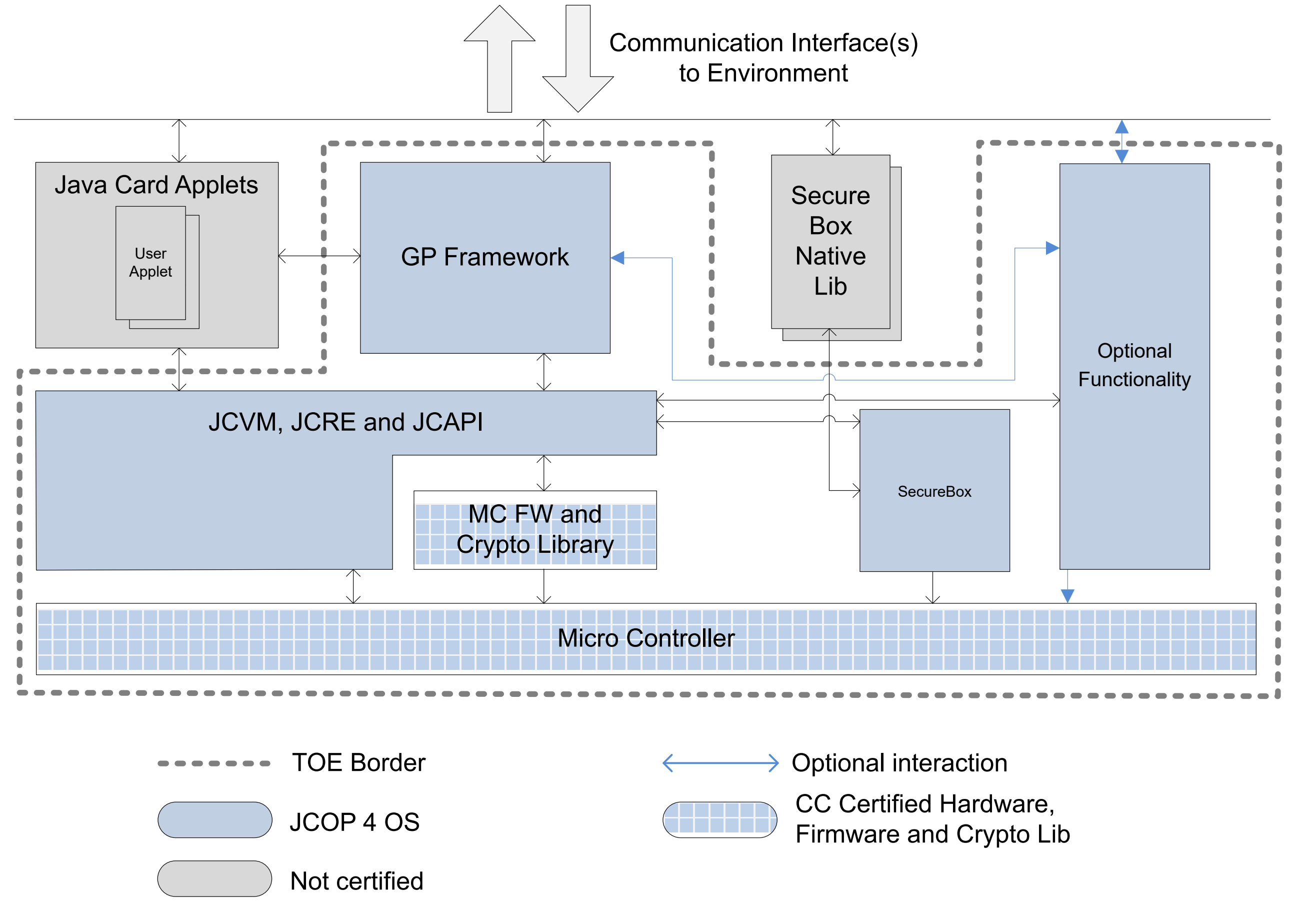


Figure 1. Components of the TOE

Figure 1 shows the components of the TOE. The TOE is a composite product on top of CC certified Hardware, Firmware and Crypto Library. Part of the TOE are the JCVM, JCRE, JCAPI and the GP Framework. Also included is optional functionality and the Secure Box mechanism. The Secure Box Native Librarys provide native functions for untrusted third parties and are not part of the TOE.

The figure shows Java Card applets which are small programs in Java language that can be executed by the TOE, but are not part of the TOE.

The TOE is available in two configurations as follows:

- JCOP 4 SE051 v4.7 R3.01.11 which is built on top of either configuration R1, R2 or R3 of the micro controller [10].
- JCOP 4 SE051 v4.7 R3.02.11 which is built on top of configuration R4 of the micro controller [10].

1.2.1 Usage and Major Security Features of the TOE

The usage of the TOE is focused on security critical applications in small form factors. One main usage scenario is the use of so called smart cards. Examples of such cards are banking cards or electronic drivers’ licenses. The TOE can also be used in an electronic passport. Another usage scenario is device authentication, where the TOE can be used to prove the authenticity or originality of a device like an accessory for a gaming console.

The TOE provides a variety of security features. The hardware of the Micro Controller already protects against logical and physical attacks by applying various sensors to detect manipulations and by processing data in ways which protect against leakage of data by side channel analysis. With the software stack the TOE provides many cryptographic primitives for encryption and decryption of data but also for signing and signature verification. Also the software stack contains security features to protect against attacks.

The following list contains the features of this TOE:

- Supported communication protocols:
  1. ISO 7816 T=1.
  2. ISO 7816 T=0.
  3. ISO 14443 T=CL.
  4. I2C Slave.
- Cryptographic algorithms and functionality:
  1. Data Encryption Standard with 3 keys (3DES) for en-/decryption (CBC and ECB) and MAC generation and verification (Retail-MAC, CMAC and CBC-MAC).
  2. Advanced Encryption Standard (AES) for en-/decryption (CBC, ECB and counter mode) and MAC generation and verification (CMAC, CBC-MAC).
  3. Rivest Shamir Adleman asymmetric algorithm (RSA) and RSA CRT for en-/decryption and signature generation and verification.
  4. RSA and RSA Chinese Remainder Theorem (CRT) key generation[1].
  5. Elliptic Curve Cryptography (ECC) over GF(p) for signature generation and verification (ECDSA)[1].
  6. ECC over GF(p) key generation[1].
  7. Random number generation according to class DRG.3 or DRG.4 of AIS 20 [21].
  8. Diffie-Hellman with ECDH and modular exponentiation[1].
  9. SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 hash algorithm.
  10. Following cryptographic algorithms are part of the TOE but without claims for security functional requirements:
      a. AES in Counter with CBC-MAC mode (AES CCM)[1].
      b. Keyed-Hash Message Authentication Code (HMAC)[1].
      c. HMAC based Key Derivation Function (HKDF) [28][1].
      d. Elliptic Curve Direct Anonymous Attestation (ECDAA) [30][1].
      e. Elliptic curve cryptography based on Edwards and Montgomery curves[1].
      f. PACE key agreement (Generic Mapping) and PACE CAM support[1].
- Java Card functionality:
  1. Executing the Java byte codes which are generated from the Java compiler when Java source code is compiled.
  2. Managing memory allocation of code and data of applets.
  3. Enforcing access rules between applets and the JCRE.
  4. Mapping of Java method calls to native implementations of e.g. cryptographic operation.

[1] Optional functionality

5. Support for Extended Length APDUs.
6. Garbage Collection with memory reclamation and compaction.
7. Persistent Memory Management and Transaction Mechanism.

- GlobalPlatform functionality:
  1. Loading of Java packages.
  2. Instantiating applet instances.
  3. Removing of Java packages.
  4. Removing of applet instances.
  5. Issuer Security Domain (ISD), Supplementary Security Domain (SSD).
  6. Creating SSDs.
  7. Associating applets to Security Domains.
  8. Installation of keys.
  9. Verification of signatures of signed applets.
  10. Verification of signatures for commands.
  11. CVM Management (Global PIN).
  12. Secure Channel Protocol (SCP01, SCP02 and SCP03).
  13. Delegated Management, Data Authentication Pattern (DAP).
  14. Post-issuance installation and deletion of applets and packages.
  15. Compliance to several GP configurations.
  16. Executable Load File Upgrade.
  17. Secure Element Management Service.
- NXP Proprietary Functionality
  1. Secure Box[2].
  2. Java Card APIs for:
     – Data encryption via PUF [12].
     – Data integrity protection with an EDC.
     – Asserting results of sensitive functions.
  3. Time representation and counter functionality[2].

1.2.2 TOE Type

The TOE is a Java Card with a GP Framework. It can be used to load and execute off-card verified Java Card applets.

1.2.3 Required non-TOE Hardware/Software/Firmware

Three groups of users shall be distinguished here.

The first group is the **end-users** group, which uses the TOE with one or more loaded applets in the final form factor like a banking card or an electronic passport. These users only require a communication device to be able to communicate with the TOE. The communication protocol of the TOE is standardized in either ISO7816 [31] (T=1, T=0), ISO14443 [32] (T=CL), or UM10204 [33] (I2C Slave).

The second group of users are **administrators of cards**. They want to configure the card by using the Configuration Module, to install additional applets and to configure and personalise these applets. These users require the same equipment as end-users.

[2] Optional functionality

The third group of users wants to develop Java Card applets and execute them on the TOE. These **applet developers** need in addition to the communication device a set of tools for the development of applets. This set of tools can be obtained from the TOE vendor and comprises elements such as PC development environment, byte code verifier, compiler, linker and debugger.

1.3 TOE Description

1.3.1 TOE Components and Composite Certification

The certification of this TOE is a composite certification. The following sections provide a more detailed description of the components of Figure 1. It is also made clear whether a component is covered by a previous certification or whether it is covered in the certification of this TOE.

1.3.1.1 Micro Controller

The Micro Controller is a secure smart card controller from NXP’s SmartMX3 family. The Micro Controller contains a co-processor for symmetric cryptographic operations, supporting DES and AES, as well as an accelerator for asymmetric cryptographic algorithms. The Micro Controller further contains a physical random number generator. The supported memory technologies are volatile (Random Access Memory (RAM)) and non-volatile (Read Only Memory (ROM) and FLASH) memory.

Access to all memory types is controlled by a Memory Management Unit (MMU) which allows to separate and restrict access to parts of the memory.

The Micro Controller has been certified in a previous certification and the results are re-used for this certification. The exact reference to the previous certification is given in the following Table 2 "Reference to Certified Micro Controller":

Table 2. Reference to Certified Micro Controller

Name	NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library (R1/R2/R3/R4)[1]
Certification ID	BSI-DSZ-CC-1136-V5-2026
Reference	[10]

[1] The SE051 hardware is an instantiation of the N7121 hard macro with I2C side-car.

1.3.1.2 Security IC Dedicated Software

1.3.1.2.1 MC FW (Micro Controller Firmware)

The Micro Controller Firmware is used for testing of the Micro Controller at production, for booting of the Micro Controller after power-up or after reset, for configuration of communication devices and for writing data to volatile and non-volatile memory.

The MC FW has been certified together with the Micro Controller and the same reference [10] as given for the Micro Controller also apply for the MC FW.

1.3.1.2.2 Crypto Library

The Crypto Library provides implementations for symmetric and asymmetric cryptographic operations, hashing, the generation of hybrid deterministic and hybrid physical random numbers and further functions like secure copy and compare.

Some of the cryptographic algorithms offered by the Crypto Lib are not certified, see Section 1.3.1.4 "Excluded functionality".

The symmetric cryptographic operations comprise the algorithms 3DES and AES. These algorithms use the symmetric co-processor of the Micro Controller.

The supported asymmetric cryptographic operations are ECC and RSA. These algorithms use the Public Key Crypto Coprocessor (PKCC) of the Micro Controller for the cryptographic operations.

The Crypto Library has been certified together with the Micro Controller and the same reference [10] as given for the Micro Controller also applies.

1.3.1.3 Security IC Embedded Software

1.3.1.3.1 JCOP 4.7 SE051

The OS of the TOE consists of JCVM, JCRE, JCAPI and GP framework. It is implemented according to the Java Card Specification and GlobalPlatform. Additionally it consists of a proprietary API, which is described in the UGM [9].

The TOE can be identified by using the IDENTIFY APDU command (see UGM [9]). This command returns the card identification data, which includes a Platform ID, a Patch ID and other information that allows to identify the content in ROM, FLASH and loaded patches (if any).

The TOE also includes a Configuration Module (see Section 1.3.2 "Optional TOE Functionality") which is used for personalisation and configuration of the TOE. It must be deleted after the personalisation is finished (end of Phase 6 ”Personalisation”) by using the DELETE APDU command. Once the Configuration Module is deleted, it is no longer possible to configure the TOE.

The TOE contains further functionality for integrity protection of user data via an EDC, encryption of user data via PUF [12] and optional functionality as described in Section 1.3.2 "Optional TOE Functionality".

1.3.1.4 Excluded functionality

All Secure Box Native Libraries are not part of the TOE. No security functional requirements are claimed on AES CCM, HMAC, HKDF, ECDAA, elliptic curve cryptography based on Edwards and Montgomery curves and FIPS self-tests, they are TSF non-interfering.

1.3.2 Optional TOE Functionality

Some dedicated functionality of the TOE as listed below can be removed:

- RSA key generation,
- Elliptic curve cryptographic functionality,
- eGov accelerators,
- TOE self-tests according to FIPS 140-2 [22],
- SecureBox,
- TOE Configuration Module (the TOE Configuration Module has to be deleted at the end of life cycle phase 6) [9],
- AES CCM as defined in the Java Card AEADCipher API [14], HMAC and HKDF cryptographic functionality as defined in the Java Card API [14] and the UGM [9]. Timer functionality as defined in the UGM [9].

- ECDAA [30] and elliptic curve cryptography based on Edwards and Montgomery curves.
- I2C slave protocol and T = 1 over I2C.

1.3.3 TOE Reduced Feature Set

The TOE is also available to customers with a reduced feature set. This variant of the TOE contains a different base mask configuraton and has some features removed from the TOE. The removed features are:

- No support for SecureBox
- No RSA support in GP
- No DES support in GP
- SCP01 not supported
- SCP02 not supported
- GP CVM service not supported
- GP command chaining is not supported
- ISO 7816 T=1, T=0 (UART) interface not supported

The base mask with the reduced feature set is known as IOT Reduced, while the base mask with full functionality is known as IOT Full. For more information on the difference in feature set, please refer to Table 3.1 in the User Guidance and Administration Manual [9].

1.3.3.1 Reduced Security Functional Requirements

Some security functionality claimed in this Security Target is no longer available in the IOT Reduced base mask or has limited functionality and the corresponding Security Functional Requirements are not (completely) applicable for this variant of the TOE. The table below lists which SFRs are affected in the IOT Reduced variant of the TOE.

Table 3. SFRs affected in the IOT Reduced variant of the TOE

Functionality	SFRs
Secure Box	FDP_ACC.2[SecureBox] is no longer available. FDP_ACF.1[SecureBox] is no longer available. FMT_MSA.1[SecureBox] is no longer available. FMT_MSA.3[SecureBox] is no longer available. FMT_SMF.1[SecureBox] is no longer available.
RSA support in GP	FCS_COP.1.1[DAP] no longer supports algorithm ALG_RSA_SHA_PKCS1.

1.3.4 TOE Life Cycle

The life cycle for this Java Card is based on the general smart card life cycle defined in the Java Card Protection Profile - Open Configuration [6], see Figure 2.

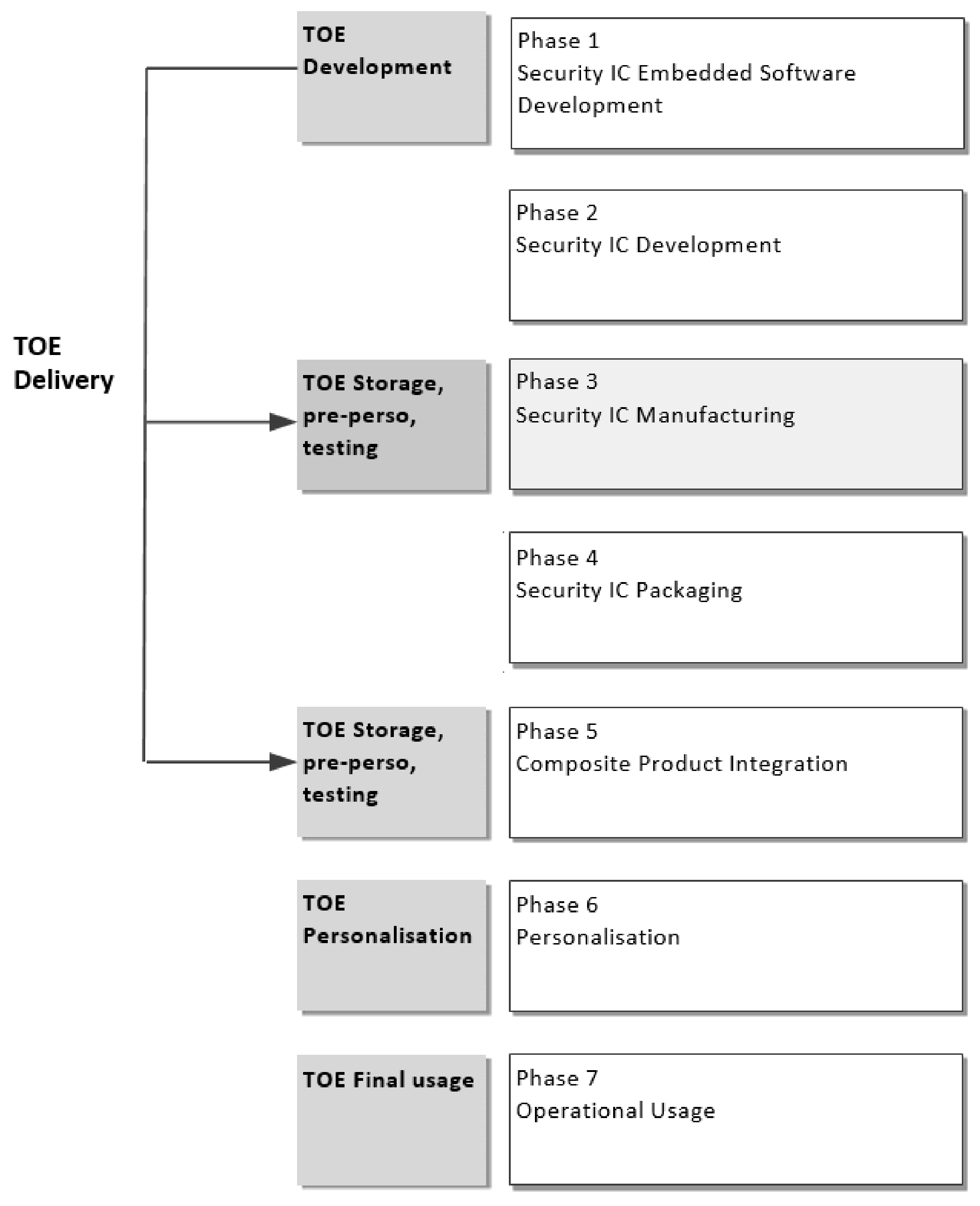


Figure 2. TOE Life Cycle within Product Life Cycle

Table 4. TOE Life Cycle phases

Phase	Name	Description
1	Security IC Embedded Software Development	The Security IC Embedded Software Developer is in charge of • SmartCard embedded software development including the development of Java Card applets and • specification of IC pre-personalization requirements, though the actual data for IC pre-personalization come from phase 4, 5, or 6.

Table 4. TOE Life Cycle phases*...continued*

Phase	Name	Description
2	Security IC Development	The IC Developer • designs the IC, • develops Security IC Dedicated Software, • provides information, software or tools to the Security IC Embedded Software Developer, and • receives the embedded software from the developer, through trusted delivery and verification procedures. From the IC design, Security IC Dedicated Software and Smart-Card Embedded Software, the IC Developer • constructs the SmartCard IC database, necessary for the IC photomask fabrication.
3	Security IC Manufacturing	The IC Manufacturer is responsible for • producing the IC through three main steps: IC manufacturing, IC testing, and IC pre-personalization. The IC Mask Manufacturer • generates the masks for the IC manufacturing based upon an output from the SmartCard IC database. Configuration items may be changed.
4	Security IC Packaging	The IC Packaging Manufacturer is responsible for • IC packaging and testing.
5	Composite Product Integration	The Composite Product Manufacturer is responsible for • SmartCard product finishing process including applet loading and testing. Configuration items may be changed by using the Configuration Module.
6	Personalization	The Personalizer is responsible for • SmartCard (including applet) personalization and final tests. User Applets may be loaded onto the chip at the personalization process and configuration items may be changed by using the Configuration Module, which must be deleted at the end of this cycle by using the DELETE APDU command.
7	Operational Usage	The Consumer of Composite Product is responsible for • SmartCard product delivery to the SmartCard end-user, and the end of life process. • applets may be loaded onto the chip.

The evaluation process is limited to phases 1 to 5. User Applet development is outside the scope of this evaluation. Applets can be loaded into FLASH in phases 3, 4, 5, and 6. Applet loading in phase 7 is also allowed. This means post-issuance loading of applets can be done for a certified TOE.

The Configuration Module is loaded into FLASH and has special privileges to personalize and configure the TOE. Before life cycle Phase 7 "Operational Use" the Configuration Module is deleted and hence it is ensured that its functionality cannot be used afterwards. It is possible to load patch code into FLASH in phases 3, 4, 5, and 6. The certification is only valid for the ROM code having the Platform Identifiers and the Patch IDs (if applicable) as stated in Table 5 "Delivery Items".

The delivery process from NXP to their customers (to phase 4 or phase 5 of the life cycle) guarantees that the customer is aware of the exact versions of the different parts of the TOE as outlined above.

TOE documentation is delivered in electronic form (encrypted according to defined mailing procedures).

Note: *Phases 1 to 3 are under the TOE developer scope of control. Therefore, the objectives for the environment related to phase 1 to 3 are covered by Assurance measures, which are materialized by documents and procedures evaluated through the TOE evaluation process.*

During phases 4 to 7 the TOE is no more under the developer control. In this environment, the TOE protects itself with its own Security functions. But some additional usage procedures must also be followed in order to ensure that the TOE is correctly and securely handled, and not damaged or comprised. This ST assumes (A.USE_DIAG, A.USE_KEYS) that users handle securely the TOE and related Objectives for the environment are defined (OE.USE_DIAG, OE.USE_KEYS).

1.3.5 TOE Identification

1.3.5.1 TOE Delivery

The delivery comprises the following items:

Table 5. Delivery Items

Type	Name	Version	Form of Delivery
Hardware	NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library	• B1 (R1, R2 or R3 configurations) for JCOP 4 SE051 v4.7 R3.01.11 • B1 (R4 only configuration) for JCOP 4 SE051 v4.7 R3.02.11	Micro Controller including on-chip software: Firmware and Crypto Lib[1]
JCOP 4 OS (SE051 v4.7 R3.01.11 or R3.02.11)	ROM Code (Platform ID) FLASH content (FLASH ID) Patch Code (Patch ID)	see UGM [9]	On-chip software[2]: JCOP 4 OS (SE051 v4.7 R3.01.11 or R3.02.11)
Document	User Guidance and Administration Manual [9]	2.5	Electronic document[3]
Document	HW Product Data Sheet [11]	1.6	Electronic document[3]

[1] The TOE is delivered as wafer or module. The TOE can be collected at NXP site or is being shipped to the customer. See UGM [9] for details.

[2] Included in the Micro Controller.

[3] Via the NXP Docstore [13].

1.3.5.2 TOE Identification

The TOE can be identified by using the Platform ID, the FLASH ID and the Patch ID. The IDENTIFY command and the identification output for this TOE are described in detail in Chapter 2 of the UGM [9]. The IDENTIFY command also returns information about presence of optional functionality and allows to identify the IOT Full and IOT Reduced base mask configuration.

1.3.6 Evaluated Package Types

A number of package types are supported for this TOE. All package types, which are covered by the certification of the used hardware (see [10]), are also allowed to be used in combination with each product of this TOE.

The package types do not influence the security functionality of the TOE. They only define which pads are connected in the package and for what purpose and in which environment the chip can be used. Note that the security of the TOE is not dependent on which pad is connected or not - the connections just define how the product can be used. If the TOE is delivered as wafer the customer can choose the connection on his own.

2 Conformance Claims (ASE_CCL)

This Chapter is divided into the following sections: "CC Conformance Claim", "Package Claim", "PP Claim", and "Conformance Claim Rationale".

2.1 CC Conformance Claim

This Security Target claims to be conformant to version 3.1 of Common Criteria for Information Technology Security Evaluation according to

- "Common Criteria for Information Technology Security Evaluation, Part 1, Version 3.1, Revision 5, April 2017" [1]
- "Common Criteria for Information Technology Security Evaluation, Part 2, Version 3.1, Revision 5, April 2017" [2]
- "Common Criteria for Information Technology Security Evaluation, Part 3, Version 3.1, Revision 5, April 2017" [3]

The following methodology will be used for the evaluation:

- Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1, Revision 5, April 2017" [4]

This Security Target claims to be CC Part 2 extended and CC Part 3 conformant. The extended Security Functional Requirements are defined in Section 6 "Extended Components Definition (ASE_ECD)".

2.2 Package Claim

This Security Target claims conformance to the assurance package EAL6. The augmentation to EAL6 is ASE_TSS.2 “TOE summary specification with architectural design summary” and ALC_FLR.1 “Basic flaw remediation”.

2.3 PP Claim

The Security Target claims demonstrable conformance to the Java Card System - Open Configuration Protection Profile, December 2017, Version 3.0.5 [6], certified by Bundesamt für Sicherheit in der Informationstechnik (BSI, BSI-CC-PP-0099-2017). The Java Card Protection Profile makes the use of Java Card RMI optional. The TOE does not support Java Card RMI. This ST is more restrictive than the PP [6] which Section 2.4 "Conformance Claim Rationale" provides a rational for.

2.4 Conformance Claim Rationale

2.4.1 TOE Type

The TOE type as stated in Section Section 1.2 "TOE Overview" of this ST corresponds to the TOE type of the PP as stated in Section 2.1 of [6] namely a Java Card platform, implementing the Java Card Specification Version 3.0.5 [15], [16], [14].

2.4.2 SPD Statement

2.4.2.1 Threats

The Security Problem Definition (SPD) statement that is presented in Section 4 "Security Problem Definition (ASE_SPD)" includes the threats as presented in the PP [6], but also includes additional threats. These threats are:

- T.OS_OPERATE
- T.RND
- T.COM_EXPLOIT
- T.LIFE_CYCLE
- T.UNAUTHORIZED_CARD_MNGT
- T.INTEG-APPLI-DATA[REFINED]
- T.CONFIG
- T.SEC_BOX_BORDER
- T.MODULE_EXEC
- T.MODULE_REPLACEMENT

The threat T.OS_OPERATE is an additional threat added to cover incorrect operating system behavior, it is an addition to the threats in the PP [6].

The threat T.RND is taken from the Security IC PP [5].

The threat T.COM_EXPLOIT is included to cover communication channels attacks and it is an addition to the threats in the PP [6].

The threat T.LIFE_CYCLE is included to cover content management attacks and it is an addition to the threats in the PP [6].

The threat T.UNAUTHORIZED_CARD_MNGT refines the threats T.INSTALL and T.DELETION from the Security IC PP [5].

The threat T.INTEG-APPLI-DATA[REFINED] refines the threat T.INTEG-APPLI-DATA in the Security IC PP [5].

The threat T.CONFIG is an additional threat to cover unauthorized modifications and read access of the configuration area in the TOE. It is an addition to the threats defined in the PP [6].

The threat T.SEC_BOX_BORDER is included for the Secure Box which is additional functionality the PP [6] allows.

The threats T.MODULE_EXEC and T.MODULE_REPLACEMENT are included for the Modular Design which is additional functionality the PP [6] allows. Furthermore some threats from the PP [6] are refined to cover additional assets from the Modular Design. This comprises threats T.CONFID-JCS-CODE, T.CONFID-JCS-DATA, T.INTEG-APPLI-CODE, T.INTEG-JCS-CODE, T.INTEG-JCS-DATA, and T.SID.1.

Note that the threat T.EXE-CODE-REMOTE is not included, since the TOE does not support Java Card RMI. The Java Card Protection Profile [6] makes the use of Java Card RMI optional.

2.4.2.2 Organizational Security Policies

The SPD statement presented in Section 4 "Security Problem Definition (ASE_SPD)", copies the OSP from the PP [6], and adds following additional OSPs:

- OSP.PROCESS-TOE

- OSP.KEY-CHANGE
- OSP.SECURITY-DOMAINS
- OSP.SECURE-BOX

The Organizational Security Policy (OSP) OSP.PROCESS-TOE is introduced for the pre-personalisation feature of the TOE and is an addition to the OSPs in PP [6]. This OSP is copied from the Security IC PP [5].

The OSP OSP.KEY-CHANGE is introduced for the Security Domain (SD) feature of the TOE and is an addition to the OSPs in PP [6].

The OSP OSP.SECURITY-DOMAINS is introduced for the SD feature of the TOE and is an addition to the OSPs in PP [6].

The OSP.SECURE-BOX is introduced to allow execution of third party native code and is an addition to the OSPs in PP [6].

2.4.2.3 Assumptions

The SPD statement includes two of the three assumptions from the PP [6]. The assumption A.Deletion is excluded. The Card Manager is part of the TOE and therefore the assumption is no longer relevant. Leaving out the assumption, makes the SPD of this ST more restrictive than the SPD in the PP [6]. As the Card Manager is part of the TOE, it is ensuring that the deletion of applets through the Card Manager is secure, instead of assuming that it is handled by the Card Manager in the environment of the TOE.

Besides the assumptions from the PP [6], following additional assumptions are added:

- A.PROCESS-SEC-IC
- A.USE_DIAG
- A.USE_KEYS
- A.APPS-PROVIDER
- A.VERIFICATION-AUTHORITY

The assumption A.PROCESS-SEC-IC is taken from the underlying certified Micro Controller [10], which is compliant to the Security IC PP [5].

The assumptions A.USE_DIAG and A.USE_KEYS are included because the Card Manager is part of the TOE and no longer part of the environment.

The assumptions A.APPS-PROVIDER and A.VERIFICATION-AUTHORITY are added because Security Domains from the GlobalPlatform Specification are introduced. All the applets and packages are signed by the Application Provider Security Domain (APSD) and the correctness is verified on the TOE by Verification Authority Security Domain (VASD) before the package or applet is installed or loaded. A.APPS-PROVIDER and A.VERIFICATION-AUTHORITY are additions to PP [6] for card content management environment.

2.4.3 Security Objectives Statement

The statement of security objectives in the ST presented in Section 5 "Security Objectives" includes all security objectives as presented in the PP [6], but also includes a number of additional security objectives. These security objectives are:

- OT.IDENTIFICATION
- OT.DOMAIN-RIGHTS
- OT.APPLI-AUTH

- OT.COMM_AUTH
- OT.COMM_INTEGRITY
- OT.COMM_CONFIDENTIALITY
- OT.CARD-CONFIGURATION
- OT.SEC_BOX_FW
- OT.SID_MODULE
- OT.SECURE_LOAD_ACODE
- OT.SECURE_ACTIVATION_ADDITIONAL_CODE
- OT.TOE_IDENTIFICATION

The security objective OT.IDENTIFICATION is part of the security objectives of the certified Micro Controller [10] (see also Section 1.3.1.1 "Micro Controller") and Crypto Lib [10] (see also Section 1.3.1.2.2 "Crypto Library"), which are also components of this composite certification. Therefore the security objective statement is equivalent to the PP [6] for these two security objectives. OT.IDENTIFICATION is also included for the pre-personalisation feature of the TOE, which is additional functionality the PP allows.

The security objectives OT.DOMAIN-RIGHTS, OT.APPLI-AUTH, OT.COMM_AUTH, OT.COMM_INTEGRITY, OT.COMM_CONFIDENTIALITY are objectives for the TOE as the GlobalPlatform API and the definitions for Secure Channel, Security Domains and Card Content Management are used from it.

The security objectives OT.CARD-CONFIGURATION, OT.SECURE_LOAD_ACODE, OT.SECURE_ACTIVATION_ADDITIONAL_CODE, OT.TOE_IDENTIFICATION are related to the configuration of the TOE via the Configuration Module, which is additional functionality the PP [6] allows.

The security objective OT.SEC_BOX_FW is related to the Secure Box, which is additional functionality the PP allows.

The security objective OT.SID_MODULE is related to the Modular Design of the TOE, which is additional functionality the PP [6] allows.

The ST contains OE.APPLET, OE.VERIFICATION and OE.CODE-EVIDENCE from Security Objectives for the Operational Environment from [6]. Additionally, some of the Security Objectives for the Operational Environment from [6] are listed as TOE Security Objectives in this ST:

- OT.SCP.RECOVERY instead of OE.SCP.RECOVERY
- OT.SCP.SUPPORT instead of OE.SCP.SUPPORT
- OT.SCP.IC instead of OE.SCP.IC
- OT.CARD-MANAGEMENT instead of OE.CARD-MANAGEMENT

OT.SCP.RECOVERY, OT.SCP.SUPPORT, and OT.SCP.IC are objectives for the TOE as the Smart Card Platform belongs to the TOE for this evaluation. OT.CARD-MANAGEMENT is an objective for the TOE as the Card Manager belongs to the TOE for this evaluation. Moving objectives from the environment to the TOE adds objectives to the TOE without changing the overall objectives. The statement of security objectives is therefore equivalent to the security objectives in the PP [6] to which conformance is claimed.

The security objectives O.INSTALL, O.LOAD, and O.DELETION from the PP [6] are not included since these functionality and objectives are covered by the refined OT.CARD-MANAGEMENT.

Note that the objective O.REMOTE is not included, since the TOE does not support Java Card RMI. The Java Card Protection Profile makes the use of Java Card RMI optional.

A part of the security objectives for the environment defined in the PP [6] has been included in this ST. The other part of security objectives for the environment, which is present in the PP [6], is used as part of the security objectives for the TOE in this ST. The ST also introduces following additional security objectives for the environment:

- OE.PROCESS_SEC_IC
- OE.USE_DIAG
- OE.USE_KEYS
- OE.APPS-PROVIDER
- OE.VERIFICATION-AUTHORITY
- OE.KEY-CHANGE
- OE.SECURITY-DOMAINS

The security objective for the environment OE.PROCESS_SEC_IC is from the hardware platform (Micro Controller [10] see also Section 1.3.1.1 "Micro Controller") that is part of this composite product evaluation. Therefore the statement of security objectives for the environment is equivalent to the statement in the Security IC PP [5].

OE.USE_KEYS and OE.USE_DIAG are included because the Card Manager is part of the TOE and not a security objective for the environment as in PP [6].

OE.APPS-PROVIDER and OE.VERIFICATION-AUTHORITY cover trusted actors which enable the creation, distribution and verification of secure applications.

OE.KEY-CHANGE covers the switch to trusted keys for the AP.

OE.SECURITY-DOMAINS covers the management of security domains in the context of the GlobalPlatform Specification.

The statement of security objectives for the environment is therefore considered to be equivalent to the security objectives in the PP [6] to which conformance is claimed.

2.4.4 Security Functional Requirements Statement

The statement of security functional requirements copies most SFRs as defined in the PP [6], with the exception of a number of options. For the copied set of SFRs the ST is considered equivalent to the statement of SFRs in the PP [6]. Moreover as requested by the PP [6] the ST adds additional threats, objectives and SFRs to fully cover and describe additional security functionality implemented in the TOE.

The TOE does not implement Java Card RMI, therefore this ST restricts remote access from the CAD to the services implemented by the applets on the card to none. As a result the SFRs concerning Java Card RMI (FDP_ACF.1/JCRMI, FDP_IFC.1/JCRMI, FDP_IFF.1/JCRMI, FMT_MSA.1/EXPORT, FMT_MSA.1/REM_ REFS, FMT_MSA.3/ JCRMI, FMT_SMF.1/JCRMI, FMT_REV.1/JCRMI, and FMT_SMR.1/JCRMI) are not included in the ST. In the PP [6] the use of the Java Card RMI is optional.

The SFR FDP_ITC.2/INSTALLER from the PP [6] is replaced by FDP_ITC.2[CCM] which enforces the Security Domain access control policy and the Secure Channel Protocol information flow policy and which are more restrictive than the PACKAGE LOADING information flow control SFP from PP [6].

The set of SFRs that define the card content management mechanism CarG are partly replaced or refined and are considered to be equivalent or more restrictive because of the newly introduced SFPs:

1. Security Domain access control policy,
2. Secure Channel Protocol information flow policy

provide a concrete and more restrictive implementation of the PACKAGE LOADING information flow control SFP from PP [6].

The table below lists the SFRs from CarG of PP [6] and their corresponding refinements in this ST.

Table 6. CarG SFRs refinements

SFR from PP [6]	Refinement
FCO_NRO.2/CM	FCO_NRO.2[SC]
FDP_IFC.2/CM	FDP_IFC.2[SC]
FDP_IFF.1/CM	FDP_IFF.1[SC]
FDP_UIT.1/CM	FDP_UIT.1[CCM]
FIA_UID.1/CM	FIA_UID.1[SC]
FMT_MSA.1/CM	FMT_MSA.1[SC]
FMT_MSA.3/CM	FMT_MSA.3[SC]
FMT_SMF.1/CM	FMT_SMF.1[SC]
FMT_SMR.1/CM	FMT_SMR.1[SD]
FTP_ITC.1/CM	FTP_ITC.1[SC]

The following SFRs realize refinements of SFRs from PP [6] and add functionality to the TOE making the statement of security requirements more restrictive than the PP [6]:

FDP_ROL.1[CCM] and FPT_FLS.1[CCM] realize additional security functionality for the card manager which is allowed by the PP [6].

The set of SFRs that define the security domains mechanism as specified by GlobalPlatform realize refinements of SFRs from PP [6] (see above Table 6 "CarG SFRs refinements") and additional security functionality which is allowed by the PP [6]. This set of SFRs comprise FDP_ACC.1[SD], FDP_ACF.1[SD], FMT_MSA.1[SD], FMT_MSA.3[SD], FMT_SMF.1[SD], and FMT_SMR.1[SD].

The set of SFRs that define the secure channel mechanism as specified by GlobalPlatform realize refinements of SFRs from PP [6] (see above Table 6 "CarG SFRs refinements"), add additional security functionality and include a JCOPX API which is allowed by the PP [6]. This set of SFRs comprise FCO_NRO.2[SC], FDP_IFC.2[SC], FDP_IFF.1[SC], FMT_MSA.1[SC], FMT_MSA.3[SC], FMT_SMF.1[SC], FIA_UID.1[SC], FIA_UAU.1[SC], FIA_UAU.4[SC], and FTP_ITC.1[SC].

The set of SFRs that define the Configuration Module realize additional security functionality, which is allowed by the PP [6]. This set of SFRs comprise FDP_IFC.2[CFG], FDP_IFF.1[CFG], FIA_UID.1[CFG], FMT_MSA.1[CFG], FMT_MSA.3[CFG], FMT_SMF.1[CFG] and FMT_SMR.1[CFG].

The set of SFRs that define the Secure Box, realize additional security functionality which is allowed by the Protection Profile (PP) [6]. This set of SFRs comprise FDP_ACC.2[SecureBox], FDP_ACF.1[SecureBox], FMT_MSA.1[SecureBox], FMT_MSA.3[SecureBox], and FMT_SMF.1[SecureBox].

The set of SFRs that define the Modular Design realize additional security functionality, which is allowed by the PP [6]. This set of SFRs comprise FDP_IFC.1[MODULAR-DESIGN], FDP_IFF.1[MODULAR-DESIGN], FIA_ATD.1[MODULAR-DESIGN], FIA_USB.1[MODULAR-DESIGN], FMT_MSA.1[MODULAR-DESIGN], FMT_MSA.3[MODULAR_DESIGN, FMT_SMF.1[MODULAR-DESIGN], FMT_SMR.1[MODULAR-DESIGN], and FPT_FLS.1[MODULAR-DESIGN].

Some SFRs from the PP [6] are refined to cover deletion of Modules. This makes the SFRs more restrictive which is allowed by the PP [6]. This set of SFRs comprise FDP_ACC.2[ADEL], FDP_ACF.1[ADEL], FMT_SMF.1[ADEL], and FPT_FLS.1[ADEL].

The SFRs FAU_SAS.1[SCP], FIA_AFL.1[PIN], FPT_EMSEC.1 and FPT_PHP.3 realize additional security functionality which is allowed by the PP [6]. The SFRs FCS_CKM.2 and FCS_CKM.3 realize security functionality required by the Java Card API [14] which is allowed by the PP [6].

3 Security Aspects

This Chapter describes the main security issues of the Java Card System and its environment addressed in this ST, called "security aspects", in a CC-independent way. In addition to this, they also give a semi-formal framework to express the CC security environment and objectives of the TOE. They can be instantiated as assumptions, threats, objectives (for the TOE and the environment) or organizational security policies. The description is based on [6].

This chapters described only the Security Aspects which have been added in comparison to the PP [6].

3.1 Integrity

3.1.1 SA.INTEG-APPLI-DATA-PHYS: Integrity of Application Data (Sensitive Result)

Integrity-sensitive application data must be protected against unauthorized modification by physical attacks.

3.2 Configuration Module

3.2.1 SA.CONFIGURATION-MODULE: Configuration Module

The Configuration Module is a JCOP functionality which allows to read and modify configuration items in the configuration area of the TOE.

3.3 Modular Design

3.3.1 SA.MODULAR-DESIGN: Modular Design

The TOE might contain one or more Modules implementing particular functionality. The list of Modules present in the TOE must be retrievable. The Modules have an associated AID which allows to identify them. The AID is equivalent to the Package AID of JavaCard packages. Modules can only be deleted, re-loading of a previously deleted module or replacing a present module must not be possible. Interfaces to a Module can be Public or TOE internal. Public Interfaces can directly be accessed by any Applet or via an APDU, TOE internal interfaces can only be accessed by the TOE itself, Applets use the corresponding JavaCard API [14].

3.3.2 SA.MODULE-INVOCATION: Module Invocation

Invoking a module must be transparent to the user. If a Module has a TOE internal interface, is not present and is invoked by the user, the TOE must preserve a secure state by throwing an exception or returning an appropriate error status word to the CAD.

4 Security Problem Definition (ASE_SPD)

The Security Problem Definition is described within the following sections. Only additional and refined items compared to the PP [6] are described.

4.1 Assets

Assets are security-relevant elements to be directly protected by the TOE. Confidentiality of assets is always intended with respect to un-trusted people or software, as various parties are involved during the first stages of the smart card product life-cycle. Details concerning the threats are given in Section 4.2 "Threats" hereafter.

Assets have to be protected, some in terms of confidentiality and some in terms of integrity or both integrity and confidentiality. These assets might get compromised by the threats that the TOE is exposed to.

The assets to be protected by the TOE are listed below. They are grouped according to whether it is data created by and for the user (User data) or data created by and for the TOE (TSF data). This definition of grouping is taken from Section 5.1 of [6].

4.1.1 User Data

D.APSD_KEYS	Refinement of D.APP_KEYS of [6]. Application Provider Security Domains cryptographic keys are needed to establish secure channels with the AP. These keys can be used to load and install applications on the card if the Security Domain has the appropriate privileges. To be protected from unauthorized disclosure and modification.
D.ISD_KEYS	Refinement of D.APP_KEYS of [6]. Issuer Security Domain cryptographic keys are needed to perform card management operations on the card. To be protected from unauthorized disclosure and modification.
D.VASD_KEYS	Refinement of D.APP_KEYS of [6]. Verification Authority Security Domain cryptographic keys needed to verify applications Mandated DAP signature. To be protected from unauthorized disclosure and modification.
D.CARD_MNGT_DATA	The data of the card management environment, like for instance, the identifiers, the privileges, life cycle states. To be protected from unauthorized modification.

4.1.2 TSF Data

D.CONFIG_ITEM	A configuration that can be changed using the Configuration Mechanism.
D.MODULE_CODE	The code of a Module. The code of a module might comprise Java code, native code, code of a native Library or a combination of them. To be protected against unauthorized disclosure and modification. Further to be protected against unauthorized removal or presence forgery.
D.MODULE_DATA	Private data of a Module, like the contents of its private fields. To be protected from unauthorized disclosure and modification.

4.2 Threats

4.2.1 Integrity

4.2.1.1 T.INTEG-APPLI-DATA[REFINED]: Integrity of Application Data

The attacker executes an application to alter (part of) another application's data. Directly threatened asset(s): D.APP_I_ DATA, D.PIN, D.APP_KEYS, D.ISD_KEYS, D.VASD_KEYS and S.APSD_KEYS.

This threat is a refinement of the Threat T.INTEG-APPLI-DATA from [6].

4.2.2 Unauthorized Execution

4.2.2.1 T.MODULE_EXEC: Code Execution of Modules

The attacker bypasses the presence check of a Module which is not present with TOE internal interface to execute arbitrary code. See SA.MODULAR-DESIGN and SA.MODULE-INVOCATION for details. Directly threatened asset(s): D.MODULE_CODE.

4.2.3 Card Management

4.2.3.1 T.UNAUTHORIZED_CARD_MNGT: Unauthorized Card Management

The attacker performs unauthorized card management operations (for instance impersonates one of the actor represented on the card) in order to take benefit of the privileges or services granted to this actor on the card such as fraudulent:

- load of a package file
- installation of a package file
- extradition of a package file or an applet
- personalization of an applet or a Security Domain
- deletion of a package file or an applet
- privileges update of an applet or a Security Domain

Directly threatened asset(s): D.ISD_KEYS, D.APSD_KEYS, D.APP_C_DATA, D.APP_I_DATA, D.APP_CODE, D.SEC_DATA, and D.CARD_MNGT_DATA (any other asset may be jeopardized should this attack succeed, depending on the virulence of the installed application).

This security objective is a refinement of the Threats T.INSTALL and T.DELETION from [6].

4.2.3.2 T.COM_EXPLOIT: Communication Channel Remote Exploit

An attacker remotely exploits the communication channels established between a third party and the TOE in order to modify or disclose confidential data.

All assets are threatened.

4.2.3.3 T.LIFE_CYCLE: Life Cycle

An attacker accesses to an application outside of its expected availability range thus violating irreversible life cycle phases of the application (for instance, an attacker repersonalizes the application). Directly threatened asset(s): D.APP_I_ DATA, D.APP_C_DATA, and D.CARD_MNGT_DATA.

4.2.4 Operating System

4.2.4.1 T.OS_OPERATE: Incorrect Operating System Behavior

Modification of the correct OS behavior by unauthorized use of TOE or use of incorrect or unauthorized instructions or commands or sequence of commands, in order to obtain an unauthorized execution of the TOE code. An attacker may cause a malfunction of TSF or of the Smart Card embedded OS in order to (1) by-pass the security mechanisms (i.e. authentication or access control mechanisms) or (2) obtain unexpected result from the embedded OS behavior. Different kind of attack path may be used as:

1. Applying incorrect unexpected or unauthorized instructions, commands or command sequences,
2. Provoking insecure state by insertion of interrupt (reset), premature termination of transaction or communication between IC and the reading device.

Info: Any implementation flaw in the OS itself can be exploited with this attack path to lead to an unsecured state of the state machine of the OS. The attacker uses the available interfaces of the TOE. A user could have certain specified privileges that allow loading of selected programs. Unauthorized programs, if allowed to be loaded, may include either the execution of legitimate programs not intended for use during normal operation (such as patches, filters, Trojan horses, etc.) or the unauthorized loading of programs specifically targeted at penetration or modification of the security functions. Attempts to generate a non-secure state in the Smart Card may also be made through premature termination of transactions or communications between the IC and the card reading device, by insertion of interrupts, or by selecting related applications that may leave files open.

4.2.5 Random Numbers

4.2.5.1 T.RND: Deficiency of Random Numbers

An attacker may predict or obtain information about random numbers generated by the TOE for instance because of a lack of entropy of the random numbers provided. An attacker may gather information about the produced random numbers which might be a problem because they may be used for instance to generate cryptographic keys. Here the attacker is expected to take advantage of statistical properties of the random numbers generated by the TOE without specific knowledge about the TOE’s generator. Malfunctions or premature ageing are also considered which may assist in getting information about random numbers.

4.2.6 Configuration Module

4.2.6.1 T.CONFIG: Unauthorized configuration

The attacker tries to change configuration items without authorization. Directly threatened asset(s): D.CONFIG_ITEM.

4.2.7 Secure Box

4.2.7.1 T.SEC_BOX_BORDER: SecureBox Border Infringement

An attacker may try to use malicious code placed in the Secure Box to modify the correct behavior of the Operating System (OS). With the aim to

1. disclose the Java Card System code,
2. disclose or alter applet code, disclose or alter Java Card System data, or disclose or alter applet data.

4.2.8 Module replacement

4.2.8.1 T.MODULE_REPLACEMENT: Replacement of Module

An attacker loads a Module with functionality differing from a previously deleted Module to bypass TOE Security Functions. See SA.MODULAR-DESIGN for details. Directly threatened assets: D.JCS_DATA.

4.3 Organisational Security Policies

4.3.1 OSP.PROCESS-TOE: Identification of the TOE

An accurate identification must be established for the TOE. This requires that each instantiation of the TOE carries this identification.

4.3.2 OSP.KEY-CHANGE: Security Domain Keys Change

The Application Provider (AP) shall change its initial security domain keys (APSD) before any operation on its Security Domain.

4.3.3 OSP.SECURITY-DOMAINS: Security Domains

Security domains can be dynamically created, deleted and blocked during usage phase in post-issuance mode.

4.3.4 OSP.SECURE-BOX: Secure Box Border

Execution of untrusted native code shall be possible without any harm, manipulation, or influence on other parts of the TOE.

4.4 Assumptions

Note that the assumption A.DELETION is excluded. The Card Manager is part of the TOE and therefore the assumption is no longer relevant.

4.4.1 A.USE_DIAG: Usage of TOE’s Secure Communication Protocol by OE

It is assumed that the operational environment supports and uses the secure communication protocols offered by the TOE.

4.4.2 A.USE_KEYS: Protected Storage of Keys Outside of TOE

It is assumed that the keys which are stored outside the TOE and which are used for secure communication and authentication between Smart Card and terminals are protected for confidentiality and integrity in their own storage environment. This is especially true for D.APSD_KEYS, D.ISD_KEYS, and D.VASD_KEYS.

Info: This is to assume that the keys used in terminals or systems are correctly protected for confidentiality and integrity in their own environment, as the disclosure of such information which is shared with the TOE but is not under the TOE control, may compromise the security of the TOE.

4.4.3 A.PROCESS-SEC-IC: Protection during Packaging, Finishing and Personalisation

It is assumed that security procedures are used after delivery of the TOE by the TOE Manufacturer up to delivery to the end consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use). This means that the Phases after TOE Delivery are assumed to be protected appropriately.

The assets to be protected are: The information and material produced and/or processed by the Security IC Embedded Software Developer in Phase 1 and by the Composite Product Manufacturer can be grouped as follows:

1. the Security IC Embedded Software including specifications, implementation and related documentation,
2. pre-personalisation and personalisation data including specifications of formats and memory areas, test related data,
3. the User Data and related documentation, and
4. material for software development support

as long as they are not under the control of the TOE Manufacturer.

4.4.4 A.APPS-PROVIDER: Application Provider

The AP is a trusted actor that provides basic or secure applications. He is responsible for his security domain keys (APSD keys).

Info: An AP generally refers to the entity that issues the application. For instance it can be a financial institution for a payment application such as EMV or a transport operator for a transport application.

4.4.5 A.VERIFICATION-AUTHORITY: Verification Authority

The VA is a trusted actor who is able to guarantee and check the digital signature attached to a basic or secure application.

Info: As a consequence, it guarantees the success of the application validation upon loading.

5 Security Objectives

The Security Objectives for the TOE, for the operational environment and the security objectives rationale are described within the following sections. Only additional and refined items compared to the PP [6] are described.

5.1 Security Objectives for the TOE

5.1.1 Identification

5.1.1.1 OT.SID_MODULE: Subject Identification of Modules

The TOE shall uniquely identify every Module before granting it access to any service.

5.1.2 Execution

5.1.2.1 OT.SENSITIVE_RESULTS_INTEG: Sensitive Result

The TOE shall ensure that the sensitive results (com.nxp.id.jcopx.security.SensitiveResultX) of sensitive operations executed by applications through the Java Card API are protected in integrity specifically against physical attacks.

5.1.3 Services

5.1.3.1 OT.RNG: Random Numbers Generation

The TOE shall ensure the cryptographic quality of random number generation. For instance random numbers shall not be predictable and shall have sufficient entropy. The TOE shall ensure that no information about the produced random numbers is available to an attacker since they might be used for instance to generate cryptographic keys.

5.1.4 Applet Management

5.1.4.1 OT.APPLI-AUTH: Application Authentication

The card manager shall enforce the application security policies established by the card issuer by requiring application authentication during application loading on the card. This security objective is a refinement of the Security Objective O.LOAD from [6].

AppNote: Each application loaded onto the TOE has been signed by a VA. The VA will guarantee that the security policies established by the card issuer on applications are enforced. For example this authority (DAP) or a third party (Mandated DAP) can be present on the TOE as a Security Domain whose role is to verify each signature at application loading.

5.1.4.2 OT.DOMAIN-RIGHTS: Domain Rights

The Card issuer shall not get access or change personalized AP Security Domain keys which belong to the AP. Modification of a Security Domain keyset is restricted to the AP who owns the security domain.

AppNote: APs have a set of keys that allows them to establish a secure channel between them and the platform. These keys sets are not known by the TOE issuer. The security domain initial keys are changed before any operation on the SD (OE.KEY-CHANGE).

5.1.4.3 OT.COMM_AUTH: Communication Mutual Authentication

The TOE shall authenticate the origin of the card management requests that the card receives, and authenticate itself to the remote actor.

5.1.4.4 OT.COMM_INTEGRITY: Communication Request Integrity

The TOE shall verify the integrity of the card management requests that the card receives.

5.1.4.5 OT.COMM_CONFIDENTIALITY: Communication Request Confidentiality

The TOE shall be able to process card management requests containing encrypted data.

5.1.5 Card Management

5.1.5.1 OT.CARD-MANAGEMENT: Card Management

The TOE shall provide card management functionalities (loading, installation, extradition, deletion of applications and GP registry updates) in charge of the life cycle of the whole device and installed applications (applets). The card manager, the application with specific rights responsible for the administration of the smart card, shall control the access to card management functions. It shall also implement the card issuer's policy on card management.

The Security Objective from [6] for the environment OE.CARD-MANAGEMENT is listed as TOE Security Objective OT.CARD-MANAGEMENT for the TOE as the Card Manager belongs to the TOE for this evaluation. This security objective is a refinement for the Security Objectives O.INSTALL, O.LOAD, and O.DELETION from [6]. Thus, the following objectives are also covered:

- The TOE shall ensure that the installation of an applet performs as expected (See SA.INSTALL for details).
- The TOE shall ensure that the loading of a package into the card is secure.
- The TOE shall ensure that the deletion of a package from the TOE is secure.

AppNote: The card manager will be tightly connected in practice with the rest of the TOE, which in return shall very likely rely on the card manager for the effective enforcement of some of its security functions. The mechanism used to ensure authentication of the TOE issuer, that manages the TOE, or of the Service Providers owning a Security Domain with card management privileges is a secure channel. This channel will be used afterwards to protect commands exchanged with the TOE in confidentiality and integrity. The platform guarantees that only the ISD or the Service Providers owning a Security Domain with the appropriate privilege (Delegated Management) can manage the applications on the card associated with its Security Domain. This is done accordingly with the card issuer's policy on card management. The actor performing the operation must beforehand authenticate with the Security Domain. In the case of Delegated Management, the card management command will be associated with an electronic signature (GlobalPlatform token) verified by the ISD before execution.

The Security Objective from [6] for the environment OE.CARD-MANAGEMENT is listed as TOE Security Objective OT.CARD-MANAGEMENT for the TOE as the Card Manager belongs to the TOE for this evaluation. This security objective is a refinement for the Security Objectives O.INSTALL, O.LOAD, and O.DELETION from [6]. Thus, the following AppNote applicable to O.DELETION applies also:

- Usurpation of identity resulting from a malicious installation of an applet on the card may also be the result of perturbing the communication channel linking the CAD and the card. Even if the CAD is placed in a secure environment, the attacker may try to capture, duplicate, permute or modify the packages sent to the card. He may also try to send one of its own applications as if it came from the card issuer. Thus, this objective is intended to ensure the integrity and authenticity of loaded CAP files.

5.1.6 Smart Card Platform

5.1.6.1 OT.SCP.IC IC: Physical Protection

The SCP shall provide all IC security features against physical attacks. This security objective for the environment refers to the point (7) of the security aspect SA.SCP. AppNote: The Security Objectives from [6] for the environment OE.SCP.RECOVERY,

OE.SCP.SUPPORT, and OE.SCP.IC are listed as TOE Security Objectives (OT.SCP.RECOVERY, OT.SCP.SUPPORT, and OT.SCP.IC) for the TOE in this section as the Smart Card Platform belongs to the TOE for this evaluation.

5.1.6.2 OT.SCP.RECOVERY: SCP Recovery

If there is a loss of power, or if the smart card is withdrawn from the CAD while an operation is in progress, the SCP must allow the TOE to eventually complete the interrupted operation successfully, or recover to a consistent and secure state. This security objective for the environment refers to the security aspect SA.SCP.

AppNote: The Security Objectives from [6] for the environment OE.SCP.RECOVERY, OE.SCP.SUPPORT, and OE.SCP.IC are listed as TOE Security Objectives (OT.SCP.RECOVERY, OT.SCP.SUPPORT, and OT.SCP.IC) for the TOE in this section as the Smart Card Platform belongs to the TOE for this evaluation.

5.1.6.3 OT.SCP.SUPPORT: SCP Support

The SCP shall support the TSFs of the TOE. This security objective refers to the security aspects 2, 3, 4 and 5 of SA.SCP.

AppNote: The Security Objectives from [6] for the environment OE.SCP.RECOVERY, OE.SCP.SUPPORT, and OE.SCP.IC are listed as TOE Security Objectives (OT.SCP.RECOVERY, OT.SCP.SUPPORT, and OT.SCP.IC) for the TOE in this section as the Smart Card Platform belongs to the TOE for this evaluation.

5.1.6.4 OT.IDENTIFICATION: TOE identification

The TOE must provide means to store Initialization Data and Pre-personalization Data in its non-volatile memory. The Initialization Data (or parts of them) are used for TOE identification.

5.1.7 Secure Box

5.1.7.1 OT.SEC_BOX_FW: SecureBox firewall

The TOE shall provide separation between the Secure Box native code and the Java Card System. The separation shall comprise software execution and data access.

5.1.8 Configuration Module

5.1.8.1 OT.CARD-CONFIGURATION: Card Configuration

The TOE shall ensure that the customer can only configure customer configuration items and that NXP can configure customer and NXP configuration items.

5.1.8.2 OT.SECURE_LOAD_ACODE: Secure loading of the Additional Code

This objective is taken over from [8].

The Loader of the Initial TOE shall check an evidence of authenticity and integrity of the loaded Additional Code.

The Loader enforces that only the allowed version of the Additional Code can be loaded on the Initial TOE. The Loader shall forbid the loading of an Additional Code not intended to be assembled with the Initial TOE.

During the Load Phase of an Additional Code, the TOE shall remain secure.

5.1.8.3 OT.SECURE_ACTIVATION_ADDITIONAL_CODE: Secure activation of the Additional Code

This objective is taken over from [8] (O.Secure_AC_Activation) with editorial modification.

Activation of the Additional Code and update of the Identification Data shall be performed at the same time in an Atomic way.

All the operations needed for the code to be able to operate as in the Final TOE shall be completed before activation.

If the Atomic Activation is successful, then the resulting product is the Final TOE, otherwise (in case of interruption or incident which prevents the forming of the Final TOE), the Initial TOE shall remain in its initial state or fail secure.

5.1.8.4 OT.TOE_IDENTIFICATION: Secure identification of the TOE

This objective is taken over from [8] and further refined.

The Identification Data identifies the Initial TOE and Additional Code. The TOE provides means to store Identification Data in its non-volatile memory and guarantees the integrity of these data.

After Atomic Activation of the Additional Code, the Identification Data of the Final TOE allows identifications of Initial TOE and Additional Code or the Final TOE. The user shall be able to uniquely identify Initial TOE and Additional Code(s) or the final TOE which are embedded in the Final TOE.

5.2 Security Objectives for the Operational Environment

5.2.1 OE.APPS-PROVIDER: Application Provider

The AP shall be a trusted actor that provides applications. The AP is responsible for its security domain keys.

5.2.2 OE.VERIFICATION-AUTHORITY: Verification Authority

The VA should be a trusted actor who is able to verify bytecode of an application loaded on the card, guarantee and generate the digital signature attached to an application and ensure that its public key for verifying the application signature is on the TOE.

5.2.3 OE.PROCESS_SEC_IC: Protection during composite product manufacturing

Security procedures shall be used after TOE Delivery up to delivery to the end-consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use). This means that Phases after TOE Delivery up to the end of Phase 6 must be protected appropriately.

5.2.4 OE.KEY-CHANGE: Security Domain Key Change

The AP must change its security domain initial keys before any operation on it.

5.2.5 OE.SECURITY-DOMAINS: Security Domains

Security domains can be dynamically created, deleted and blocked during usage phase in post-issuance mode.

5.2.6 OE.USE_DIAG: Secure TOE communication protocols

Secure TOE communication protocols shall be supported and used by the environment.

5.2.7 OE.USE_KEYS: Protection of OPE keys

During the TOE usage, the terminal or system in interaction with the TOE shall ensure the protection (integrity and confidentiality) of their own keys by operational means and/or procedures.

5.3 Security Objectives Rationale

In this section it is proven that the security objectives described in Section 4 "Security Problem Definition (ASE_SPD)" can be traced for all aspects identified in the TOE-security environment and that they are suited to cover them. At least one security objective results from each assumption, OSP, and each threat. At least one threat, one OSP or assumption exists for each security objective.

Security Problem Definition	Security Objective
T.CONFID-APPLI-DATA	OT.CARD-MANAGEMENT OT.SCP.RECOVERY OT.SCP.SUPPORT OT.RNG OT.SECURE_LOAD_ACODE
T.CONFID-JCS-CODE	OT.CARD-MANAGEMENT OT.SECURE_LOAD_ACODE
T.CONFID-JCS-DATA	OT.CARD-MANAGEMENT OT.SCP.RECOVERY OT.SCP.SUPPORT OT.SID_MODULE OT.SECURE_LOAD_ACODE
T.INTEG-APPLI-CODE	OT.CARD-MANAGEMENT OT.SECURE_LOAD_ACODE
T.INTEG-APPLI-CODE.LOAD	OT.CARD-MANAGEMENT OT.APPLI-AUTH OT.SECURE_LOAD_ACODE
T.INTEG-APPLI-DATA[REFINED]	OT.CARD-MANAGEMENT OT.SCP.RECOVERY OT.SCP.SUPPORT OT.DOMAIN-RIGHTS OT.RNG OT.SECURE_LOAD_ACODE
T.INTEG-APPLI-DATA.LOAD	OT.CARD-MANAGEMENT OT.APPLI-AUTH OT.SECURE_LOAD_ACODE
T.INTEG-JCS-CODE	OT.CARD-MANAGEMENT OT.SECURE_LOAD_ACODE
T.INTEG-JCS-DATA	OT.CARD-MANAGEMENT OT.SCP.RECOVERY OT.SCP.SUPPORT OT.SID_MODULE OT.SECURE_LOAD_ACODE
T.SID.1	OT.CARD-MANAGEMENT OT.SID_MODULE
T.SID.2	OT.CARD-MANAGEMENT OT.SCP.RECOVERY OT.SCP.SUPPORT
T.MODULE_EXEC	OT.SCP.SUPPORT OT.SID_MODULE
T.RESOURCES	OT.CARD-MANAGEMENT OT.SCP.RECOVERY OT.SCP.SUPPORT

Security Problem Definition	Security Objective
T.UNAUTHORIZED_CARD_MNGT	OT.CARD-MANAGEMENT OT.DOMAIN-RIGHTS OT.COMM_AUTH OT.COMM_INTEGRITY OT.APPLI-AUTH
T.LIFE_CYCLE	OT.CARD-MANAGEMENT OT.DOMAIN-RIGHTS
T.COM_EXPLOIT	OT.COMM_AUTH OT.COMM_INTEGRITY OT.COMM_CONFIDENTIALITY
T.CONFIG	OT.CARD-CONFIGURATION
T.PHYSICAL	OT.SCP.IC OT.SENSITIVE_RESULTS_INTEG
T.OS_OPERATE	OT.OPERATE OT.SECURE_LOAD_ACODE OT.SECURE_ACTIVATION_ADDITIONAL_CODE
T.SEC_BOX_BORDER	OT.SEC_BOX_FW
T.RND	OT.RNG
T.MODULE_REPLACEMENT	OT.OPERATE OE.APPLET OT.SCP.SUPPORT OT.SID_MODULE
OSP.VERIFICATION	OT.CARD-MANAGEMENT OT.APPLI-AUTH
OSP.PROCESS-TOE	OT.IDENTIFICATION OT.TOE_IDENTIFICATION
OSP.KEY-CHANGE	OE.KEY-CHANGE
OSP.SECURITY-DOMAINS	OE.SECURITY-DOMAINS
OSP.SECURE-BOX	OT.SEC_BOX_FW
A.USE_DIAG	OE.USE_DIAG
A.USE_KEYS	OE.USE_KEYS
A.PROCESS-SEC-IC	OE.PROCESS_SEC_IC
A.APPS-PROVIDER	OE.APPS-PROVIDER
A.VERIFICATION-AUTHORITY	OE.VERIFICATION-AUTHORITY

5.3.1 Threats

5.3.1.1 Confidentiality

5.3.1.1.1 T.CONFID-APPLI-DATA

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions.
OT.SCP.RECOVERY	Intended to support the OT.OPERATE and OT.ALARM objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.SCP.SUPPORT	Intended to support the OT.OPERATE and OT.ALARM objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.RNG	Counters this threat by providing appropriate management of keys, PIN’s which are particular cases of an application’s sensitive data.
OT.SECURE_LOAD_ACODE	Counters this threat by checking the authenticity and integrity of the loaded Additional Code.

5.3.1.1.2 T.CONFID-JCS-CODE

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions.
OT.SECURE_LOAD_ACODE	Counters this threat by checking the authenticity and integrity of the loaded Additional Code.

5.3.1.1.3 T.CONFID-JCS-DATA

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions.
OT.SCP.RECOVERY	Intended to support the OT.OPERATE and OT.ALARM objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.SCP.SUPPORT	Intended to support the OT.OPERATE and OT.ALARM objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.SID_MODULE	Counters this threat by providing correct identification of applets.
OT.SECURE_LOAD_ACODE	Counters this threat by checking the authenticity and integrity of the loaded Additional Code.

5.3.1.2 Integrity

5.3.1.2.1 T.INTEG-APPLI-CODE

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions.
OT.SECURE_LOAD_ACODE	Counters this threat by checking the authenticity and integrity of the loaded Additional Code.

5.3.1.2.2 T.INTEG-APPLI-CODE.LOAD

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions such as the installation, update or deletion of applets.
OT.APPLI-AUTH	Counters this threat by ensuring that the loading of packages is done securely and thus preserves the integrity of packages code.
OT.SECURE_LOAD_ACODE	Counters this threat by checking the authenticity and integrity of the loaded Additional Code.

5.3.1.2.3 T.INTEG-APPLI-DATA[REFINED]

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions.
OT.SCP.RECOVERY	Intended to support the OT.OPERATE and OT.ALARM objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.SCP.SUPPORT	Intended to support the OT.OPERATE and OT.ALARM objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.DOMAIN-RIGHTS	Contributes to counter this threat by ensuring that personalization of the application by its associated security domain is only performed by the authorized AP.
OT.RNG	Counters this threat by providing appropriate management of keys, PINs which are particular cases of an application’s sensitive data.
OT.SECURE_LOAD_ACODE	Counters this threat by checking the authenticity and integrity of the loaded Additional Code.

5.3.1.2.4 T.INTEG-APPLI-DATA.LOAD

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions such as the installation, update or deletion of applets.

Objective	Rationale
OT.APPLI-AUTH	Counters this threat by ensuring that the loading of packages is done securely and thus preserves the integrity of packages code.
OT.SECURE_LOAD_ACODE	Counters this threat by checking the authenticity and integrity of the loaded Additional Code.

5.3.1.2.5 T.INTEG-JCS-CODE

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions.
OT.SECURE_LOAD_ACODE	Counters this threat by checking the authenticity and integrity of the loaded Additional Code.

5.3.1.2.6 T.INTEG-JCS-DATA

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions.
OT.SCP.RECOVERY	Intended to support the OT.OPERATE and OT.ALARM objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.SCP.SUPPORT	Intended to support the OT.OPERATE and OT.ALARM objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.SID_MODULE	Counters this threat by providing correct identification of applets.
OT.SECURE_LOAD_ACODE	Counters this threat by checking the authenticity and integrity of the loaded Additional Code.

5.3.1.3 Identity Usurpation

5.3.1.3.1 T.SID.1

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by preventing usurpation of identity resulting from a malicious installation of an applet on the card.
OT.SID_MODULE	Counters this threat by providing unique subject identification.

5.3.1.3.2 T.SID.2

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by ensuring that installing an applet has no effect on the state of other applets and thus can't change the TOE's attribution of privileged roles.

Objective	Rationale
OT.SCP.RECOVERY	Intended to support the OT.OPERATE and objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.SCP.SUPPORT	Intended to support the OT.OPERATE and objectives of the TOE, thus indirectly related to the threats that these latter objectives contribute to counter.

5.3.1.4 Unauthorized Excecution

5.3.1.4.1 T.MODULE_EXEC

Objective	Rationale
OT.SCP.SUPPORT	Intended to support the OT.OPERATE and OT.ALARM objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.SID_MODULE	Counters this threat by providing correct identification of Modules.

5.3.1.5 Denial of Service

5.3.1.5.1 T.RESOURCES

Objective	Rationale
OT.CARD-MANAGEMENT	Counters this threat by controlling the consumption of resources during installation and other card management operations.
OT.SCP.RECOVERY	Intended to support the OT.OPERATE and OT.RESOURCES objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.SCP.SUPPORT	Intended to support the OT.OPERATE and OT.RESOURCES objectives of the TOE, thus indirectly related to the threats that these objectives contribute to counter.

5.3.1.6 Card Management

5.3.1.6.1 T.UNAUTHORIZED_CARD_MNGT

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions such as the loading, installation, extradition or deletion of applets.
OT.DOMAIN-RIGHTS	Contributes to counter this threat by restricting the modification of an AP security domain keyset to the AP who owns it.
OT.COMM_AUTH	Contributes to counter this threat by preventing unauthorized users from initiating a malicious card management operation.

Objective	Rationale
OT.COMM_INTEGRITY	Contributes to counter this threat by protecting the integrity of the card management data while it is in transit to the TOE.
OT.APPLI-AUTH	Counters this threat by ensuring that the loading of a package is safe.

5.3.1.6.2 T.LIFE_CYCLE

Objective	Rationale
OT.CARD-MANAGEMENT	Contributes to counter this threat by controlling the access to card management functions such as the loading, installation, extradition or deletion of applets.
OT.DOMAIN-RIGHTS	Contributes to counter this threat by restricting the use of an AP security domain keysets, and thus the management of the applications related to this SD, to the AP who owns it.

5.3.1.6.3 T.COM_EXPLOIT

Objective	Rationale
OT.COMM_AUTH	Contributes to counter this threat by preventing unauthorized users from initiating a malicious card management operation.
OT.COMM_INTEGRITY	Contributes to counter this threat by protecting the integrity of the card management data while it is in transit to the TOE.
OT.COMM_CONFIDENTIALITY	Contributes to counter this threat by preventing from disclosing encrypted data transiting to the TOE.

5.3.1.7 Configuration Module

5.3.1.7.1 T.CONFIG

Objective	Rationale
OT.CARD-CONFIGURATION	Counters the threat by ensuring that the customer can only read and write customer configuration items using the Customer Configuration Token and NXP can read and write configuration items using the NXP Configuration Token. If access is disabled configuration items can not be read or written.

5.3.1.8 Miscellaneous

5.3.1.8.1 T.PHYSICAL

Objective	Rationale
OT.SCP.IC	Counters physical attacks. Physical protections rely on the underlying platform and are therefore an environmental issue.

Objective	Rationale
OT.SENSITIVE_RESULTS_INTEG	If the sensitive result is supported by the TOE, this threat is partially covered by the security objective OT.SENSITIVE_RE-SULTS_INTEG which ensures that sensitive results are protected against unauthorized modification by physical attacks.

5.3.1.9 Operating System

5.3.1.9.1 T.OS_OPERATE

Objective	Rationale
OT.OPERATE	Contributes to counter the threat by ensuring the correct continuation of operation of the TOE’s logical security functions. Security mechanisms have to be implemented to avoid fraudulent usage of the TOE, usage of certain memory regions, or usage of incorrect or unauthorized instructions or commands or sequence of commands. The security mechanisms must be designed to always put the TOE in a known and secure state.
OT.SECURE_LOAD_ACODE	Counters this threat by checking the authenticity and integrity of the loaded Additional Code.
OT.SECURE_ACTIVATION_ADDITIONAL_CODE	Counters this threat by atomically activating the Additional Code after all operations needed for the Additional Code to operate are completed. This prevents the TOE from executing incorrect or unauthorized instructions.

5.3.1.10 Secure Box

5.3.1.10.1 T.SEC_BOX_BORDER

Objective	Rationale
OT.SEC_BOX_FW	Counters the threat by ensuring that the native code and data in Secure Box is separated from the rest of the TOE. Due to this separation the native code in the Secure Box cannot harm the code and data outside the Secure Box.

5.3.1.11 Random Numbers

5.3.1.11.1 T.RND

Objective	Rationale
OT.RNG	Counters the threat by ensuring the cryptographic quality of random number generation. For instance random numbers shall not be predictable and shall have sufficient entropy. Furthermore, the TOE ensures that no information about the produced random numbers is available to an attacker.

5.3.1.12 Module replacement

5.3.1.12.1 T.MODULE_REPLACEMENT

Objective	Rationale
OT.OPERATE	Counters the threat by ensuring correct working order.
OE.APPLET	Contributes to counter this threat by ensuring that no native applets shall be loaded in post-issuance.
OT.SCP.SUPPORT	Intended to support the OT.OPERATE objective of the TOE, thus indirectly related to the threats that these objectives contribute to counter.
OT.SID_MODULE	Counters this threat by providing correct identification of Modules.

5.3.2 Organisational Security Policies

5.3.2.1 OSP.VERIFICATION

Objective	Rationale
OT.CARD-MANAGEMENT	Contributing to enforce the OSP by ensuring that the loading of a package into the card is safe.
OT.APPLI-AUTH	Contributing to enforce the OSP by ensuring that the loading of a package into the card is safe.

5.3.2.2 OSP.PROCESS-TOE

Objective	Rationale
OT.IDENTIFICATION	Enforces this organisational security policy by ensuring that the TOE can be uniquely identified.
OT.TOE_IDENTIFICATION	Enforces this organisational security policy by ensuring that the TOE can be uniquely identified after loading of Additional Code.

5.3.2.3 OSP.KEY-CHANGE

Objective	Rationale
OE.KEY-CHANGE	Enforces the OSP by ensuring that the initial keys of the security domain are changed before any operation on them are performed.

5.3.2.4 OSP.SECURITY-DOMAINS

Objective	Rationale
OE.SECURITY-DOMAINS	Enforces the OSP by dynamically create, delete, and block the security domain during usage phase in post-issuance mode.

5.3.2.5 OSP.SECURE-BOX

Objective	Rationale
OT.SEC_BOX_FW	Addresses directly this organizational security policy by ensuring that the native code and data in Secure Box is separated from the rest of the TOE. Due to this separation the native code in the Secure Box cannot harm the code and data outside the Secure Box.

5.3.3 Assumptions

5.3.3.1 A.USE_DIAG

Objective	Rationale
OE.USE_DIAG	Directly upholds this assumption.

5.3.3.2 A.USE_KEYS

Objective	Rationale
OE.USE_KEYS	Directly upholds this assumption.

5.3.3.3 A.PROCESS-SEC-IC

Objective	Rationale
OE.PROCESS_SEC_IC	Directly upholds this assumption.

5.3.3.4 A.APPS-PROVIDER

Objective	Rationale
OE.APPS-PROVIDER	Directly upholds this assumption.

5.3.3.5 A.VERIFICATION-AUTHORITY

Objective	Rationale
OE.VERIFICATION-AUTHORITY	Directly upholds this assumption.

6 Extended Components Definition (ASE_ECD)

The component FCS_RNG is taken over from the Java Card PP [6]. In addition following components are defined for the TOE.

6.1 Definition of Family ”Audit Data Storage (FAU_SAS)”

This section has been taken over from the certified (BSI-PP-0084-2014) Security IC Platform Protection profile [5].

To define the security functional requirements of the TOE an additional family (FAU_SAS) of the Class FAU (Security Audit) is defined here. This family describes the functional requirements for the storage of audit data. It has a more general approach than FAU_GEN, because it does not necessarily require the data to be generated by the TOE itself and because it does not give specific details of the content of the audit records.

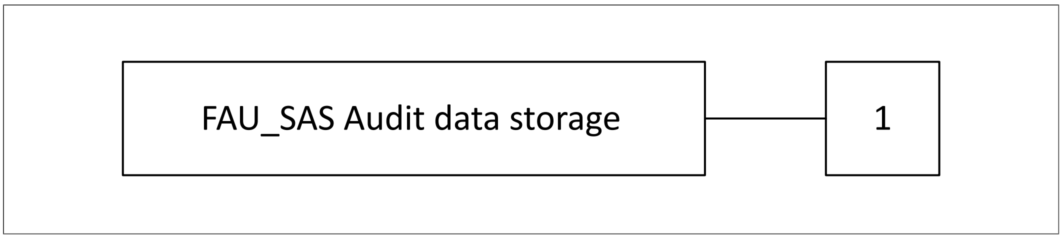
The family “Audit data storage (FAU_SAS)” is specified as follows.

FAU_SAS Audit data storage

Family behavior

This family defines functional requirements for the storage of audit data.

Component leveling



FAU_SAS.1	Requires the TOE to provide the possibility to store audit data.
Management:	FAU_SAS.1
There are no management activities foreseen.	
Audit:	FAU_SAS.1
There are no actions defined to be auditable.	
FAU_SAS.1	**Audit storage**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FAU_SAS.1.1	The TSF shall provide **[assignment: list of subjects]** with the capability to store **[assignment: list of audit information]** in the **[assignment: type of persistent memory]**.

6.2 Definition of Family ”TOE emanation (FPT_EMSEC)”

This section has been taken over from the certified (BSI-PP-0055) Protection Profile Machine Readable travel Document with "ICAO Application", Basic Access Control [7].

The sensitive family FPT_EMSEC (TOE Emanation) of the Class FPT (Protection of the TSF) is defined here to describe the IT security functional requirements of the TOE.

The TOE shall prevent attacks against the TOE and other secret data where the attack is based on external observable physical phenomena of the TOE. Examples of such attacks are evaluation of TOE's electromagnetic radiation, simple power analysis (SPA), differential power analysis (DPA), timing attacks, etc. This family describes the functional requirements for the limitation of intelligible emanations which are not directly addressed by any other component of CC part 2 [2].

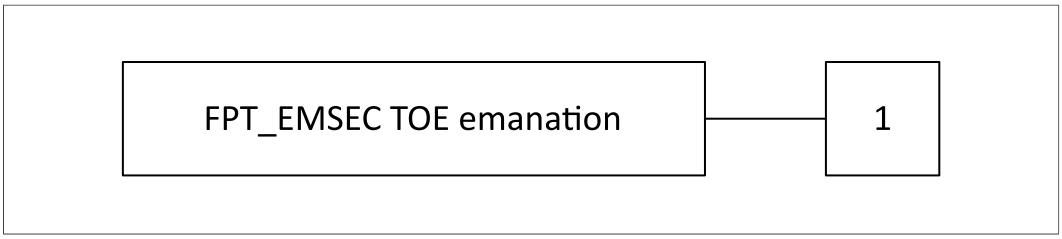
The family "TOE emanation (FPT_EMSEC)" is specified as follows.

FPT_EMSEC TOE emanation

Family behavior

This family defines requirements to mitigate intelligible emanations.

Component leveling



FPT_EMSEC.1 TOE emanation has two constituents:

FPT_EMSEC.1.1 Limit of emissions requires to not emit intelligible emissions enabling access to TSF data or user data.

FPT_EMSEC.1.2 Interface emanation requires not emit interface emanation enabling access to TSF data or user data.

Management: FPT_EMSEC.1

There are no management activities foreseen.

Audit: FPT_EMSEC.1

There are no actions defined to be auditable.

FPT_EMSEC.1 TOE Emanation

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_EMSEC.1.1 The TOE shall not emit **[assignment: types of emissions]** in excess of **[assignment: specified limits]** enabling access to **[assignment: list of types of TSF data]** and **[assignment: list of types of user data]**.

FPT_EMSEC.1.2 The TSF shall ensure **[assignment: type of users]** are unable to use the following interface **[assignment: type of connection]** to gain access to **[assignment: list of types of TSF data]** and **[assignment: list of types of user data]**.

7 Security Requirements (ASE_REQ)

This section defines the security requirements for the TOE.

The Security Functional Requirements for the JCOP component of the TOE implement all SFRs of the Java Card PP [6] however some are refined and some are added (see Conformance Claim Rationale).

In the following, only modified or added items regarding the Java Card PP [6] are described.

7.1 Definitions

7.1.1 Groups

Following groups are defined within this ST in addition to the groups defined in the PP [6].

Table 7. SFR Groups

Group	Description
Configuration (ConfG)	This group contains security requirements related to the configuration of the TOE.
Secure Box (SecBoxG)	This group contains security requirements to separate the native code executed in the Secure Box environment from the rest of the TOE.
Modular Design (ModDesG)	This group contains security requirements concerning the modular design of the TOE.
Further Security Functional Requirements	This group contains further security requirements not covered by the PP [6].

7.1.2 Subjects

Following subjects are defined within this ST in addition to the subjects defined by the PP [6].

Table 8. TOE Subjects

Subject	Description
S.SD	A GlobalPlatform Security Domain representing on the card a off-card entity. This entity can be the Issuer, an Application Provider, the Controlling Authority or the Verification Authority.
S.SBNativeCode	The third party native code executed via the Secure Box mechanism.
S.Customer	The subject that has the Customer Configuration Token.
S.NXP	The subject that has the NXP Configuration Token.
S.ConfigurationMechanism	On card entity which can read and write configuration items.

7.1.3 Objects

Following objects are defined within this ST in addition to the objects defined by the PP [6].

Table 9. TOE Objects

Objects	Description
O.SB_Content	The code and data elements of the native code library residing in the Secure Box. This includes SecureBox support functionality provided by the TOE, like functionality to write into FLASH memory or execute Crypto Library code.
O.NON_SB_Content	Any code and data elements not assigned to the native code library residing in the Secure Box.
O.SB_SFR	The pool of Special Function Registers assigned to be accessible by native code residing in the Secure Box.
O.NON_SB_SFR	All Special Function Registers which are not assigned to the Secure Box. Especially the Segment Tables to configure the MMU.
O.CODE_MODULE	Contains Applets, Java code, native code, native code of a library or a combination of those. The code of O.CODE_MODULE is called via a dedicated interface. The interface can be TOE internal (if the module implements functionality of the JavaCard API) or Public (if the Module implements functionality of the JCOPX API or is accessed via APDUs). Each O.CODE_MODULE has an unique internal AID.

7.1.4 Informations

Following informations are defined within this ST in addition to the informations defined by the PP [6].

Table 10. TOE Informations

Information	Description
I.MODULE_INVOCATION	Code execution flow when invoking code inside O.CODE_MODULE.

7.1.5 Security Attributes

Following security attributes are defined or redefined within this ST in addition to the security attributes defined by the PP [6].

Table 11. TOE Security attributes

Security Attributes	Description
Package AID	The AID of each package indicated in the export file or the internal AID of a Module.
Customer Configuration Token generation key	The customer key to generate tokens for product configuration.
NXP Configuration Token generation key	The NXP key to generate tokens for product configuration.
Configuration Token verification key	The keys to verify tokens for product configuration.
NXP Configuration Access	The NXP Configuration Access can either be enabled or disabled.
Customer Configuration Access	The Customer Configuration Access can either be enabled or disabled.

Table 11. TOE Security attributes*...continued*

Security Attributes	Description
access privilege	For each configuration item the access privilege attribute defines who (Customer and/or NXP) is allowed to read/write the item.
Key Set	Key Set for Secure Channel.
Security Level	Secure Communication Security Level defined in Section 10.6 of [17].
Secure Channel Protocol	Secure Channel Protocol version used.
Session Key	Secure Channel's session key.
Sequence Counter	Secure Channel Session's Sequence Counter.
Initial Chaining Vector (ICV)	Secure Channel Session's ICV.
Card Life Cycle	Defined in Section 5.1.1 of [17].
Privileges	Defined in Section 6.6.1 of [17].
Life-Cycle Status	Defined in Section 5.3.2 of [17].
CPU Mode	The execution mode of the CPU. Can be either user mode, system mode or firmware mode.
MMU Segment Table	Defines the memory areas which can be accessed for read / write operations or code execution if the CPU is in user mode. Further defines which of the Special Function Registers of the hardware can be accessed in user mode.
Special Function Registers	Special Function Registers allow to set operation modes of functional blocks of the hardware.
Module Presence	Presence of a particular O.CODE_MODULE inside the TOE with the values ”present” or ”not present”.
Resident Modules	The set of AIDs of the Modules already present in the card.

7.1.6 Operations

Following operations are defined within this ST in addition to the operations defined by the PP [6].

Table 12. TOE Operations

Operations	Description
OP.SB_ACCESS	Any read, write or execution access to a memory area.
OP.SB_ACCESS_SFR	Any read/write access to a Special Function Register.
OP.INVOKE_MODULE	Invocation of an O.CODE_MODULE. The invocation of the code is transparent to the user. In case O.CODE_MODULE has a TOE internal interface and is not present in the TOE, a secure state is preserved by throwing an exception or sending an appropriate error status word to the CAD.
OP.DELETE_MODULE	Deletion of a Module.

7.2 Security Functional Requirements

This section defines the security functional requirements for the TOE. The permitted operations (assignment, iteration, selection and refinement) of the SFRs taken from

Common Criteria [2] are printed in bold. Completed operations related to the PP [6] are additionally marked within [ ] where assignments are marked with the keyword "assignment".

7.2.1 COREG_LC Security Functional Requirements

The list of SFRs of this category are taken from the PP [6].

7.2.1.1 Firewall Policy

The following table provides the assignments and/or selections of related SFRs taken from the PP [6]:

SFR ID	Selection / Assignment text	Selection / Assignment value
FDP_IFF.1.3[JCVM]	[assignment: additional information flow control SFP rules]	no additional information flow control SFP rules
FDP_IFF.1.4[JCVM]	[assignment: rules, based on security attributes, that explicitly authorise information flows]	none
FDP_IFF.1.5[JCVM]	[assignment: rules, based on security attributes, that explicitly authorise information flows]	none

7.2.1.2 Application Programming Interface

The following SFRs are related to the Java Card API.

The following tables provide the assignments and/or selections of related SFRs taken from the PP [6]. Constants as defined in the Java Card API Spec [14] and the UGM [9] are used where appropriate.

Table 13. FCS_CKM.1 Cryptographic key generation

SFR ID	Selection / Assignment text	Selection / Assignment value
FCS_CKM.1.1	[assignment: cryptographic key generation algorithm]	JCOP RNG
	[assignment: cryptographic key sizes]	• DES Key lengths: – LENGTH_DES3_2KEY – LENGTH_DES3_3KEY • AES Key lengths: – LENGTH_AES_128 – LENGTH_AES_192 – LENGTH_AES_256
	[assignment: list of standards]	FCS_RNG.1 or FCS_RNG.1[HDT]
FCS_CKM.1.1[RSA][1][2]	[assignment: cryptographic key generation algorithm]	RSA key generation
	[assignment: cryptographic key sizes]	512, 736, 768, 896, 1024, 1280, 1536, 1984, 2048, 4096 bit and from 2000 bit to 4096 bit in one bit steps

Table 13. FCS_CKM.1 Cryptographic key generation...*continued*

SFR ID	Selection / Assignment text	Selection / Assignment value
	[assignment: list of standards]	FIPS 186-4
FCS_CKM.1.1[ECDSA][3][4]	[assignment: cryptographic key generation algorithm]	ECDSA (ECC over GF(p)) key generation
	[assignment: cryptographic key sizes]	160, 192, 224, 256, 320, 384, 512 and 521 bits
	[assignment: list of standards]	ISO/IEC 14888-3, ANSI X9.62 and FIPS 186-4
FCS_CKM.1.1[PUF]	[assignment: cryptographic key generation algorithm]	Key derivation function based on PUF
	[assignment: cryptographic key sizes]	128 bits
	[assignment: list of standards]	[12]

[1] FCS_CKM.1.1[RSA] is applicable only if the corresponding Module for the cryptographic operation is present in the TOE.
[2] The functionality of FCS_CKM.1.1[RSA] is provided by the Crypto Library [10].
[3] FCS_CKM.1.1[ECDSA] is applicable only if the corresponding Module for the cryptographic operation is present in the TOE.
[4] The functionality of FCS_CKM.1.1[ECDSA] is provided by the Crypto Library [10].

Table 14. FCS_CKM.4 Cryptographic key destruction

SFR ID	Selection / Assignment text	Selection / Assignment value
FCS_CKM.4.1[1]	[assignment: cryptographic key destruction method]	physically overwriting the keys in a randomized manner
	[assignment: list of standards]	none
FCS_CKM.4.1[PUF]	[assignment: cryptographic key destruction method]	flushing of key registers
	[assignment: list of standards]	none

[1] FCS_CKM.4 for ECC keys is applicable only if the corresponding Module for the cryptographic operation is present in the TOE.

Table 15. FCS_COP.1 Cryptographic operation

SFR ID	Selection / Assignment text	Selection / Assignment value
FCS_COP.1.1[PUF AES]	[assignment: list of cryptographic operations]	Data encryption and decryption
	[assignment: cryptographic algorithm]	AES in CBC mode
	[assignment: cryptographic key sizes]	128 bits

Table 15. FCS_COP.1 Cryptographic operation*...continued*

SFR ID	Selection / Assignment text	Selection / Assignment value
	[assignment: list of standards]	FIPS 197, NIST SP 800-38A
FCS_COP.1.1[PUF MAC]	[assignment: list of cryptographic operations]	AES CBC-MAC used for calculation of a PUF authentication
	[assignment: cryptographic algorithm]	AES in CBC-MAC mode
	[assignment: cryptographic key sizes]	128 bits
	[assignment: list of standards]	FIPS 197, NIST SP 800-38A and ISO/IEC 9797-1
FCS_COP.1.1[TripleDES]	[assignment: list of cryptographic operations]	Data encryption and decryption
	[assignment: cryptographic algorithm]	ALG_DES_CBC_ISO9797_M1 ALG_DES_CBC_ISO9797_M2 ALG_DES_CBC_NOPAD ALG_DES_ECB_ISO9797_M1 ALG_DES_ECB_ISO9797_M2 ALG_DES_ECB_NOPAD
	[assignment: cryptographic key sizes]	LENGTH_DES3_2KEY LENGTH_DES3_3KEY
	[assignment: list of standards]	Java Card API Spec [14]
FCS_COP.1.1[AES]	[assignment: list of cryptographic operations]	Data encryption and decryption
	[assignment: cryptographic algorithm]	ALG_AES_BLOCK_128_CBC_NOPAD ALG_AES_BLOCK_128_CBC_NOPAD_STANDARD ALG_AES_BLOCK_128_ECB_NOPAD ALG_AES_CBC_ISO9797_M1 ALG_AES_CBC_ISO9797_M2 ALG_ AES_CBC_ISO9797_M2_STANDARD ALG_AES_ECB_ISO9797_M1 ALG_AES_ECB_ISO9797_M2 ALG_AES_CTR
	[assignment: cryptographic key sizes]	LENGTH_AES_128 LENGTH_AES_192 LENGTH_AES_256
	[assignment: list of standards]	Java Card API Spec [14] and JCOPX API [9]
FCS_COP.1.1[RSACipher]	[assignment: list of cryptographic operations]	Data encryption and decryption

Table 15. FCS_COP.1 Cryptographic operation*...continued*

SFR ID	Selection / Assignment text	Selection / Assignment value
	[assignment: cryptographic algorithm]	ALG_RSA_NOPAD ALG_RSA_PKCS1 ALG_RSA_PKCS1_OAEP
	[assignment: cryptographic key sizes]	LENGTH_RSA_2048 LENGTH_RSA_4096 and from 2000 bit to 4096 bit in one bit steps
	[assignment: list of standards]	Java Card API Spec [14] and for the one bit step range see JCOPX API [9]
FCS_COP.1.1 [ECDH_P1363][1]	[assignment: list of cryptographic operations]	Diffie-Hellman Key Agreement
	[assignment: cryptographic algorithm]	ALG_EC_SVDP_DH ALG_EC_SVDP_DH_KDF ALG_EC_SVDP_DH_PLAIN ALG_EC_SVDP_DHC ALG_EC_SVDP_DHC_KDF ALG_EC_SVDP_DHC_PLAIN ALG_EC_SVDP_DH_PLAIN_XY
	[assignment: cryptographic key sizes]	LENGTH_EC_FP_160 LENGTH_EC_FP_192 LENGTH_EC_FP_224 LENGTH_EC_FP_256 LENGTH_EC_FP_320 LENGTH_EC_FP_384 LENGTH_EC_FP_521 from 160 bit to 521 bit in 1 bit steps
	[assignment: list of standards]	Java Card API Spec [14] and JCOPX API [9]
FCS_COP.1.1[DESMAC]	[assignment: list of cryptographic operations]	MAC generation and verification
	[assignment: cryptographic algorithm]	Triple-DES in outer CBC for Mode: ALG_DES_MAC4_ISO9797_1_M1_ALG3 ALG_DES_MAC4_ISO9797_1_M2_ALG3 ALG_DES_MAC4_ISO9797_M1 ALG_DES_MAC4_ISO9797_M2 ALG_DES_MAC8_ISO9797_1_M1_ALG3 ALG_DES_MAC8_ISO9797_1_M2_ALG3 ALG_DES_MAC8_ISO9797_M1 ALG_DES_MAC8_ISO9797_M2 ALG_DES_MAC8_NOPAD
	[assignment: cryptographic key sizes]	LENGTH_DES3_2KEY LENGTH_DES3_3KEY

Table 15. FCS_COP.1 Cryptographic operation*...continued*

SFR ID	Selection / Assignment text	Selection / Assignment value
	[assignment: list of standards]	Java Card API Spec [14]
FCS_COP.1.1[AESMAC]	[assignment: list of cryptographic operations]	MAC generation and verification
	[assignment: cryptographic algorithm]	AES in CBC Mode: ALG_AES_MAC_128_NOPAD ALG_AES_MAC_128_ISO9797_1_M2_ALG3
	[assignment: cryptographic key sizes]	LENGTH_AES_128 LENGTH_AES_192 LENGTH_AES_256
	[assignment: list of standards]	Java Card API Spec [14]
FCS_COP.1.1 [RSASignaturePKCS1]	[assignment: list of cryptographic operations]	Digital signature generation and verification
	[assignment: cryptographic algorithm]	ALG_RSA_SHA_ISO9796[2] ALG_RSA_SHA_ISO9796_MR[2] ALG_RSA_SHA_PKCS1[2] ALG_RSA_SHA_PKCS1_PSS[2] ALG_RSA_SHA_224_PKCS1 ALG_RSA_SHA_224_PKCS1_PSS ALG_RSA_SHA_256_PKCS1 ALG_RSA_SHA_256_PKCS1_PSS ALG_RSA_SHA_384_PKCS1 ALG_RSA_SHA_384_PKCS1_PSS ALG_RSA_SHA_512_PKCS1 ALG_RSA_SHA_512_PKCS1_PSS ALG_RSA_SHA_ISO9796[2] ALG_RSA_SHA_256_ISO9796 or SIG_CIPHER_RSA in combination with MessageDigest.ALG_SHA[2] MessageDigest.ALG_SHA_224 MessageDigest.ALG_SHA_256 MessageDigest.ALG_SHA_384 MessageDigest.ALG_SHA_512 and in combination with Cipher.PAD_PKCS1_PSS Cipher.PAD_ISO9796 Cipher.PAD_ISO9796_MR
	[assignment: cryptographic key sizes]	LENGTH_RSA_2048 LENGTH_RSA_4096 and from 2000 bit to 4096 bit in one bit steps
	[assignment: list of standards]	Java Card API Spec [14] and for the one bit step range see JCOPX API [9]

Table 15. FCS_COP.1 Cryptographic operation*...continued*

SFR ID	Selection / Assignment text	Selection / Assignment value
FCS_COP.1.1 [ECSignature][1]	[assignment: list of cryptographic operations]	Digital signature generation and verification
	[assignment: cryptographic algorithm]	ALG_ECDSA_SHA[2] ALG_ECDSA_SHA_224 ALG_ECDSA_SHA_256 ALG_ECDSA_SHA_384 ALG_ECDSA_SHA_512 or SIG_CIPHER_ECDSA or SIG_CIPHER_ECDSA_PLAIN in combination with MessageDigest.ALG_SHA[2] MessageDigest.ALG_SHA_224 MessageDigest.ALG_SHA_256 MessageDigest.ALG_SHA_384 MessageDigest.ALG_SHA_512
	[assignment: cryptographic key sizes]	LENGTH_EC_FP_160 LENGTH_EC_FP_192 LENGTH_EC_FP_224 LENGTH_EC_FP_256 LENGTH_EC_FP_320 LENGTH_EC_FP_384 LENGTH_EC_FP_521 from 160 bit to 521 bit in 1 bit steps
	[assignment: list of standards]	Java Card API Spec [14] and JCOPX API [9]
FCS_COP.1.1 [ModMath][1]	[assignment: list of cryptographic operations]	Secure modular arithmetic: • addition • subtraction • reduction • multiplication
	[assignment: cryptographic algorithm]	None
	[assignment: cryptographic key sizes]	None
	[assignment: list of standards]	JCOPX API [9]
FCS_COP.1.1[SHA]	[assignment: list of cryptographic operations]	Secure hash computation
	[assignment: cryptographic algorithm]	ALG_SHA[2] ALG_SHA_224 ALG_SHA_256 ALG_SHA_384 ALG_SHA_512

Table 15. FCS_COP.1 Cryptographic operation*...continued*

SFR ID	Selection / Assignment text	Selection / Assignment value
	[assignment: cryptographic key sizes]	LENGTH_SHA LENGTH_SHA_224 LENGTH_SHA_256 LENGTH_SHA_384 LENGTH_SHA_512
	[assignment: list of standards]	Java Card API Spec [14] and JCOPX API [9]
FCS_COP.1.1 [AES_CMAC]	[assignment: list of cryptographic operations]	AES CMAC generation and verification
	[assignment: cryptographic algorithm]	ALG_AES_CMAC8 ALG_AES_CMAC16 ALG_AES_CMAC16_STANDARD ALG_AES_CMAC_128
	[assignment: cryptographic key sizes]	LENGTH_AES_128 LENGTH_AES_192 LENGTH_AES_256
	[assignment: list of standards]	Java Card API Spec [14] and JCOPX API [9]
FCS_COP.1.1[DAP]	[assignment: list of cryptographic operations]	Verification of the DAP signature attached to Executable Load Applications
	[assignment: cryptographic algorithm]	ALG_ECDSA_SHA_256[1] ALG_RSA_SHA_PKCS1[3] ALG_AES_CMAC16
	[assignment: cryptographic key sizes]	LENGTH_EC_FP_256 LENGTH_RSA_1024 LENGTH_AES_128 LENGTH_AES_192 LENGTH_AES_256
	[assignment: list of standards]	Global Platform Specifications [17], [18], [19]

[1] Applicable only if the corresponding module for the cryptographic operation is present in the TOE.

[2] Due to mathematical weakness only resistant against AVA_VAN.5 for temporary data (e.g. as used for generating session keys), but not if repeatedly applied to the same input data.

[3] Available only in the IOT Full variant of the TOE and not on the IOT Reduced variant. See Section 1.3.3 "TOE Reduced Feature Set" for details.

For resistance against attackers with High Attack Potential the user should always refer to the guidance given by the Certification Body in the jurisdiction. The website www.keylength.com provides a good reference to recommended key lengths.

Table 16. FCS_RNG.1 Random number generation

SFR ID	Selection / Assignment text	Selection / Assignment value
FCS_RNG.1.1[1]	[selection: physical, non-physical true, deterministic, hybrid physical, hybrid deterministic]	deterministic
	[assignment: list of security capabilities]	• (DRG.3.1) If initialized with a random seed using a PTRNG of class PTG.2 (as defined in [21]) as random source, the internal state of the RNG shall have at least 256 bit of entropy • (DRG.3.2) The RNG provides forward secrecy (as defined in [21]) • (DRG.3.3) The RNG provides backward secrecy even if the current internal state is known (as defined in [21])
FCS_RNG.1.2	[assignment: a defined quality metric]	• (DRG.3.4) The RNG, initialized with a random seed using a PTRNG of class PTG.2 (as defined in [21]) as random source, generates output for which for AES-mode 2^{48} and for TDEA-mode 2^{35} strings of bit length 128 are mutually different with probability at least $1 - 2^{-24}$ • (DRG.3.5) Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A (as defined in [21])

[1] The functionality of FCS_RNG.1.1 is provided by the Crypto Library [10].

Table 17. FCS_RNG.1[HDT] Random number generation

SFR ID	Selection / Assignment text	Selection / Assignment value
FCS_RNG.1.1[HDT][1]	[selection: physical, non-physical true, deterministic, hybrid physical, hybrid deterministic]	hybrid deterministic

Table 17. FCS_RNG.1[HDT] Random number generation*...continued*

SFR ID	Selection / Assignment text	Selection / Assignment value
	[assignment: list of security capabilities]	• (DRG.4.1) The internal state of the RNG shall use PTRNG of class PTG.2 (as defined in [21]) as random source • (DRG.4.2) The RNG provides forward secrecy (as defined in [21]) • (DRG.4.3) The RNG provides backward secrecy even if the current internal state is known (as defined in [21]) • (DRG.4.4) The RNG provides enhanced forward secrecy on demand (as defined in [21]) • (DRG.4.5) The internal state of the RNG is seeded by an PTRNG of class PTG.2 (as defined in [21])
FCS_RNG.1.2[HDT]	[assignment: a defined quality metric]	• (DRG.4.6) The RNG generates output for which for AES-mode 2^{48} and for TDEA-mode 2^{35} strings of bit length 128 are mutually different with probability at least $1 - 2^{-24}$ • (DRG.4.7) Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A (as defined in [21])

[1] The functionality of FCS_RNG.1.1[HDT] is provided by the Crypto Library [10].

7.2.1.3 Card Security Management

The following table provides the assignments and/or selections of related SFRs taken from the PP [6].

SFR ID	Selection / Assignment text	Selection / Assignment value
FAU_ARP.1.1	[assignment: list of other actions]	Response with error code to S.CAD
FDP_SDI.2.1[DATA]	[assignment: integrity errors]	Integrity errors
	[assignment: user data attributes]	Integrity protected data
FDP_SDI.2.2[DATA]	[assignment: action to be taken]	Perform the action defined in FAU_ ARP.1
FPR_UNO.1.1	[assignment: list of users and/or subjects]	All users
	[assignment: list of operations]	All operations
	[assignment: list of objects]	D.APP_KEYs, D.PIN, D.Crypto

SFR ID	Selection / Assignment text	Selection / Assignment value
	[assignment: list of protected users and/or subjects]	Another user
FPT_TDC.1.2	[assignment: list of interpretation rules to be applied by the TSF]	ISO 7816-6 EMV specification

7.2.1.4 AID Management

The following table provides the assignments and/or selections of related SFRs taken from the PP [6].

SFR ID	Selection / Assignment text	Selection / Assignment value
FIA_USB.1.2[AID]	[assignment: rules for the initial association of attributes]	Each uploaded package is associated with an unique Package AID
FIA_USB.1.3[AID]	[assignment: rules for the changing of attributes]	The initially assigned Package AID is unchangeable

7.2.2 INSTG Security Functional Requirements

The following table provides the assignments and/or selections of related SFRs taken from the PP [6].

Note that the SFR FDP_ITC.2[INSTALLER] has been refined and is now part of the card management SFRs (FDP_ITC.2[CCM]) in Section 7.2.6 "CarG Security Functional Requirements").

SFR ID	Selection / Assignment text	Selection / Assignment value
FPT_RCV.3.1 [INSTALLER]	[assignment: list of failures/service discontinuities]	None
FPT_RCV.3.2 [INSTALLER]	[assignment: list of failures/service discontinuities]	A failure during load/installation of a package/applet and deletion of a package/applet/object
FPT_RCV.3.3 [INSTALLER]	[assignment: quantification]	0%

7.2.3 ADELG Security Functional Requirements

The ADELG SFRs from the PP [6] are refined and replaced by the following SFRs.

7.2.3.1 FDP_ACC.2[ADEL] Complete access control (ADEL)

Hierarchical to:	FDP_ACC.1 Subset access control
Dependencies:	FDP_ACF.1 Security attribute based access control

FDP_ACC.2.1 [ADEL]

The TSF shall enforce the **[assignment: ADEL access control SFP]** on **[assignment: S.ADEL, S.JCRE, S.JCVM, O.JAVAOBJECT, O.APPLET, O.CODE_PKG and O.CODE_MODULE]** and all operations among subjects and objects covered by the SFP.

FDP_ACC.2.2 [ADEL]

The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

Refinement

The operations involved in the policy are:

- OP.DELETE_APPLET,
- OP.DELETE_PCKG(O.CODE_PKG, ...),
- OP.DELETE_PCKG_APPLET(O.CODE_PKG, ...),
- OP.DELETE_MODULE.

7.2.3.2 FDP_ACF.1[ADEL] Security attribute based access control (ADEL)

Hierarchical to:

No other components.

Dependencies:

FDP_ACC.1 Subset access control, FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1 [ADEL]

The TSF shall enforce the **[assignment: ADEL access control SFP]** to objects based on the following **[assignment:**

Subject/Object	Security Attributes
S.JCVM	**Active Applets**
S.JCRE	**Selected Applet Context**, **Registered Applets**, **Resident Packages**, **Resident Modules**
O.CODE_PKG	**Package AID**, **Dependent Package AID**, **Static References**
O.APPLET	**Applet Selection Status**
O.JAVAOBJECT	**Owner**
O.CODE_MODULE	**Module Presence**

]

FDP_ACF.1.2 [ADEL]

The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **[assignment:**

In the context of this policy, an object O is reachable if and only one of the following conditions hold:

1. **the owner of O is a registered applet instance A (O is reachable from A),**
2. **a static field of a resident package P contains a reference to O (O is reachable from P),**

3. a static field of a resident Module M contains a reference to O (O is reachable from M),
4. there exists a valid remote reference to O (O is remote reachable),
5. there exists an object O' that is reachable according to either (1) or (2) or (3) or (4) above and O' contains a reference to O (the reachability status of O is that of O').

The following access control rules determine when an operation among controlled subjects and objects is allowed by the policy:

- R.JAVA.14 ([16], §11.3.4.1, Applet Instance Deletion): S.ADEL may perform OP.DELETE_APPLET upon an O.APPLET only if,
  1. S.ADEL is currently selected,
  2. there is no instance in the context of O.APPLET that is active in any logical channel and
  3. there is no O.JAVAOBJECT owned by O.APPLET such that either O.JAVAOBJECT is reachable from an applet instance distinct from O.APPLET, or O.JAVAOBJECT is reachable from a package P or Module M, or ([16], §8.5) O.JAVAOBJECT is remote reachable.
- R.JAVA.15 ([16], §11.3.4.1, Multiple Applet Instance Deletion): S.ADEL may perform OP.DELETE_APPLET upon several O.APPLET only if,
  1. S.ADEL is currently selected,
  2. there is no instance of any of the O.APPLET being deleted that is active in any logical channel and
  3. there is no O.JAVAOBJECT owned by any of the O.APPLET being deleted such that either O.JAVAOBJECT is reachable from an applet instance distinct from any of those O.APPLET, or O.JAVAOBJECT is reachable from a package P or Module M, or ([16], §8.5) O.JAVAOBJECT is remote reachable.
- R.JAVA.16 ([16], §11.3.4.2, Applet/Library Package Deletion): S.ADEL may perform OP.DELETE_PCKG(O.CODE_PKG, ...) upon an O.CODE_PKG only if,
  1. S.ADEL is currently selected,
  2. no reachable O.JAVAOBJECT, from a package or Module distinct from O.CODE_PKG that is an instance of a class that belongs to O.CODE_PKG, exists on the card and
  3. there is no resident package or resident Module on the card that depends on O.CODE_PKG.
- R.JAVA.17 ([16], §11.3.4.3, Applet Package and Contained Instances Deletion): S.ADEL may perform OP.DELETE_PCKG_APPLET(O.CODE_PKG, ...) upon an O.CODE_PKG only if,
  1. S.ADEL is currently selected,

2. **no reachable O.JAVAOBJECT, from a package or Module distinct from O.CODE_PKG, which is an instance of a class that belongs to O.CODE_PKG, exists on the card,**
3. **there is no package or Module loaded on the card that depends on O.CODE_PKG, and**
4. **for every O.APPLET of those being deleted it holds that:**
    a. **there is no instance in the context of O.APPLET that is active in any logical channel and**
    b. **there is no O.JAVAOBJECT owned by O.APPLET such that either O.JAVAOBJECT is reachable from an applet instance not being deleted, or O.JAVAOBJECT is reachable from a package or Module not being deleted, or ([16], §8.5) O.JAVAOBJECT is remote reachable.**

- **Module deletion: If a Module contains Java code then S.ADEL may perform OP.DELETE_MODULE upon a Module only if the following rules are satisfied:**
    1. **R.JAVA.14, if the Module contains an Applet Instance O.APPLET,**
    2. **R.JAVA.15, if the Module contains Multiple Applet Instances of O.APPLET,**
    3. **R.JAVA.16, if the Module contains an Applet/Library Package O.CODE_PKG and**
    4. **R.JAVA.17, if the Module contains an Applet Package O.CODE_PKG and Contained Instances O.APPLET.**

]

FDP_ACF.1.3 [ADEL] The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **[assignment: deletion of O.CODE_MODULE with a TOE internal interface is allowed even if other Resident Packages or other Resident Modules depend on it]**.

FDP_ACF.1.4 [ADEL] The TSF shall explicitly deny access of subjects to objects based on the following additional rules: **[assignment: any subject but S.ADEL to O.CODE_PKG, O.APPLET or O.CODE_MODULE for the purpose of deleting them from the card.**

7.2.3.3 FDP_RIP.1[ADEL] Subset residual information protection (ADEL)

Hierarchical-To: No other components.

Dependencies: No dependencies.

FDP_RIP.1.1[ADEL]: The TSF shall ensure that any previous information content of a resource is made unavailable upon the **[selection: deallocation of the resource from]** the following objects: **[assignment: applet instances and/or packages and/or Modules when one of the deletion operations in FDP_ACC.2[ADEL] is performed on them]**.

7.2.3.4 FMT_MSA.1[ADEL] Management of security attributes (ADEL)

This SFR is taken from the PP [6] without modification.

7.2.3.5 FMT_MSA.3[ADEL] Static attribute initialisation (ADEL)

This SFR is taken from the PP [6] without modification.

7.2.3.6 FMT_SMF.1[ADEL] Specification of Management Functions (ADEL)

Hierarchical-To: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1[ADEL]: The TSF shall be capable of performing the following management functions: **[assignment: modify the list of registered applets' AIDs, the Resident Packages and Resident Modules]**.

7.2.3.7 FMT_SMR.1[ADEL] Security roles (ADEL)

This SFR is taken from the PP [6] without modification.

7.2.3.8 FPT_FLS.1[ADEL] Failure with preservation of secure state (ADEL)

Hierarchical-To: No other components.

Dependencies: No dependencies.

FPT_FLS.1.1[ADEL]: The TSF shall preserve a secure state when the following types of failures occur: **[assignment: the applet deletion manager fails to delete a package/ applet as described in [16], §11.3.4 or it fails to delete a Module]**.

7.2.4 RMIG Security Functional Requirements

Not used in this ST because RMI is optional in the PP [6] and the TOE does not support RMI.

7.2.5 ODELG Security Functional Requirements

This group is taken from the PP [6] without modification.

7.2.6 CarG Security Functional Requirements

The card management SFRs from the PP [6] are refined and replaced by the following SFRs.

7.2.6.1 FDP_UIT.1[CCM] Data exchange integrity (CCM)

(refines FDP_UIT.1/CM)

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path].

FDP_UIT.1.1[CCM] The TSF shall enforce the **[assignment: Secure Channel Protocol information flow control policy and the Security Domain access control policy]** to **[selection: receive]** user data in a manner protected from **[selection: modification, deletion, insertion and replay]** errors.

FDP_UIT.1.2[CCM] The TSF shall be able to determine on receipt of user data, whether **[selection: modification, deletion, insertion, replay]** has occurred.

7.2.6.2 FDP_ROL.1[CCM] Basic rollback (CCM)

(added)

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control].

FDP_ROL.1.1[CCM] The TSF shall enforce **[assignment: Security Domain access control policy]** to permit the rollback of the **[assignment: installation operation]** on the **[assignment: executable files and application instances]**.

FDP_ROL.1.2[CCM] The TSF shall permit operations to be rolled back within the **[assignment: boundaries of available memory before the card content management function started]**.

7.2.6.3 FDP_ITC.2[CCM] Import of user data with security attributes (CCM)

(replaces FDP_ITC.2/INSTALLER)

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] FPT_TDC.1 Inter-TSF basic TSF data consistency.

FDP_ITC.2.1[CCM] The TSF shall enforce the **[assignment: Security Domain access control policy and the Secure Channel Protocol information flow policy]** when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.2.2[CCM] The TSF shall use the security attributes associated with the imported user data

FDP_ITC.2.3[CCM] The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.

FDP_ITC.2.4[CCM] The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.

FDP_ITC.2.5[CCM] The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: **[assignment: Package loading is allowed only if, for each dependent package, its AID attribute is equal to a resident package AID attribute, the major (minor) Version attribute associated to the dependent package is lesser than or equal to the major (minor) Version attribute associated to the resident package ([15], §4.5.2)]**.

7.2.6.4 FPT_FLS.1[CCM] Failure with preservation of secure state (CCM)

(added)

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_FLS.1.1[CCM] The TSF shall preserve a secure state when the following types of failures occur: **[assignment: the Security Domain fails to load/install an Executable File/application instance as described in [15], Section 11.1.5]**.

7.2.6.5 FDP_ACC.1[SD] Subset access control (SD)

(added)

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control.

FDP_ACC.1.1 [SD] The TSF shall enforce the **[assignment: Security Domain access control policy]** on **[assignment:**

- **Subjects: S.INSTALLER, S.ADEL, S.CAD (from** [6]**) and S.SD,**
- **Objects: Delegation Token, DAP Block and Load File,**
- **Operations: GlobalPlatform's card content management APDU commands and API methods**

].

7.2.6.6 FDP_ACF.1[SD] Security attribute based access control (SD)

(added)

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control, FMT_MSA.3 Static attribute initialisation.

FDP_ACF.1.1[SD] The TSF shall enforce the **[assignment: Security Domain access control policy]** to objects based on the following **[assignment:**

- **Subjects:**
  - **S.INSTALLER, defined in [6] and represented by the GlobalPlatform Environment (OPEN) on the card, the Card Life Cycle attributes (defined in Section 5.1.1 of [17]),**
  - **S.ADEL, also defined in [6] and represented by the GlobalPlatform Environment (OPEN) on the card,**
  - **S.SD receiving the Card Content Management commands (through APDUs or APIs) with a set of Privileges (defined in Section 6.6.1 of [17]), a Life-cycle Status (defined in Section 5.3.2 of [17]) and a Secure Communication Security Level (defined in Section 10.6 of [17]),**
  - **S.CAD, defined in [6], the off-card entity that communicates with the S.INSTALLER and S.ADEL through S.SD.**
- **Objects:**
  - **The Delegation Token, in case of Delegated Management operations, with the attributes Present or Not Present,**
  - **The DAP Block, in case of application loading, with the attributes Present or Not Present,**
  - **The Load File or Executable File, in case of application loading, installation, extradition or registry update, with a set of intended privileges and its targeted associated SD AID.**
- **Mapping subjects/objects to security attributes:**
  - **S.INSTALLER: Security Level, Card Life Cycle, Life-cycle Status, Privileges, Resident Packages, Registered Applets,**
  - **S.ADEL: Active Applets, Static References, Card Life Cycle, Life-cycle Status, Privileges, Applet Selection Status, Security Level,**
  - **S.SD: Privileges, Life-cycle Status, Security Level,**
  - **S.CAD: Security Level**

].

FDP_ACF.1.2[SD] The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **[assignment: Runtime behavior rules defined by GlobalPlatform for:**

- **loading (Section 9.3.5 of [17])**
- **installation (Section 9.3.6 of [17])**
- **extradition (Section 9.4.1 of [17])**
- **registry update (Section 9.4.2 of [17])**

- **content removal (Section 9.5 of [17])**

].

FDP_ACF.1.3[SD] The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **[assignment: none]**

FDP_ACF.1.4[SD] The TSF shall explicitly deny access of subjects to objects based on the following additional rules:**[assignment: when at least one of the rules defined by GlobalPlatform does not hold]**

7.2.6.7 FMT_MSA.1[SD] Management of security attributes (SD)

(added)

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control], FMT_SMR.1 Security roles, FMT_SMF.1 Specification of Management Functions.

FMT_MSA.1.1[SD] The TSF shall enforce the **[assignment: Security Domain access control policy]** to restrict the ability to **[selection: modify]** the security attributes **[assignment:**

- **Card Life Cycle,**
- **Privileges,**
- **Life-cycle Status,**
- **Security Level**

] to **[assignment: the Security Domain and the application instance itself]**.

7.2.6.8 FMT_MSA.3[SD] Static attribute initialisation (SD)

(added)

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes, FMT_SMR.1 Security roles.

FMT_MSA.3.1[SD] The TSF shall enforce the **[assignment: Security Domain access control policy]** to provide **[selection: restrictive]** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2[SD] The TSF shall allow the **[assignment: Card Issuer or the Application Provider]** to specify alternative initial values to override the default values when an object or information is created.

Refinement	Alternative initial values shall be at least as restrictive as the default values defined in FMT_MSA.3.1[SD].

7.2.6.9 FMT_SMF.1[SD] Specification of Management Functions (SD)

(refines FMT_SMF.1/CM)

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1[SD] The TSF shall be capable of performing the following management functions: **[assignment:**

Management functions specified in GlobalPlatform specifications:

- **card locking (Section 9.6.3 of [17]),**
- **application locking and unlocking (Section 9.6.2 of [17]),**
- **card termination (Section 9.6.4 of [17]),**
- **card status interrogation (Section 9.6.6 of [17]),**
- **application status interrogation (Section 9.6.5 of [17])**

].

7.2.6.10 FMT_SMR.1[SD] Security roles (SD)

(refines FMT_SMR.1/CM)

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1[SD] The TSF shall maintain the roles **[assignment: ISD, SSD]**.

FMT_SMR.1.2[SD] The TSF shall be able to associate users with roles.

7.2.6.11 FCO_NRO.2[SC] Enforced proof of origin (SC)

(refines FCO_NRO.2/CM)

Hierarchical to: FCO_NRO.1 Selective proof of origin.

Dependencies: FIA_UID.1 Timing of identification.

FCO_NRO.2.1[SC] The TSF shall enforce the generation of evidence of origin for transmitted **[assignment: Executable load files]** at all times.

FCO_NRO.2.2[SC] The TSF shall be able to relate the **[assignment: DAP Block]** of the originator of the information, and the **[assignment: identity]** of the information to which the evidence applies.

FCO_NRO.2.3[SC] The TSF shall provide a capability to verify the evidence of origin of information to **[selection: originator]** given **[assignment: at the time the Executable load files are received as no evidence is kept on the card for future verification]**.

Application Note FCO_NRO.2.1[SC]

- Upon reception of a new application package for installation, the card manager shall first check that it actually comes from the verification authority. The verification authority is the entity responsible for bytecode verification.

FCO_NRO.2.3[SC]:

- The exact limitations on the evidence of origin are implementation dependent. In most of the implementations, the card manager performs an immediate verification of the origin of the package using an electronic signature mechanism, and no evidence is kept on the card for future verifications.

7.2.6.12 FDP_IFC.2[SC] Complete information flow control (SC)

(refines FDP_IFC.2/CM)

Hierarchical to: FDP_IFC.1 Subset information flow control.

Dependencies: FDP_IFF.1 Simple security attributes.

FDP_IFC.2.1[SC] The TSF shall enforce the **[assignment: Secure Channel Protocol information flow control policy]** on **[assignment:**

- **the subjects S.CAD and S.SD, involved in the exchange of messages between the TOE and the CAD through a potentially unsafe communication channel,**
- **the information controlled by this policy are the card content management commands, including personalization commands, in the APDUs sent to the card and their associated responses returned to the CAD**

] and all operations that cause that information to flow to and from subjects covered by the SFP.

FDP_IFC.2.2[SC] The TSF shall ensure that all operations that cause any information in the TOE to flow to and from any subject in the TOE are covered by an information flow control SFP.

7.2.6.13 FDP_IFF.1[SC] Simple security attributes (SC)

(refines FDP_IFF.1/CM)

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control, FMT_MSA.3 Static attribute initialisation.

FDP_IFF.1.1[SC] The TSF shall enforce the **[assignment: Secure Channel Protocol information flow control policy]** based on the following types of subject and information security attributes **[assignment:**

- **Subjects:**
  - **S.SD receiving the Card Content Management commands (through APDUs or APIs).**
  - **S.APPLET receiving commands (through the JCOPX API [9]).**
  - **S.CAD the off-card entity that communicates with S.SD or S.APPLET.**
- **Information:**
  - **executable load file, in case of application loading,**
  - **applications or SD privileges, in case of application installation or registry update,**
  - **personalization keys and/or certificates, in case of application or SD personalization],**
  - **any command, in case of JCOPX API [9]**

].

FDP_IFF.1.2[SC] The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: **[assignment:**

- **Runtime behavior rules defined by GlobalPlatform for:**
  - **loading (Section 9.3.5 of [17]),**
  - **installation (Section 9.3.6 of [17]),**
  - **extradition (Section 9.4.1 of [17]),**
  - **registry update (Section 9.4.2 of [17]),**
  - **content removal (Section 9.5 of [17]),**
- **Runtime behavior rules defined by GlobalPlatform, implemented in JCOPX API [9]**

].

FDP_IFF.1.3[SC] The TSF shall enforce the **[assignment: no additional information flow control SFP rules]**

FDP_IFF.1.4[SC] The TSF shall explicitly authorise an information flow based on the following rules: **[assignment: none]**

FDP_IFF.1.5[SC] The TSF shall explicitly deny an information flow based on the following rules: **[assignment:**

- **When none of the conditions listed in the element FDP_IFF.1.4[SC] of this component hold and at least one of those listed in the element FDP_IFF.1.2[SC] does not hold**

].

Application note The subject S.SD can be the ISD or APSD.

Application note The on-card and the off-card subjects have security attributes such as MAC, Cryptogram, Challenge, Key Set, Static Keys, etc.

7.2.6.14 FMT_MSA.1[SC] Management of security attributes (SC)

(refines FMT_MSA.1/CM)

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control], FMT_SMR.1 Security roles, FMT_SMF.1 Specification of Management Functions.

FMT_MSA.1.1[SC] The TSF shall enforce the **[assignment: Secure Channel Protocol information flow control policy]** to restrict the ability to **[selection: modify]** the security attributes **[assignment:**

- **Key Set,**
- **Security Level,**
- **Secure Channel Protocol,**
- **Session Keys,**
- **Sequence Counter,**
- **ICV**

] to **[assignment: the actor associated with the according security domain:**

- **The Card Issuer for ISD,**
- **The Application Provider for APSD,**
- **The Applet for JCOPX API [9]**

].

Application note The key data used for setting up a secure channel is according to GlobalPlatform [17] and GP Amendment D [18].

7.2.6.15 FMT_MSA.3[SC] Static attribute initialisation (SC)

(refines FMT_MSA.3/CM)

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes, FMT_SMR.1 Security roles.

FMT_MSA.3.1[SC] The TSF shall enforce the **[assignment: Secure Channel Protocol information flow control policy]** to provide **[selection: restrictive]** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2[SC] The TSF shall allow the **[assignment: Card Issuer, Application Provider, Applet]** to specify alternative initial values to override the default values when an object or information is created.

7.2.6.16 FMT_SMF.1[SC] Specification of Management Functions (SC)

(refines FMT_SMF.1/CM)

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1[SC] The TSF shall be capable of performing the following management functions: **[assignment:**

- **Management functions specified in GlobalPlatform specifications [GP]:**
  – loading (Section 9.3.5 of [17]),
  – installation (Section 9.3.6 of [17]),
  – extradition (Section 9.4.1 of [17]),
  – registry update (Section 9.4.2 of [17]),
  – content removal (Section 9.5 of [17]),
- **Attach and retrieve sessions**

].

Application note All management functions related to secure channel protocols shall be relevant.

7.2.6.17 FIA_UID.1[SC] Timing of identification (SC)

(refines FIA_UID.1/CM)

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UID.1.1[SC] The TSF shall allow **[assignment:**

- **application selection,**
- **initializing a secure channel with the card,**
- **requesting data that identifies the card or the Card Issuer**

] on behalf of the user to be performed before the user is identified.

FIA_UID.1.2[SC] The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

7.2.6.18 FIA_UAU.1[SC] Timing of authentication (SC)

(added)

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification.

FIA_UAU.1.1[SC] The TSF shall allow **[assignment: the TSF mediated actions listed in FIA_UID.1[SC]]** on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2[SC] The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

7.2.6.19 FIA_UAU.4[SC] Single-use authentication mechanisms

(added)

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UAU.4.1[SC] The TSF shall prevent reuse of authentication data related to **[assignment: the authentication mechanism used to open a secure communication channel with the card].**

7.2.6.20 FTP_ITC.1[SC] Inter-TSF trusted channel (SC)

(refines FTP_ITC.1/CM)

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1[SC] The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2[SC] [Refined] The TSF shall permit **the CAD placed in the card issuer secured environment** to initiate communication via the trusted channel.

FTP_ITC.1.3[SC] The TSF shall initiate communication via the trusted channel for **[assignment: all card management functions including:**

- **loading,**
- **installation,**
- **extradition,**
- **registry update,**
- **content removal,**
- **changing the Application Life Cycle or Card Life Cycle**

].

7.2.7 ConfG Security Functional Requirements

The list of SFRs of this category define additional requirements related to the configuration of the TOE.

7.2.7.1 FDP_IFC.2[CFG] Complete information flow control (CFG)

Hierarchical to: FDP_IFC.1 Subset information flow control.

Dependencies: FDP_IFF.1 Simple security attributes.

FDP_IFC.2.1[CFG] The TSF shall enforce the **[assignment: Configuration information flow control SFP]** on **[assignment: S.Customer, S.NXP, S.ConfigurationMechanism and D.CONFIG_ITEM]** and all operations that cause that information to flow to and from subjects covered by the SFP.

FDP_IFC.2.2[CFG] The TSF shall ensure that all operations that cause any information in the TOE to flow to and from any subject in the TOE are covered by an information flow control SFP.

7.2.7.2 FDP_IFF.1[CFG] Simple security attributes (CFG)

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control, FMT_MSA.3 Static attribute initialisation.

FDP_IFF.1.1[CFG] The TSF shall enforce the [assignment: Configuration information flow control SFP] based on the following types of subject and information security attributes: **[assignment:**

Subject/Information	Security attributes
S.Customer	**Customer Configuration Token**
S.NXP	**NXP Configuration Token**
S.ConfigurationMechanism	**NXP Configuration Access** , **Customer Configuration Access**

Subject/Information	Security attributes
D.CONFIG_ITEM	access privilege

].

FDP_IFF.2.1[CFG] The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: **[assignment:**

- **Read and write operations of D.CONFIG_ITEM between S.ConfigurationMechanism and S.NXP shall only be possible when S.NXP is authenticated with its token using the Customer Configuration Token.**
- **Read and write operations of D.CONFIG_ITEM between S.ConfigurationMechanism and S.Customer shall only be possible when S.Customer is authenticated with its token using the Customer Configuration Token and if access privilege allows it.**
- **Enabling or disabling of NXP Configuration Access between S.ConfigurationMechanism and S.NXP shall only be possible when S.NXP is authenticated with its token using the NXP Configuration Token.**

]

FDP_IFF.3.1[CFG] The TSF shall enforce the additional information flow control SFP rules: **[assignment: none]**.

FDP_IFF.4.1[CFG] The TSF shall explicitly authorise an information flow based on the following rules: **[assignment: none]**.

FDP_IFF.5.1[CFG] The TSF shall explicitly deny an information flow based on the following rules: **[assignment:**

- **If the NXP Configuration Access is disabled then nobody can read or write D.CONFIG_ITEM.**
- **If the Customer Configuration Access is disabled then S.Customer can not read or write D.CONFIG_ITEM.**

]

7.2.7.3 FMT_MSA.1[CFG] Management of security attributes (CFG)

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control], FMT_ SMR.1 Security roles, FMT_SMF.1 Specification of Management Functions.

FMT_MSA.1.1[CFG] The TSF shall enforce the **[assignment: Configuration information flow control SFP]** to restrict the ability to **[selection: modify]** the security attributes **[assignment: NXP**

Configuration Access **and** Customer Configuration Access] to **[assignment: none]**.

7.2.7.4 FMT_MSA.3[CFG] Static attribute initialisation (CFG)

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_MSA.3.1[CFG] The TSF shall enforce the **[assignment: Configuration information flow control SFP]** to provide **[selection: restrictive]** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2[CFG] The TSF shall allow the **[assignment: nobody]** to specify alternative initial values to override the default values when an object or information is created.

7.2.7.5 FMT_SMR.1[CFG] Security roles (CFG)

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification.

FMT_SMR.1.1[CFG] The TSF shall maintain the roles **[assignment: S.NXP and S.Customer]**.

FMT_SMR.2.1[CFG] The TSF shall be able to associate users with roles.

Application Note: The roles of the Configuration information flow control SFP are defined by the NXP Configuration Token and the Customer Configuration Token.

7.2.7.6 FMT_SMF.1[CFG] Specification of Management Functions (CFG)

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1[CFG] The TSF shall be capable of performing the following management functions: **[assignment: none]**.

7.2.7.7 FIA_UID.1[CFG] Timing of identification (CFG)

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UID.1.1[CFG] The TSF shall allow **[assignment: to select the ISD]** on behalf of the user to be performed before the user is identified.

FIA_UID.1.2[CFG] The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

7.2.8 SecBoxG Security Functional Requirements

The SFRs in this group provide additional requirements to separate the native code executed in the Secure Box environment from the rest of the TOE.

7.2.8.1 FDP_ACC.2[SecureBox] Complete access control (SecureBox)

Hierarchical to: FDP_ACC.1 Subset access control.

Dependencies: FDP_ACF.1 Security attribute based access control.

FDP_ACC.2.1 [SecureBox] The TSF shall enforce the **[assignment: SecureBox access control SFP]** on **[assignment: S.SBNativeCode, O.SB_Content, O.NON_SB_Content, O.SB_SFR, O.NON_SB_SFR]** and all operations among subjects and objects covered by the SFP.

FDP_ACC.2.2 [SecureBox] The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

Refinement: The operations involved in this policy are:

- OP.SB_ACCESS,
- OP.SB_ACCESS_SFR.

7.2.8.2 FDP_ACF.1[SecureBox] Security attribute based access control (SecureBox)

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control, FMT_MSA.3 Static attribute initialisation.

FDP_ACF.1.1 [SecureBox] The TSF shall enforce the **[assignment: SecureBox access control SFP]** to all objects based on the following: **[assignment: S.SBNativeCode, O.SB_Content, O.NON_SB_Content, O.SB_SFR, O.NON_SB_SFR and the attributes CPU Mode, the MMU Segment Table and the Special Function Registers related to system management]**.

FDP_ACF.1.2 [SecureBox] The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **[assignment:**

- **Code assigned to S.SBNativeCode is only executed in CPU Mode User Mode**.
- **Code assigned to S.SBNativeCode is only able to perform OP.SB_ACCESS to O.SB_Content. The ROM, FLASH, and RAM which belongs to O.SB_Content is controlled by the MMU Segment Table used by the Memory Management Unit**.
- **Code assigned to S.SBNativeCode is able to perform OP.SB_ACCESS_SFR to O.SB_SFR**.

].

FDP_ACF.1.3 [SecureBox]: The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **[assignment: none]**.

FDP_ACF.1.4 [SecureBox]: The TSF shall explicitly deny access of subjects to objects based on the following additional rules: **[assignment:**

- **For S.SBNativeCode it is not possible to perform OP.SB_ACCESS to O.NON_SB_Content.**
- **For S.SBNativeCode it is not possible to perform OP.SB_ACCESS_SFR to O.NON_SB_SFR.**

]

7.2.8.3 FMT_MSA.1[SecureBox] Management of security attributes (SecureBox)

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control], FMT_ SMR.1 Security roles, FMT_SMF.1 Specification of Management Functions.

FMT_MSA.1.1 [SecureBox]: The TSF shall enforce the **[assignment: SecureBox access control SFP]** to restrict the ability to **[selection: modify]** the security attributes **[assignment: CPU Mode and the MMU Segment Table]** to **[assignment: S.JCRE]**.

Application Note: The dependency with FMT_SMR.1 is not applicable. Only **S.JCRE** is allowed to modify security attributes for the Secure Box before **S.SBNativeCode** is executed.

7.2.8.4 FMT_MSA.3[SecureBox] Static attribute initialisation (SecureBox)

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes, FMT_SMR.1 Security roles.

FMT_MSA.3.1 [SecureBox]

The TSF shall enforce the **[assignment: SecureBox access control SFP]** to provide **[selection: restrictive]** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 [SecureBox]

The TSF shall allow the **[assignment: S.JCRE]** to specify alternative initial values to override the default values when an object or information is created.

Application Note:

The dependency with FMT_SMR.1 is not applicable. The TOE does not allow to specify alternative initial values for the security attributes of the Secure Box.

7.2.8.5 FMT_SMF.1[SecureBox] Specification of Management Functions (SecureBox)

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1 [SecureBox]

The TSF shall be capable of performing the following management functions: **[assignment:**

- **Switch the CPU Mode,**
- **Change the values in the MMU Segment Table to assign RAM to the Secure Box,**
- **Change the values in the MMU Segment Table to assign FLASH to the Secure Box**

].

7.2.9 ModDesG Security Functional Requirements

The SFRs in this group provide additional requirements related to the Modular Design of the TOE.

7.2.9.1 FDP_IFC.1[MODULAR-DESIGN] Subset information flow control (MODULAR-DESIGN)

Hierarchical to: No other components.

Dependencies: FDP_IFF.1 Simple security attributes.

FDP_IFC.1.1 [MODULAR-DESIGN]

The TSF shall enforce the **[assignment: modular design information flow control SFP]** on **[assignment: S.APPLET, S.SD, S.JCRE, I.MODULE_INVOCATION and OP.INVOKE_MODULE]**.

7.2.9.2 FDP_IFF.1[MODULAR-DESIGN] Simple security attributes (MODULAR-DESIGN)

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control, FMT_MSA.3 Static attribute initialisation.

FDP_IFF.1.1 [MODULAR-DESIGN] The TSF shall enforce the **[assignment: modular design information flow control SFP]** based on the following types of subject and information security attributes: **[assignment: S.APPLET, S.SD, S.JCRE and I.MODULE_INVOCATION with the security attribute Module Presence of the invoked O.CODE_MODULE]**.

FDP_IFF.1.2 [MODULAR-DESIGN] The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: **[assignment: Operation OP.INVOKE_MODULE is allowed for S.APPLET, S.SD and S.JCRE on I.MODULE_INVOCATION if the security attribute Module Presence of the invoked O.CODE_MODULE has the value ”present”]**.

FDP_IFF.1.3 [MODULAR-DESIGN] The TSF shall enforce the additional information flow control SFP rules: **[assignment: none]**.

FDP_IFF.1.4 [MODULAR-DESIGN] The TSF shall explicitly authorise an information flow based on the following rules: **[assignment: none]**.

FDP_IFF.1.5 [MODULAR-DESIGN] The TSF shall explicitly deny an information flow based on the following rules: **[assignment: deny access to O.CODE_MODULE if the security attribute Module Presence has the value ”not present”]**.

7.2.9.3 FIA_ATD.1[MODULAR-DESIGN] User attribute definition (MODULAR-DESIGN)

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_ATD.1.1 [MODULAR-DESIGN] The TSF shall maintain the following list of security attributes belonging to individual users: **[assignment:**

- **Module Presence,**
- **Package AID**

].

Refinement: ”Individual users” stands for Modules.

7.2.9.4 FIA_USB.1[MODULAR-DESIGN] User-subject binding (MODULAR-DESIGN)

Hierarchical to: No other components.

Dependencies: FIA_ATD.1 User attribute definition.

FIA_USB.1.1 [MODULAR-DESIGN] The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: **[assignment: Package AID]**.

FIA_USB.1.2 [MODULAR-DESIGN] The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: **[assignment: Each Module is associated with an unique Package AID]**.

FIA_USB.1.3 [MODULAR-DESIGN] The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: **[assignment: The Package AID of a Module is unchangeable]**.

Application Note: The user is a Module and the subjects are the **S.APPLET**, **S.SD** and **S.JCRE**.

7.2.9.5 FMT_MSA.1[MODULAR-DESIGN] Management of security attributes (MODULAR-DESIGN)

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control], FMT_ SMR.1 Security roles, FMT_SMF.1 Specification of Management Functions.

FMT_MSA.1.1 [MODULAR-DESIGN] The TSF shall enforce the **[assignment: ADEL access control SFP and modular design information flow control SFP]** to restrict the ability to **[selection: modify]** the security attributes **[assignment: Module Presence** of **O.CODE_MODULE]** to **[assignment: S.ADEL]**.

7.2.9.6 FMT_MSA.3[MODULAR-DESIGN] Static attribute initialisation (MODULAR-DESIGN)

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes, FMT_SMR.1 Security roles.

FMT_MSA.3.1 [MODULAR-DESIGN] The TSF shall enforce the **[assignment: modular design information flow control SFP]** to provide **[selection: restrictive]** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 [MODULAR-DESIGN] The TSF shall allow **[assignment: none]** to specify alternative initial values to override the default values when an object or information is created.

7.2.9.7 FMT_SMF.1[MODULAR-DESIGN] Specification of Management Functions (MODULAR-DESIGN)

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1 [MODULAR-DESIGN] The TSF shall be capable of performing the following management functions: **[assignment: modify the list of Resident Modules]**.

7.2.9.8 FMT_SMR.1[MODULAR-DESIGN] Security roles (MODULAR-DESIGN)

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification.

FMT_SMR.1.1 [MODULAR-DESIGN] The TSF shall maintain the roles: **[assignment: Module Invoker]**.

FMT_SMR.1.2 [MODULAR-DESIGN] The TSF shall be able to associate users with roles.

7.2.9.9 FPT_FLS.1[MODULAR-DESIGN] Failure with preservation of secure state (MODULAR-DESIGN)

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_FLS.1.1 [MODULAR-DESIGN] The TSF shall preserve a secure state when the following types of failures occur: **[assignment: OP.INVOKE_MODULE is performed on a TOE internal interface of O.CODE_MODULE where the security attribute Module Presence has the value ”not present”]**.

Application Note: A secure state is being preserved by throwing an exception or sending an error status word to the CAD.

7.2.9.10 FIA_UID.1[MODULAR-DESIGN] Timing of identification (MODULAR-DESIGN)

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UID.1.1 [MODULAR-DESIGN]	The TSF shall allow **[assignment:** • **direct invocation of Modules with public interface and the security attribute Module Presence having the value ’present’,** • **invocation of Modules via Java Card API with TOE internal interface and the security attribute Module Presence having the value ’present’** **]** on behalf of the user to be performed before the user is identified.
FIA_UID.1.2 [MODULAR-DESIGN]	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

7.2.10 Further Security Functional Requirements

The SFRs in this section provide additional proprietary features not covered by the PP [6].

7.2.10.1 FAU_SAS.1[SCP] Audit Data Storage (SCP)

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FAU_SAS.1.1	The TSF shall provide **[assignment: test personnel before TOE Delivery]** with the capability to store the **[assignment: Initialisation Data and/or Prepersonalisation Data and/or supplements of the SmartCard Embedded Software]** in the **[assignment: audit records]**.

7.2.10.2 FIA_AFL.1[PIN] Basic Authentication Failure Handling (PIN)

Hierarchical to:	No other components.
Dependencies:	FIA_UAU.1 Timing of authentication.
FIA_AFL.1.1[PIN]	The TSF shall detect when **[selection: an administrator configurable positive integer within [1 and 127]]** unsuccessful authentication attempts occur related to **[assignment: any user authentication using D.PIN]**.
FIA_AFL.1.2[PIN]	When the defined number of unsuccessful authentication attempts has been **[selection:surpassed]**, the TSF shall **[assignment: block the authentication with D.PIN]**.

Application Note	The dependency with FIA_UAU.1 is not applicable. The TOE implements the firewall access control SFP, based on which access to the object implementing FIA_AFL.1[PIN] is organized.

7.2.10.3 FPT_EMSEC.1 TOE emanation

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_EMSEC.1.1	The TOE shall not emit **[assignment: variations in power consumption or timing during command execution]** in excess of **[assignment: non-useful information]** enabling access to **[assignment: TSF data: D.CRYPTO]** and **[assignment: User data: D.PIN, D.APP_KEYs]**.
FPT_EMSEC.1.2	The TSF shall ensure **[assignment: that unauthorized users]** are unable to use the following interface **[assignment: electrical contacts or Radio Frequency (RF) field]** to gain access to **[assignment: TSF data: D.CRYPTO]** and **[assignment: User data: D.PIN, D.APP_KEYs]**.

7.2.10.4 FPT_PHP.3 Resistance to physical attack

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_PHP.3.1	The TSF shall resist **[assignment: physical manipulation and physical probing]** to the **[assignment: TSF]** by responding automatically such that the SFRs are always enforced.
Refinement:	The TSF will implement appropriate mechanisms to continuously counter physical manipulation and physical probing. Due to the nature of these attacks (especially manipulation) the TSF can by no means detect attacks on all of its elements. Therefore, permanent protection against these attacks is required ensuring that security functional requirements are enforced. Hence, ”automatic response” means here (i) assuming that there might be an attack at any time and (ii) countermeasures are provided at any time.
Application Note:	This SFR is taken from the certified Security IC Platform Protection Profile [5].

7.2.10.5 FCS_CKM.2 Cryptographic key distribution

Hierarchical to:	No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction.

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method **[assignment: methods: set keys and components of DES, AES, RSA, RSA-CRT, ECC and secure messaging]** that meets the following: **[assignment: [14], [9]]**.

Application Notes:

- The keys can be accessed as specified in [14] Key class and [9] for proprietary classes.
- FCS_CKM.2 for ECC keys is applicable only if the corresponding Module for the cryptographic operation is present in the TOE.

7.2.10.6 FCS_CKM.3 Cryptographic key access

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction.

FCS_CKM.3.1 The TSF shall perform **[assignment: management of DES, AES, RSA, RSA-CRT, ECC, Diffie-Hellman and EC Diffie-Hellman]** in accordance with a specified cryptographic key access method **[assignment: methods/commands defined in packages javacard.security of [14] and [9] for proprietary classes]** that meets the following: **[assignment:** [14] and [9]**]**.

Application Note:

- The keys can be accessed as specified in [14] and [9] for proprietary classes.
- FCS_CKM.3 for ECC keys is applicable only if the corresponding Module for the cryptographic operation is present in the TOE.

7.2.10.7 FDP_SDI.2[SENSITIVE_RESULT] Stored data integrity monitoring and action (Sensitive Result)

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring.

Dependencies: No dependencies.

FDP_SDI.2.1 [SENSITIVE _RESULT] The TSF shall monitor user data stored in containers controlled by the TSF for **[assignment: integrity errors]** on all objects, based on the following attributes:

[assignment: sensitive API result stored in the com.nxp.id.jcopx.security.SensitiveResultX class].

FDP_SDI.2.2 [SENSITIVE _RESULT]	Upon detection of a data integrity error, the TSF shall **[assignment: throw an exception]**.

7.3 Security Assurance Requirements

The assurance requirements of this evaluation are EAL6 augmented by ASE_TSS.2 and ALC_FLR.1. The assurance requirements ensure, among others, the security of the TOE during its development and production.

7.3.1 ADV_SPM.1 Formal TOE security policy model

Hierarchical to:	No other components.
Dependencies:	ADV_FSP.4 Complete functional specification.
ADV_SPM.1.1	The developer shall provide a formal security policy model for the **[assignment: FIREWALL access control SFP (FDP_ACC.2[FIREWALL])]**.

7.4 Security Requirements Rationale for the TOE

The following rationales of Security Functional Requirements for the TOE only cover the modifications regarding the PP [6] due to additions and refinements in Security Functional Requirements and Security Objectives.

7.4.1 Identification

OT.SID

SFR	Rationale
FDP_ITC.2[CCM]	Subjects’ identity is AID-based (applets, packages) and is met by the SFR.
FMT_MSA.1[SC]	Subjects’ identity is AID-based (applets, packages) and is met by the SFR.
FMT_MSA.3[SC]	Subjects’ identity is AID-based (applets, packages) and is met by the SFR.
FMT_SMF.1[SC]	Subjects’ identity is AID-based (applets, packages) and is met by the SFR.

OT.SID_MODULE

SFR	Rationale
FDP_IFC.1[MODULAR-DESIGN]	The modular design information flow control policy contributes to meet this objective.

SFR	Rationale
FDP_IFF.1[MODULAR-DESIGN]	The modular design information flow control policy contributes to meet this objective.
FIA_ATD.1[MODULAR-DESIGN]	Subject’s identity is AID-based and is met by the SFR.
FIA_USB.1[MODULAR-DESIGN]	Subject’s identity is AID-based and is met by the SFR. (Re-) loading of a previously deleted Module or Module replacement are not possible, protecting against identity forgery.
FMT_MSA.1[MODULAR-DESIGN]	Contributes indirectly to meet this objective.
FMT_MSA.3[MODULAR-DESIGN]	Contributes indirectly to meet this objective.
FMT_SMF.1[MODULAR-DESIGN]	Contributes indirectly to meet this objective.
FMT_SMR.1[MODULAR-DESIGN]	Contributes indirectly to meet this objective.
FPT_FLS.1[MODULAR-DESIGN]	Contributes indirectly to meet this objective.
FIA_UID.1[MODULAR-DESIGN]	Contributes to meet the objective by only allowing invocation of Modules if they are present.

7.4.2 Execution

OT.FIREWALL

SFR	Rationale
FDP_ITC.2[CCM]	Contributes indirectly to meet this objective.
FMT_SMR.1[SD]	Contributes indirectly to meet this objective.
FMT_MSA.1[SC]	Contributes indirectly to meet this objective.
FMT_MSA.3[SC]	Contributes indirectly to meet this objective.
FMT_SMF.1[SC]	Contributes indirectly to meet this objective.

OT.OPERATE

SFR	Rationale
FIA_AFL.1[PIN]	Contributes to meet the objective by protecting the authentication.
FDP_ITC.2[CCM]	Contributes to meet this objective by detecting and blocking various failures or security violations during usual working.

OT.RESOURCES

SFR	Rationale
FMT_SMR.1[SD]	Contributes to meet this objective since the TSF controls the memory management
FMT_SMF.1[SC]	Contributes to meet this objective since the TSF controls the memory management

OT.SENSITIVE_RESULTS_INTEG

SFR	Rationale
FDP_SDI.2[SENSITIVE_RESULT]	Directly contributes to meet the objective by ensuring that integrity errors related to the sensitive API result are detected by the TOE

7.4.3 Services

OT.CIPHER

SFR	Rationale
FCS_CKM.2	Covers the objective directly
FCS_CKM.3	Covers the objective directly

OT.RNG

SFR	Rationale
FCS_RNG.1[HDT]	Covers the objective directly

OT.KEY-MNGT

SFR	Rationale
FCS_CKM.2	Covers the objective directly
FCS_CKM.3	Covers the objective directly

OT.PIN-MNGT

SFR	Rationale
FIA_AFL.1[PIN]	Directly contributes to meet the objective

7.4.4 Applet Management

OT.APPLI-AUTH

SFR	Rationale
FCS_COP.1	Refinement: applies to FCS_COP.1[DAP]. Contributes to meet the security objective by ensuring that the loaded Executable Application is legitimate by specifying the algorithm to be used in order to verify the DAP signature of the Verification Authority
FDP_ROL.1[CCM]	Contributes to meet this security objective by ensures that card management operations may be cleanly aborted
FPT_FLS.1[CCM]	Contributes to meet the security objective by preserving a secure state when failures occur

OT.DOMAIN-RIGHTS

SFR	Rationale
FDP_ACC.1[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FDP_ACF.1[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FMT_MSA.1[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FMT_MSA.3[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FMT_SMF.1[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FMT_SMR.1[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FTP_ITC.1[SC]	Contributes to cover this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FCO_NRO.2[SC]	Contributes to cover this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FDP_IFC.2[SC]	Contributes to cover this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FDP_IFF.1[SC]	Contributes to cover this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FMT_MSA.1[SC]	Contributes to cover this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FMT_MSA.3[SC]	Contributes to cover this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations

SFR	Rationale
FMT_SMF.1[SC]	Contributes to cover this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FIA_UID.1[SC]	Contributes to cover this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FIA_UAU.1[SC]	Contributes to cover this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FIA_UAU.4[SC]	Contributes to cover this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations

OT.COMM_AUTH

SFR	Rationale
FCS_COP.1	Contributes to meet the security objective by specifying secure cryptographic algorithm that shall be used to determine the origin of the card management commands
FMT_SMR.1[SD]	Contributes to meet the security objective by specifying the authorized identified roles enabling to send and authenticate card management commands
FTP_ITC.1[SC]	Contributes to meet the security objective by ensuring the origin of card administration commands
FDP_IFC.2[SC]	Contributes to meet the security objective by specifying the authorized identified roles enabling to send and authenticate card management commands
FDP_IFF.1[SC]	Contributes to meet the security objective by specifying the authorized identified roles enabling to send and authenticate card management commands
FMT_MSA.1[SC]	Contributes to meet the security objective by specifying security attributes enabling to authenticate card management requests
FMT_MSA.3[SC]	Contributes to meet the security objective by specifying security attributes enabling to authenticate card management requests
FIA_UID.1[SC]	Contributes to meet the security objective by specifying the actions that can be performed before authenticating the origin of the APDU commands that the TOE receives
FIA_UAU.1[SC]	Contributes to meet the security objective by specifying the actions that can be performed before authenticating the origin of the APDU commands that the TOE receives

OT.COMM_INTEGRITY

SFR	Rationale
FCS_COP.1	Contributes to meet the security objective by by specifying secure cryptographic algorithm that shall be used to ensure the integrity of the card management commands
FMT_SMR.1[SD]	Contributes to cover this security objective by defining the roles enabling to send and authenticate the card management requests for which the integrity has to be ensured
FTP_ITC.1[SC]	Contributes to meet the security objective by ensuring the integrity of card management commands
FDP_IFC.2[SC]	Contributes to cover the security objective by enforcing the Secure Channel Protocol information flow control policy to guarantee the integrity of administration requests
FDP_IFF.1[SC]	Contributes to cover the security objective by enforcing the Secure Channel Protocol information flow control policy to guarantee the integrity of administration requests
FMT_MSA.1[SC]	Contributes to cover the security objective by specifying security attributes enabling to guarantee the integrity of card management requests
FMT_MSA.3[SC]	Contributes to cover the security objective by specifying security attributes enabling to guarantee the integrity ocard management requests
FMT_SMF.1[SC]	Contributes to meet the security objective by specifying the actions activating the integrity check on the card management commands

OT.COMM_CONFIDENTIALITY

SFR	Rationale
FCS_COP.1	Contributes to meet this objective by specifying secure cryptographic algorithm that shall be used to ensure the confidentiality of the card management commands
FMT_SMR.1[SD]	Contributes to cover the security objective by defining the roles enabling to send and authenticate the card management requests for which the confidentiality has to be ensured
FTP_ITC.1[SC]	Contributes to cover the security objective by ensuring the confidentiality of card management commands
FDP_IFC.2[SC]	Contributes to cover the security objective by enforcing the Secure Channel Protocol information flow control policy to guarantee the confidentiality of administration requests
FDP_IFF.1[SC]	Contributes to cover the security objective by enforcing the Secure Channel Protocol information flow control policy to guarantee the confidentiality of administration requests
FMT_MSA.1[SC]	Contributes to cover the security objective by specifying security attributes enabling to guarantee the confidentiality of card management requests by decrypting those requests and imposing management conditions on that attributes

SFR	Rationale
FMT_MSA.3[SC]	Contributes to cover the security objective by specifying security attributes enabling to guarantee the confidentiality of card management requests by decrypting those requests and imposing management conditions on that attributes
FMT_SMF.1[SC]	Contributes to cover the security objective by specifying the actions ensuring the confidentiality of the card management commands

7.4.5 Card Management

OT.CARD-MANAGEMENT

SFR	Rationale
FDP_ACC.2[ADEL]	Contributes to meet the objective by the ADEL access control policy which ensures the non-introduction of security holes. The integrity and confidentiality of data that does not belong to the deleted applet or package is a by-product of this policy as well
FDP_ACF.1[ADEL]	Contributes to meet the objective by the ADEL access control policy which ensures the non-introduction of security holes. The integrity and confidentiality of data that does not belong to the deleted applet or package is a by-product of this policy as well
FDP_RIP.1[ADEL]	Contributes to meet the objective by ensuring the non-accessibility of deleted data
FMT_MSA.1[ADEL]	Contributes to meet the objective by enforcing the ADEL access control SFP
FMT_MSA.3[ADEL]	Contributes to meet the objective by enforcing the ADEL access control SFP
FMT_SMR.1[ADEL]	Contributes to meet the objective by maintaining the role applet deletion manager
FPT_RCV.3[INSTALLER]	Contributes to meet the objective by protecting the TSFs against possible failures of the deletion procedures
FPT_FLS.1[INSTALLER]	Contributes to meet the objective by protecting the TSFs against possible failures of the installer
FPT_FLS.1[ADEL]	Contributes to meet the objective by protecting the TSFs against possible failures of the deletion procedures
FDP_UIT.1[CCM]	Contributes to meet the objective by enforcing the Secure Channel Protocol information flow control policy and the Security Domain access control policy which controls the integrity of the corresponding data
FDP_ROL.1[CCM]	Contributes to meet this security objective by ensures that card management operations may be cleanly aborted
FDP_ITC.2[CCM]	Contributes to meet the security objective by enforcing the Firewall access control policy and the Secure Channel Protocol information flow policy when importing card management data

SFR	Rationale
FPT_FLS.1[CCM]	Contributes to meet the security objective by preserving a secure state when failures occur
FDP_ACC.1[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FDP_ACF.1[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FMT_MSA.1[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FMT_MSA.3[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FMT_SMF.1[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FMT_SMR.1[SD]	Contributes to cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management
FTP_ITC.1[SC]	Contributes to meet this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FCO_NRO.2[SC]	Contributes to meet this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FDP_IFC.2[SC]	Contributes to meet this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FDP_IFF.1[SC]	Contributes to meet this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FMT_MSA.1[SC]	Contributes to meet this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FMT_MSA.3[SC]	Contributes to meet this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations

SFR	Rationale
FMT_SMF.1[SC]	Contributes to meet this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FIA_UID.1[SC]	Contributes to meet this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FIA_UAU.1[SC]	Contributes to meet this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations
FIA_UAU.4[SC]	Contributes to meet this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations

7.4.6 Smart Card Platform

OT.SCP.IC

SFR	Rationale
FAU_ARP.1	Contributes to the coverage of the objective by resetting the card session or terminating the card in case of physical tampering
FPR_UNO.1	Contributes to the coverage of the objective by ensuring leakage resistant implementations of the unobservable operations
FPT_EMSEC.1	Contributes to meet the objective
FPT_PHP.3	Contributes to the coverage of the objective by preventing bypassing, deactivation or changing of other security features

OT.SCP.RECOVERY

SFR	Rationale
FAU_ARP.1	Contributes to the coverage of the objective by ensuring reinitialization of the Java Card System and its data after card tearing and power failure
FPT_FLS.1	Contributes to the coverage of the objective by preserving a secure state after failure

OT.SCP.SUPPORT

SFR	Rationale
FCS_CKM.1	Contributes to meet the objective
FCS_CKM.4	Contributes to meet the objective
FCS_COP.1	Contributes to meet the objective

SFR	Rationale
FDP_ROL.1[FIREWALL]	Contributes to meet the objective

OT.IDENTIFICATION

SFR	Rationale
FAU_SAS.1[SCP]	Covers the objective.The Initialisation Data (or parts of them) are used for TOE identification

7.4.7 Random Numbers

OT.RNG

SFR	Rationale
FCS_RNG.1	Counters the threat by ensuring the cryptographic quality of random number generation. For instance random numbers shall not be predictable and shall have sufficient entropy. Furthermore, the TOE ensures that no information about the produced random numbers is available to an attacker
FCS_RNG.1[HDT]	Counters the threat by ensuring the cryptographic quality of random number generation. For instance random numbers shall not be predictable and shall have sufficient entropy. Furthermore, the TOE ensures that no information about the produced random numbers is available to an attacker

7.4.8 Configuration Module

OT.CARD-CONFIGURATION

SFR	Rationale
FDP_IFC.2[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FDP_IFF.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_MSA.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_MSA.3[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_SMR.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_SMF.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FIA_UID.1[CFG]	Contributes to meet the objective by requiring identification before modifying configuration items

OT.SECURE_LOAD_ACODE

SFR	Rationale
FDP_IFC.2[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FDP_IFF.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_MSA.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_MSA.3[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_SMR.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_SMF.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FIA_UID.1[CFG]	Contributes to meet the objective by requiring identification before modifying configuration items

OT.SECURE_ACTIVATION_ADDITIONAL_CODE

SFR	Rationale
FDP_IFC.2[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FDP_IFF.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_MSA.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_MSA.3[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_SMR.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_SMF.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FIA_UID.1[CFG]	Contributes to meet the objective by requiring identification before modifying configuration items

OT.TOE_IDENTIFICATION

SFR	Rationale
FDP_IFC.2[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FDP_IFF.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_MSA.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_MSA.3[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FMT_SMR.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items

SFR	Rationale
FMT_SMF.1[CFG]	Contributes to meet the objective by controlling the ability to modify configuration items
FIA_UID.1[CFG]	Contributes to meet the objective by requiring identification before modifying configuration items

7.4.9 Secure Box

OT.SEC_BOX_FW

SFR	Rationale
FDP_ACC.2[SecureBox]	Contributes to meet the objective by applying access control rules
FDP_ACF.1[SecureBox]	Contributes to meet the objective by applying access control rules
FMT_MSA.1[SecureBox]	Contributes to meet the objective by enforcing the Secure-Box access control SFP
FMT_MSA.3[SecureBox]	Contributes to meet the objective by enforcing the Secure-Box access control SFP
FMT_SMF.1[SecureBox]	Contributes to cover this security objective by enforcing the SecureBox access control policy which ensures a separation of the Secure Box from the rest of the TOE

7.5 SFR Dependencies

Requirements	CC Dependencies	Satisfied Dependencies
FAU_SAS.1[SCP]	No dependencies	
FCO_NRO.2[SC]	FIA_UID.1 Timing of identification	FIA_UID.1[SC]
FCS_CKM.2	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1 FCS_CKM.4
FCS_CKM.3	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1 FCS_CKM.4
FDP_ACC.1[SD]	FDP_ACF.1 Security attribute based access control	FDP_ACF.1[SD]
FDP_ACC.2[SecureBox]	FDP_ACF.1 Security attribute based access control	FDP_ACF.1(SecureBox)
FDP_ACF.1[SecureBox]	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.2[SecureBox] FMT_MSA.3[SecureBox]

Requirements	CC Dependencies	Satisfied Dependencies
FDP_ACF.1[SD]	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.1[SD] FMT_MSA.3[SD]
FDP_IFC.2[SC]	FDP_IFF.1 Simple security attributes	FDP_IFF.1[SC]
FDP_IFC.2[CFG]	FDP_IFF.1 Simple security attributes	FDP_IFF.1[CFG]
FDP_IFC.1[MODULAR-DESIGN]	FDP_IFF.1 Simple security attributes	FDP_IFF.1[MODULAR-DESIGN]
FDP_IFF.1[SC]	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation	FDP_IFC.2[SC] FMT_MSA.3[SC]
FDP_IFF.1[CFG]	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation	FDP_IFC.2[CFG] FMT_MSA.3[CFG]
FDP_IFF.1[MODULAR-DESIGN]	FDP_IFC.1 Subset information flow control, FMT_MSA.3 Static attribute initialisation	FDP_IFC.1[MODULAR-DESIGN] FMT_MSA.3[MODULAR-DESIGN]
FDP_ITC.2[CCM]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1[SD] FTP_ITC.1[SC]
FDP_ROL.1[CCM]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	FDP_ACC.1[SD]
FDP_SDI.2[SENSITIVE_RESULT]	No dependencies	
FDP_UIT.1[CCM]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path]	FDP_ACC.1[SD] FTP_ITC.1[SC]
FIA_AFL.1[PIN]	FIA_UAU.1 Timing of authentication	see AppNote in FIA_AFL.1[PIN]
FIA_ATD.1[MODULAR-DESIGN]	No dependencies	
FIA_UID.1[SC]	No dependencies	
FIA_UID.1[CFG]	No dependencies	
FIA_UID.1[MODULAR-DESIGN]	No dependencies	
FIA_USB.1[MODULAR-DESIGN]	FIA_ATD.1 User attribute definition	FIA_ATD.1[MODULAR-DESIGN]
FIA_UAU.1[SC]	FIA_UID.1 Timing of identification	FIA_UID.1[SC]
FIA_UAU.4[SC]	No dependencies	

Requirements	CC Dependencies	Satisfied Dependencies
FMT_MSA.1[SC]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_ACC.1[SD] FMT_SMR.1[SD] FMT_SMF.1[SC]
FMT_MSA.1[SecureBox]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_ACC.2[SecureBox] FMT_SMR.1 FMT_SMF.1[SecureBox]
FMT_MSA.1[CFG]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_IFC.2[CFG] FMT_SMR.1[CFG] FMT_SMF.1[CFG]
FMT_MSA.1[SD]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_ACC.1[SD] FMT_SMR.1[SD] FMT_SMF.1[SD]
FMT_MSA.1[MODULAR-DESIGN]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control], FMT_SMR.1 Security roles, FMT_SMF.1 Specification of Management Functions	FDP_IFC.1[MODULAR-DESIGN] FMT_SMR.1[MODULAR-DESIGN] FMT_SMF.1[MODULAR-
FMT_MSA.3[SecureBox]	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1[SecureBox] FMT_SMR.1
FMT_MSA.3[CFG]	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1[CFG] FMT_SMR.1[CFG]
FMT_MSA.3[SD]	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1[SD] FMT_SMR.1[SD]
FMT_MSA.3[SC]	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1[SC] FMT_SMR.1[SD]
FMT_MSA.3[MODULAR-DESIGN]	FMT_MSA.1 Management of security attributes, FMT_SMR.1 Security roles	FMT_MSA.1[MODULAR-DESIGN] FMT_SMR.1[MODULAR-DESIGN]
FMT_SMF.1[SecureBox]	No dependencies	
FMT_SMF.1[CFG]	No dependencies	
FMT_SMF.1[SD]	No dependencies	
FMT_SMF.1[SC]	No dependencies	
FMT_SMF.1[MODULAR-DESIGN]	No dependencies	

Requirements	CC Dependencies	Satisfied Dependencies
FMT_SMR.1[CFG]	FIA_UID.1 Timing of identification	FIA_UID.1[CFG]
FMT_SMR.1[SD]	FIA_UID.1 Timing of identification	FIA_UID.1[SC]
FMT_SMR.1[MODULAR-DESIGN]	FIA_UID.1 Timing of identification	FIA_UID.1[MODULAR-DESIGN]
FPT_EMSEC.1	No dependencies	
FPT_FLS.1[CCM]	No dependencies	
FPT_FLS.1[MODULAR-DESIGN]	No dependencies	
FPT_PHP.3	No dependencies	
FTP_ITC.1[SC]	No dependencies	
ADV_SPM.1	ADV_FSP.4 Complete functional specification	ADV_FSP.4

7.5.1 Rationale for Exclusion of Dependencies

The dependency FIA_UID.1 of FMT_SMR.1[INSTALLER] is unsupported. This ST does not require the identification of the "Installer" since it can be considered as part of the TSF.

The dependency FIA_UID.1 of FMT_SMR.1[ADEL] is unsupported. This ST does not require the identification of the "applet deletion manager" since it can be considered as part of the TSF.

The dependency FIA_UID.1 of FMT_SMR.1[MODULAR-DESIGN] is unsupported. This ST does not require the identification of the "Module Invoker" since it can be considered as part of the TSF.

The dependency FMT_SMF.1 of FMT_MSA.1[JCRE] is unsupported. The dependency between FMT_MSA.1[JCRE] and FMT_SMF.1 is not satisfied because no management functions are required for the Java Card RE.

The dependency FAU_SAA.1 of FAU_ARP.1 is unsupported. The dependency of FAU_ARP.1 on FAU_SAA.1 assumes that a "potential security violation" generates an audit event. On the contrary, the events listed in FAU_ARP.1 are self-contained (arithmetic exception, ill-formed bytecodes, access failure) and ask for a straightforward reaction of the TSFs on their occurrence at runtime. The JCVM or other components of the TOE detect these events during their usual working order. Thus, there is no mandatory audit recording in this ST.

The dependency FIA_UAU.1 of FIA_AFL.1[PIN] is unsupported. The TOE implements the firewall access control SFP, based on which access to the object Implementing FIA_AFL.1[PIN] is organized.

The dependencies FMT_SMR.1 of FMT_MSA.1[SecureBox] and FMT_MSA.3[SecureBox] are unsupported. Only S.JCRE is allowed to modify security attributes for the Secure Box before S.SBNativeCode is executed. Furthermore the TOE does not allow to specify alternative initial values for the security attributes of the Secure Box.

7.6 Security Assurance Requirements Rationale

The selection of assurance components is based on the underlying PP [6]. The Security Target uses the augmentations from the PP, chooses EAL6 and adds the components ASE_TSS.2 and ALC_FLR.1.

The rationale for the augmentations is the same as in the PP.

The assurance level EAL6 is an elaborated pre-defined level of the CC, Part 3 [3]. The assurance components in an EAL level are chosen in a way that they build a mutually supportive and complete set of components.

The additional requirements chosen for augmentation do not add any dependencies, which are not already fulfilled for the corresponding requirements contained in EAL6. Therefore, the components ASE_TSS.2 and ALC_FLR.1 add additional assurance to EAL6, but the mutual support of the requirements is still guaranteed.

8 TOE summary specification (ASE_TSS)

8.1 Introduction

The Security Functions (SF) introduced in this section realize the SFRs of the TOE. See Table 18 "Overview of Security Functions" for list of all Security Functions. Each SF consists of components spread over several TOE components to provide a security functionality and fulfill SFRs.

8.2 Security Functionality

Table 18. Overview of Security Functions

Name	Title
SF.JCVM	Java Card Virtual Machine
SF.CONFIG	Configuration Management
SF.OPEN	Card Content Management
SF.CRYPTO	Cryptographic Functionality
SF.RNG	Random Number Generator
SF.DATA_STORAGE	Secure Data Storage
SF.PUF	User Data Protection using PUF
SF.OM	Java Object Management
SF.MM	Memory Management
SF.PIN	PIN Management
SF.PERS_MEM	Persistent Memory Management
SF.EDC	Error Detection Code API
SF.HW_EXC	Hardware Exception Handling
SF.PID	Platform Identification
SF.SMG_NSC	No Side-Channel
SF.ACC_SBX	Secure Box
SF.MOD_INVOC	Module Invocation
SF.SENS_RES	Sensitive Result

8.2.1 SF.JVCM: Java Card Virtual Machine

SF.JCVM provides the Java Card Virtual Machine including byte code interpretation and the Java Card Firewall according to the specifications [16], [15]. This fulfills the SFRs FDP_IFC.1[JCVM], FDP_IFF.1[JCVM], FMT_SMF.1, FMT_SMR.1, FDP_ROL.1[FIREWALL], FDP_ACF.1[FIREWALL], FDP_ACC.2[FIREWALL] and FIA_UID.2[AID]. SF.JCVM supports FAU_ARP.1 and FPT_FLS.1 by throwing Java Exceptions according to these specifications. Additionally it supports these SFRs by verification of the integrity of used Java object headers.

Security attributes in SF.JCVM are separated from user data and not accessible by applets to fulfill FMT_MSA.1[JCRE] and FMT_MSA.1[JCVM]. All values for security attributes are initialized and assigned by the system itself which fulfills FMT_MSA.2[FIREWALL-JCVM], FMT_MSA.3[FIREWALL] , and FMT_MSA.3[JCVM].

SF.JCVM ensures together with SF.PERS_MEM that the system is halted in case non existing Java objects could be referenced after an aborted transaction to fulfill FDP_RIP.1[ABORT].

8.2.2 SF.CONFIG: Configuration Management

SF.CONFIG provides means to store Initialization Data and Pre-personalization Data before TOE delivery FAU_SAS.1[SCP].

SF.CONFIG provides means to change configuration items of the card. Some configuration items can be changed by the customer and some can only be changed by NXP (FDP_IFC.2[CFG], FDP_IFF.1[CFG] , FMT_MSA.3[CFG], FMT_MSA.1[CFG], FMT_SMR.1[CFG], FMT_SMF.1[CFG], FIA_UID.1[CFG]). SF.CONFIG supports FCS_COP.1 by configuring the behavior of cryptographic operations.

8.2.3 SF.OPEN: Card Content Management

SF.OPEN provides the card content management functionality according the GlobalPlatform Specification [17]. This supports FCO_NRO.2[SC], FDP_ACC.1[SD], FDP_ACF.1[SD], FDP_UIT.1[CCM], FDP_IFC.2[SC], FDP_IFF.1[SC], FDP_IFC.2[SC], FIA_UID.1[SC], FIA_UID.2[AID], FIA_USB.1[AID], FMT_MSA.1[SC], FMT_MSA.1[SD], FMT_MSA.3[SC], FMT_MSA.3[SD], FMT_SMF.1[ADEL] , FMT_SMR.1[SD], FMT_SMF.1[SC], FMT_SMF.1[SD], FTP_ITC.1[SC], FMT_MSA.3[ADEL], FMT_SMR.1[INSTALLER], FMT_SMR.1[ADEL], FDP_ITC.2[CCM], FDP_ROL.1[CCM], FIA_UAU.1[SC], FIA_UAU.4[SC], and FTP_ITC.1[SC]. In addition to the GP specification, the Java Card Runtime Environment specification [16] is followed to support FDP_ACC.2[ADEL], FDP_ACF.1[ADEL], FMT_MSA.3[SC], FMT_MSA.3[SD], FMT_MTD.1[JCRE], FMT_MTD.3[JCRE], FPT_FLS.1[INSTALLER], FDP_RIP.1[bArray], FDP_RIP.1[ADEL], FPT_TDC.1, FPT_FLS.1[ADEL], and FPT_FLS.1[CCM] for application loading, installation, and deletion.

AID management is provided by SF.OPEN according to the GlobalPlatform Specification [17], the Java Card Runtime Environment Specification [16], and the Java Card API Specification [14] to support FIA_ATD.1[AID].

SF.OPEN is part of the TOE runtime environment and thus separated from other applications to fulfill FMT_MSA.1[ADEL]. It supports FAU_ARP.1 and FPT_FLS.1 by responding with error messages according to the GlobalPlatform mapping guidelines [20] and fulfills FPT_RCV.3[INSTALLER] by inherent memory cleanup in case of aborted loading and installation.

8.2.4 SF.CRYPTO: Cryptographic Functionality

SF.CRYPTO provides key creation, key management, key deletion and cryptographic functionality. It provides the API in accordance to the Java Card API Specification [14] to fulfill FCS_CKM.1, FCS_CKM.2, FCS_CKM.3, FCS_CKM.4, and FCS_COP.1. Proprietary solutions (e.g., key lengths not supported by the Java Card API) are supported following the Java Card API. SF.CRYPTO uses SF.DATA_STORAGE to support FCS_CKM.1, FCS_CKM.2, FCS_CKM.3, FCS_CKM.4, FDP_RIP.1[KEYS], and FDP_SDI.2[DATA]. The Crypto Lib certified with the TOE hardware supports FCS_COP.1 and FPR_UNO.1.

8.2.5 SF.RNG: Random Number Generator

SF.RNG provides secure random number generation to fulfill FCS_CKM.1 and FCS_RNG.1. Random numbers are generated by the Crypto Lib certified with the TOE hardware. SF.RNG provides an API according to the Java Card API Specification [14] to generate random numbers according to FCS_RNG.1.

8.2.6 SF.DATA_STORAGE: Secure Data Storage

SF.DATA_STORAGE provides a secure data storage for confidential data. It is used to store cryptographic keys (supports FCS_CKM.1, FCS_CKM.2, FCS_CKM.3, and FCS_CKM.4) and to store PINs (supports FIA_AFL.1[PIN]). All data stored by SF.DATA_STORAGE is CRC32 integrity protected to fulfill FDP_SDI.2[DATA], FAU_ARP.1, and FPT_FLS.1. The stored data is AES encrypted to fulfill FPR_UNO.1.

8.2.7 SF.PUF: User Data Protection using PUF

SF.PUF implements a mechanism to seal/unseal the user data stored in shared memory against unintended disclosure. SF.PUF encrypts/decrypts the user data with a cryptographic key which is derived from the PUF data and stored directly in the hardware. SF.PUF calculates a MAC as a PUF authentication value. SF.PUF serves to seal/unseal the user data stored in the memory. The user data stored in the memory can be encrypted/decrypted using the PUF block. A MAC (message authentication code) can be calculated as a PUF authentication value. Hence, the user data can be sealed within the TOE and can be solely unsealed by the TOE. The cryptographic key for sealing/unsealing of the user data is generated with the help of a key derivation function based on the PUF block and the Random Number Generator (RNG). The PUF block provides the PUF data to the key derivation function and thereby the cryptographic key is derived. If the TOE is powered off, the PUF data is not available from the PUF block. Therefore SF.PUF is suitable to meet FCS_CKM.1.1[PUF] and FCS_CKM.4.1[PUF]. The encryption/decryption of user data and the calculation of a MAC as a PUF authentication value are performed within the AES coprocessor. Therefore SF.PUF is suitable to meet FCS_COP.1.1[PUF_AES] and FCS_COP.1.1[PUF_MAC].

8.2.8 SF.OM: Java Object Management

SF.OM provides the object management for Java objects which are processed by SF.JCVM. It provides object creation (FDP_RIP.1[OBJECTS]) and garbage collection according to the Java Card Runtime Environment Specification [16] to fulfill FDP_RIP.1[ODEL] and FPT_FLS.1[ODEL]. SF.OM throws an Java Exception in case an object cannot be created as requested due to too less available memory. This fulfills FAU_ARP.1 and FPT_FLS.1.

8.2.9 SF.MM: Memory Management

SF.MM provides deletion of memory for transient arrays, global arrays, and logical channels according to the Java Card Runtime Environment Specification [16]. Thus, it fulfills FDP_RIP.1[TRANSIENT] by granting access to and erasing of CLEAR_ON_RESET and CLEAR_ ON_DESELECT transient arrays. It supports FIA_ATD.1[AID] when using logical channels and it fulfills FDP_RIP.1[APDU], FDP_RIP.1[bArray] and FDP_RIP.1[GlobalArray_Refined] by clearing the APDU buffers for new incoming data, by clearing the bArray during application installation and preventing applications to keep a pointer to global arrays.

8.2.10 SF.PIN: PIN Management

SF.PIN provides secure PIN management by using SF.DATA_STORAGE for PIN objects specified in the Java Card API Specification [14] and the GlobalPlatform Specification [17]. Thus, it fulfills FDP_SDI.2[DATA], FIA_AFL.1[PIN], and FPR_UNO.1.

8.2.11 SF.PERS_MEM: Persistent Memory Management

SF.PERS_MEM provides atomic write operations and transaction management according to the Java Card Runtime Environment Specification [16]. This supports FAU_ARP.1, FPT_FLS.1, and FDP_ROL.1[FIREWALL].

SF.PERS_MEM supports FDP_RIP.1[ABORT] together with SF.JCVM by halting the system in case of object creation in aborted transactions.

Low level write routines to persistent memory in SF.PERS_MEM perform checks for defect memory cells to fulfill FAU_ARP.1 and FPT_FLS.1.

8.2.12 SF.EDC: Error Detection Code API

SF.EDC provides an Java API for user applications to perform integrity checks based on a checksum on Java arrays [9]. The API throws a Java Exception in case the checksum in invalid. This supports FAU_ARP.1 and FPT_FLS.1.

8.2.13 SF.HW_EXC: Hardware Exception Handling

SF.HW_EXC provides software exception handler to react on unforeseen events captured by the hardware (hardware exceptions). SF.HW_EXC catches the hardware exceptions, to ensure the system goes to a secure state to fulfill FAU_ARP.1 and FPT_FLS.1, as well as to increase the attack counter in order to resist physical manipulation and probing to fulfill FPT_PHP.3.

8.2.14 SF.PID: Platform Identification

SF.PID provides a platform identifier. This platform identifier is generated during the card image generation. The platform identifier contains IDs for:

- NVM content (stored during romizing)
- Patch Level (stored during romizing, can be changed during personalization if patch is loaded)
- ROM code (stored during romizing)
- ROM code checksum (stored during romizing or during first TOE boot).

It identifies unambiguously the NVM and ROM part of the TOE. This feature supports FAU_SAS.1[SCP] by using initialization data that is used for platform identification.

8.2.15 SF.SMG_NSC: No Side-Channel

The TSF ensures that during command execution there are no usable variations in power consumption (measurable at e.g. electrical contacts) or timing (measurable at e.g. electrical contacts) that might disclose cryptographic keys or PINs. All functions of SF.CRYPTO except for SHA are resistant to side-channel attacks (e.g. timing attack, SPA, DPA, DFA, EMA, DEMA) (see FPR_UNO.1 and FPT_EMSEC.1).

8.2.16 SF.ACC_SBX: Secure Box

SF.ACC_SBX provides an environment to securely execute non-certified native code from third parties. SF.ACC_SBX ensures that only program code and data contained in the secure box can be accessed from within this secure box and therefore cannot harm, manipulate, or influence other parts of the TOE. This fulfills the SFRs FDP_ACC.2[SecureBox], FDP_ACF.1[SecureBox] and FMT_MSA.1[SecureBox].

Native code executed in the Secure Box is executed in User Mode. Access to the CPU mode, memory outside the Secure Box, the MMU segment table, and Special Function Registers which allow configuration of the MMU and allow System Management is prohibited for code executed in the Secure Box to fulfill FDP_ACF.1[SecureBox].

The MMU segment table to configure the MMU is part of the Secure Box which fulfils FMT_MSA.3[SecureBox]. This MMU segment table can be modified during the prepersonalization in accordance with FMT_MSA.3[SecureBox] to specify alternative settings for initially restrictive values for the MMU segment table. This supports FMT_SMF.1[SecureBox].

8.2.17 SF.MOD_INVOC: Module Invocation

SF.MOD_INVOC limits the invocation of code inside a Module to such Modules whose security attribute Module Presence has the restrictive default value ”present”. This fulfils the FMT_SMF.1[MODULAR-DESIGN], FMT_SMR.1[MODULAR-DESIGN], FMT_MSA.3[MODULAR-DESIGN] and FIA_UID.1[MODULAR-DESIGN]. Limiting the invocation to defined subjects S.APPLET, S.SD and S.JCRE fulfils the FDP_IFC.1[MODULAR-DESIGN] and FDP_IFF.1[MODULAR-DESIGN].

Throwing an exception in cases where the security attribute Module Presence has the value ”not present” fulfils FPT_FLS.1[MODULAR-DESIGN]. Deletion of a module may only be performed by S.ADEL which fulfils FMT_MSA.1[MODULAR-DESIGN]. The Modules are identified by their associated unique AIDs, which fulfils FIA_ATD.1[MODULAR-DESIGN] and FIA_USB.1[MODULAR-DESIGN] .

8.2.18 SF.SENS_RES: Sensitive Result

SF.SENS_RES ensures that sensitive methods of the Java Card API store their results so that callers of these methods can assert their return values. If such a method returns abnormally with an exception then the stored result is tagged as Unassigned and any subsequent assertion of the result will fail. This fulfills FDP_SDI.2[SENSITIVE_RESULT].

8.3 Protection against Interference and Logical Tampering

The protection of the TOE against Interference and Logical Tampering is implemented in software within the TOE and supported by the hardware of the micro controller.

The software protection of the TOE makes use of software security services which allow to detect and react on manipulation of the TOE. Two types of reactions are used: If invalid data from outside the TOE is detected then it is assumed that the TOE was used in a wrong way. This is indicated by an appropriate Status Word or Exception. Detected deviations from the physical operating conditions and inconsistencies of internal states and program flow however are considered to be an attack to the TOE. In such cases an internal Attack Counter is increased.

Typical software security mechanisms implemented in the TOE are e.g.:

- Complex patterned values are used instead of boolean values which are sensible to tampering (only one bit needs to be changed to manipulate a *false* into a *true*.
- Small random delays are inserted in the program flow to make successful physical interfering more difficult.
- Secret information like Keys or PINs are stored encrypted in the TOE. The Masterkey to decrypt these is not accessible during normal operation.
- Critical data is read after it has been written to non volatile memory.
- Enhanced cryptographic support is based on the certified Crypto Lib for DES, AES, RSA, ECC and random number generation.
- Critical values (like PINs) are compared timing-invariant. This prevents from side channel attacks.

Further protection against Tampering and Logical Interference is realized by the MMU implemented in hardware. The MMU is able to perform access control to all types of memory and the special functions registers depending on the current operation.

The TOE defines several MMU contexts which restrict access to card internal resources. The standard context used for normal operation has no access to the cryptographic coprocessor. The context for cryptographic operation has no access to the communication interfaces. One special context has write access to the Master Key in the TOE. Afterwards the Master Keys can only be read, but only from a dedicated context which is used to decrypt keys stored in the secure data store. In all other contexts the Master Key is not accessible.

Additionally Interference and Logical Tampering is prevented by hardware security services. JCOP 4 OS runs on a certified smart card HW platform which protects against bypass by physical and logical means such as:

- cryptographic coprocessors (for symmetric and asymmetric cryptography) protected against DPA and Differential Fault Analysis (DFA),
- enhanced security sensors for clock frequency range, low and high temperature sensor, supply voltage sensors Single Fault Injection (SFI) attack detection, light sensors, and
- encryption of data stored in persistent and transient memory.

8.4 Protection against Bypass of Security Related Actions

The TOE prevents bypassing security related actions by several software counter measures. Different mechanism are used depending on the software environment.

Generally all input parameter are validated and in case of incorrect parameters the program flow is interrupted. Such event is indicated by an appropriate Status Word or Exception. This prevents the TOE from being attacked by undefined or unauthorized commands or data.

Basic protection is contributed by implementation of following standards within the TOE:

- Java Applets are separated from each other as defined in the Java Card specifications [14], [16], [15]. The separation is achieved by implementation of the firewall which prevents Applets to access data belonging to a different Java Card context. Information sharing between different contexts is possible by supervision of the well defined Java Card Firewall mechanism implemented in the TOE.
- Access to security relevant Applications in the TOE (like Security Domains) is protected by the Secure Channel mechanism defined by GlobalPlatform [17]. The secure channel allows access to Applications only if the secret keys are known. Further protection

implemented in the TOE prevent brute force attacks on the secret keys of the Secure Channel.

The following mechanisms ensure that it is not possible to access information from the Java Layer without being authorized to do so.

- Status informations like Life Cycle of Applets or the Authentication State of a Secure Channel are stored in complex patterned values which protects them from manipulation.
- Correct order of Java Card Byte Code execution is ensured by the Virtual Machine which detects if Byte Code of a wrong context is executed.
- Correct processing of Byte Codes is ensured by checking at the beginning and end of Byte Code execution that the same Byte Code is executed.

Execution of native code in the TOE is protected by following mechanisms:

- Critical execution paths of the TOE functionality are protected by program flow and call tree protection. This ensures that it is not possible to bypass security relevant checks and verifications.
- Critical conditions are evaluated twice. This ensures that physical attacks on the compared values are detected during security relevant checks and verifications.
- The true case in if-conditions leads to the less critical program flow or to an error case. This prevents attacks on the program flow during security relevant checks and verifications.
- At the exit of critical loops it is checked that the whole loop was processed. This prevents from manipulation of the program flow and jumping out of the loop.
- Critical parameters are checked for consistency. This prevents from attacks with manipulated parameters.

9 Glossary

AID Application Identifier, an ISO-7816 data format used for unique identification of Java Card applications (and certain kinds of files in card file systems). The Java Card platform uses the AID data format to identify applets and packages. AIDs are administered by the International Standards Organization (ISO), so they can be used as unique identifiers.

AIDs are also used in the security policies (see ’Context’ below): applets’ AIDs are related to the selection mechanisms, packages’ AIDs are used in the enforcement of the firewall. Note: although they serve different purposes, they share the same name space.

APDU buffer The APDU buffer is the buffer where the messages sent (received) by the card depart from (arrive to). The JCRE owns an APDU object (which is a JCRE Entry Point and an instance of the javacard.framework.APDU class) that encapsulates APDU messages in an internal byte array, called the APDU buffer. This object is made accessible to the currently selected applet when needed, but any permanent access (out-of selection-scope) is strictly prohibited for security reasons.

applet The name is given to a Java Card technology-based user application. An applet is the basic piece of code that can be selected for execution from outside the card. Each applet on the card is uniquely identified by its AID.

applet deletion manager The on-card component that embodies the mechanisms necessary to delete an applet or library and its associated data on smart cards using Java Card technology.

context A context is an object-space partition associated to a package. Applets within the same Java technology-based package belong to the same context. The firewall is the boundary between contexts (see "current context").

current context The JCRE keeps track of the current Java Card System context (also called "the active context"). When a virtual method is invoked on an object, and a context switch is required and permitted, the current context is changed to correspond to the context of the applet that owns the object. When that method returns, the previous context is restored. Invocations of static methods have no effect on the current context. The current context and sharing status of an object together determine if access to an object is permissible.

currently selected applet The applet has been selected for execution in the current session. The Java Card RE keeps track of the currently selected Java Card applet. Upon receiving a SELECT command from the CAD or PCD with this applet’s AID, the Java Card RE makes this applet the currently selected applet over the I/O interface that received the command. The Java Card RE sends all further APDU commands received over each interface to the currently selected applet on this interface ([16], Glossary).

default applet The applet that is selected after a card reset ([16], §4.1).

installer The installer is the on-card application responsible for the installation of applets on the card. It may perform (or delegate) mandatory security checks according to the card issuer policy (for bytecode-verification, for instance), loads and link packages (CAP file(s)) on the card to a suitable form for the Java Card VM to execute the code they contain. It is a subsystem of what is usually called “card manager”; as such, it can be seen as the portion of the card manager that belongs to the TOE.

The installer has an AID that uniquely identifies him, and may be implemented as a Java Card applet. However, it is granted specific privileges on an implementation-specific manner ([16], §10).

interface A special kind of Java programming language class, which declares methods, but provides no implementation for them. A class may be declared as being the implementation of an interface, and in this case must contain an implementation for each of the methods declared by the interface (See also shareable interface).

Java Card RE The runtime environment under which Java programs in a smart card are executed. It is in charge of all the management features such as applet lifetime, applet isolation, object sharing, applet loading, applet initializing, transient objects, the transaction mechanism and so on.

Java Card RE Entry Point An object owned by the Java Card RE context but accessible by any application. These methods are the gateways through which applets request privileged Java Card RE services: the instance methods associated to those objects may be invoked from any context, and when that occurs, a context switch to the Java Card RE context is performed.

There are two categories of Java Card RE Entry Point Objects: Temporary ones and Permanent ones. As part of the firewall functionality, the Java Card RE detects and restricts attempts to store references to these objects.

Java Card RMI Java Card Remote Method Invocation is the Java Card System version 2.2 and 3 Classic Edition mechanism enabling a client application running on the CAD platform to invoke a method on a remote object on the card. Notice that in Java Card System, version 2.1.1, the only method that may be invoked from the CAD is the process method of the applet class and that in Java Card System, version 3 Classic Edition, this functionality is optional.

Java Card System Java Card System includes the Java Card RE, the Java Card VM, the Java Card API and the installer.

Java Card VM The embedded interpreter of bytecodes. The Java Card VM is the component that enforces separation between applications (firewall) and enables secure data sharing.

logical channel A logical link to an application on the card. A new feature of the Java Card System, version 2.2 and 3 Classic Edition, that enables the opening of simultaneous sessions with the card, one per logical channel. Commands issued to a specific logical channel are forwarded to the active applet on that logical channel. Java Card platform, version 2.2.2 and 3 Classic Edition, enables opening up to twenty logical channels over each I/O interface (contacted or contactless).

NVRAM Non-Volatile Random Access Memory, a type of memory that retains its contents when power is turned off.

object deletion The Java Card System version 2.2 and 3 Classic Edition mechanism ensures that any unreferenced persistent (transient) object owned by the current context is deleted. The associated memory space is recovered for reuse prior to the next card reset.

PCD Proximity Coupling Device. The PCD is a contactless card reader device.

PICC Proximity Card. The PICC is a card with contactless capabilities.

Secure Box The Secure Box is a construct which allows to run non certified third party native code and ensures that this code cannot harm, influence or manipulate the JCOP operating system or any of the applets executed by the operating system.

Secure Box Native Library The Secure Box Native Library is non certified third party native code running in the Secure Box.

shareable interface An interface declaring a collection of methods that an applet accepts to share with other applets. These interface methods can be invoked from an applet in a context different from the context of the object implementing the methods, thus “traversing” the firewall.

SIO An object of a class implementing a shareable interface.

subject An active entity within the TOE that causes information to flow among objects or change the system’s status. It usually acts on the behalf of a user. Objects can be active and thus are also subjects of the TOE.

transient object An object whose contents are not preserved across CAD sessions. The contents of these objects are cleared at the end of the current CAD session or when a card reset is performed. Writes to the fields of a transient object are not affected by transactions.

user Any application interpretable by the Java Card RE. That also covers the packages. The associated subject(s), if applicable, is (are) an object(s) belonging to the javacard.framework.applet class.

10 Acronyms

3DES Data Encryption Standard with 3 keys.

AES Advanced Encryption Standard.

AES CCM AES in Counter with CBC-MAC mode.

AP Application Provider.

APSD Application Provider Security Domain.

CAD Card Acceptance Device.

CRT Chinese Remainder Theorem.

Crypto Lib Crypto Library.

DAP Data Authentication Pattern.

DFA Differential Fault Analysis.

DPA Differential Power Analysis.

ECC Elliptic Curve Cryptography.

ECDAA Elliptic Curve Direct Anonymous Attestation.

GP GlobalPlatform.

HKDF HMAC based Key Derivation Function.

HMAC Keyed-Hash Message Authentication Code.

ICV Initial Chaining Vector.

ISD Issuer Security Domain.

MC FW Micro Controller Firmware.

MMU Memory Management Unit.

OS Operating System.

OSP Organizational Security Policy.

PKCC Public Key Crypto Coprocessor.

PP Protection Profile.

RAM Random Access Memory.

RF Radio Frequency.

ROM Read Only Memory.

RSA Rivest Shamir Adleman asymmetric algorithm.

SCP Smart Card Platform. It is comprised of the integrated circuit, the operating system and the dedicated software of the smart card.

SD Security Domain.

SFR Security Functional Requirement.

SPD Security Problem Definition.

SSD Supplementary Security Domain.

VASD Verification Authority Security Domain.

11 Bibliography

11 . 1 Evaluation documents

[1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, Version 3.1, Revision 5, CCMB-2017-04-001, April 2017.

[2] Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components, Version 3.1, Revision 5, CCMB-2017-04-002, April 2017.

[3] Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components, Version 3.1, Revision 5, CCMB-2017-04-003, April 2017.

[4] Common Methodology for Information Technology Security Evaluation, Evaluation methodology, Version 3.1, Revision 5, CCMB-2017-04-004, April 2017.

[5] Security IC Platform Protection Profile with Augmentation Packages, Registered and Certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-CC-PP-0084-2014, Version 1.0, 13 January 2014.

[6] Java Card System - Open Configuration Protection Profile, December 2017, Version 3.0.5, published by Oracle, Inc. (BSI-CC-PP-0099-2017).

[7] ICAO. Common criteria protection profile, machine readable travel document with ICAO application, basic access control, registered and certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference bsi-cc-pp-0055, rev 1.10, 25 march 2009.

[8] ANSSI. Application Note. Security Requirements for Post-Delivery Code Loading, Version 2.0, 23 January 2015, ANSSI-CC-NOTE-06/2.0.

11 . 2 Developer documents

[9] JCOP 4.7 SE051, User manual for JCOP 4.7 SE051, User Guidance and Administrator Manual, Revision 2.5, DocNo 581825, NXP Semiconductors, 24 June 2025.

[10] NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library (R1/R2/R3/R4), Security Target, BSI-DSZ-CC-1136, NXP Semiconductors, Revision 3.4, 9 December 2025.

[11] SE051, Plug & Trust Secure Element, Product data sheet, Revision 1.6, DocNo 577316, 27 October 2022.

[12] NXP. PUF Key derivation function specification, NXP Semiconductors, BUID, 2014.

[13] NXP Semiconductors, https://www.docstore.nxp.com.

11 . 3 Standards

[14] Oracle. Java Card 3 Platform, Application Programming Interface, Classic Edition, Version 3.0.5.

[15] Oracle. Java Card 3 Platform, Virtual Machine Specification, Classic Edition, Version 3.0.5.

[16] Oracle. Java Card 3 Platform, Runtime Environment Specification, Classic Edition, Version 3.0.5.

[17] GlobalPlatform. GlobalPlatform Card Specification 2.3.0, GPC_SPE_034, GlobalPlatform Inc., Oct 2015.

[18] GlobalPlatform. GlobalPlatform Card Technology Secure Channel Protocol 03’, Card Specification v 2.2 - Amendment D v1.1.1, July 2014.

[19] GlobalPlatform. Security Upgrade for Card Content Management - Amendment E v1.1, November 2016.

[20] GlobalPlatform Card Mapping Guidelines of Existing GP v2.1.1 Implementation on v2.2.1, January 2011.

[21] Bundesamt fuer Sicherheit in der Informationstechnik. AIS20/31: A proposal for: Functionality classes for random number generators, Bundesamt für Sicherheit in der Informationstechnik (BSI), Version 2.1, 2. December 2011.

[22] FIPS PUB 140-2: Security Requirements for Cryptographic Modules, Federal Information Processing Standards Publication 140-2, US Department of Commerce/ National Institute of Standards and Technology, 25. May 2001.

[23] FIPS PUB 186-4: Digital Signature Standard (DSS), US Department of Commerce/ National Institute of Standards and Technology, July 2013.

[24] FIPS PUB 197: Advanced Encryption Standard (AES), Federal Information Processing Standards Publication 197, US Department of Commerce/National Institute of Standards and Technology, 26. November 2001.

[25] NIST SP 800-38A: Recommendation for Block Cipher Modes of Operation: Methods and Techniques, National Institute of Standards and Technology, December 2001.

[26] ISO/IEC 14888-3: IT Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms, International Organization for Standardization, March 2016.

[27] ISO/IEC 9797-1: IT Security techniques - Message Authentication Codes (MACs) - Part 1: Mechanisms using a block cipher, International Organization for Standardization, 1999.

[28] RFC 5869: HMAC-based Extract-and-Expand Key Derivation Function (HKDF), Request For Comments, May 2010.

[29] ANSI X9.62: Public Key Cryptography for the Financial Services Industry: the Elliptic Curve Digital Signature Algorithm (ECDSA), American Standard for Financial Services, November 2005.

[30] TPM Rev. 2.0: Trusted Platform Module Library Specification, Family "2.0", Level 00, Revision 01.07 , March 2014.

[31] ISO 7816-3: Part 3: Cards with contacts - Electrical interface and transmission protocols, November 2006.

[32] ISO/IEC 14443-4 Cards and security devices for personal identification - Contactless proximity objects - Part 4: Transmission protocol, July 2008.

[33] NXP. UM10204, I2C-bus specification and user manual, Rev. 6, 4 April 2014.

12 Legal information

12.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

12.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this data sheet expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

12.3 Licenses

ICs with DPA Countermeasures functionality

LICENSED DPA COUNTERMEASURES™

NXP ICs containing functionality implementing countermeasures to Differential Power Analysis and Simple Power Analysis are produced and sold under applicable license from Cryptography Research, Inc.

12.4 Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Tables

Figures

Contents

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 18 December 2025