Security Target of GEOP02 on GSE20 Security Chip

V1.0

Shenzhen Goodix Technology Co., Ltd

Revision History

Date	Version	Comment
Oct 14 2025	1.0	ST Lite version

Table of Content

Document information 6

Glossary 6

1 ST Introduction 8

1.1 ST Reference 8

1.2 TOE Reference 9

1.3 TOE Overview 9

1.4 TOE Description 14

2 Conformance Claim 22

2.1 CC Conformance Claim 22

2.2 PP Claim 22

2.3 Package Claim 22

2.4 Conformance Claim Rationale 22

3 Security Aspects 27

3.1 Confidentiality 27

3.2 Integrity 27

3.3 Unauthorized Executions 27

3.4 Bytecode Verification 27

3.5 Card Management 27

3.6 Services 27

3.7 Miscellaneous 27

3.8 OS Management 28

3.9 Limited Mode 28

4 Security Problem Definition..... 29
4.1 Description of Assets..... 29
4.2 Description of Threats..... 30
4.3 Organizational Security Policies..... 33
4.4 Assumptions..... 33
5 Security Objectives..... 35
5.1 Security Objectives for the TOE..... 35
5.2 Security Objectives for the operational environment..... 38
5.3 Security Objectives Rationale..... 39
6 Extended Components Definition..... 41
6.1 Definition of FCS_RNG..... 41
6.2 Definition of FPT_EMSEC..... 42
6.3 Cryptographic Key Derivation (FCS_CKM.5)..... 43
7 Security Requirements..... 45
7.1 Security Functional Requirements..... 45
7.2 Security Assurance Requirements..... 73
7.3 Security Requirements Rationale..... 75
8 IC Composition rationale..... 85
8.1 Common Criteria rationale..... 85
8.2 Compatibility between Security Objectives (TOE and IC)..... 86
8.3 Compatibility between SFRs (TOE and IC)..... 86
8.4 Compatibility between security objectives for the environment (TOE and IC)... 90
9 TOE Summary Specification..... 91

9.1 Security Functionality of the TOE 91

9.2 Security Functions 93

10 Bibliography 97

10.1 Standards 97

10.2 Developer Documents 100

11 Legal and Contact Information 101

Document information

Information	Content
Keywords	Goodix, Security OS, GSE20, Secure Element, Crypto Library, Common Criteria, Security Target
Abstract	This document is the Security Target of the Goodix Security OS running on the Security Chip of the GSE20 family with IC Dedicated Software, developed and provided by Goodix Ltd. The Security OS conforms to Evaluation Assurance Level 5 of the Common Criteria for Information Technology Security Evaluation Version 3.1 Revision 5 with augmentations ALC_DVS.2 and AVA_VAN.5.

Glossary

AES	Advanced Encryption Standard
AP	Application Provider
API	Application Process Interface
APSD	Application Provider Security Domain
CASD	Controlling Authority Security Domain
CBC	Cipher Block Chaining Mode
CRC	Cyclic Redundancy Checks
CRT	Chinese Remainder Theorem

CTR	Counter Mode
DES/TDES	Data Encryption Standard/Triple Data Encryption Standard
DRNG	Deterministic Random Number Generation
ECB	Electronic Code Book Mode
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
ELF	Executable Load File*
ES	Embedded Software
HAL	Hardware Abstraction Layer
HCI	Host Controller Interface
OFB	Output Feedback Mode
OSCCA	China Office of State Commercial Cryptography Administration
OSM	OS Management
RAM	Random Access Memory
RNG	Random Number Generator
RSA	Rivest-Shamir-Adleman Algorithm
TRNG	True Random Number Generator

VA	Verification Authority

* Actual on-card container of one or more application’s executable code (Executable Modules). It may reside in Immutable Persistent Memory or may be created in Mutable Persistent Memory as the resulting image of a Load File Data Block.

1 ST Introduction

1.1 ST Reference

See title page.

1.2 TOE Reference

TOE Name: GEOP02 on GSE20 Security Chip v1.0

TOE Developer: Shenzhen Goodix Technology Co., Ltd

TOE Version: v1.0

1.3 TOE Overview

1.3.1TOE Introduction

The TOE is a Java Card System comprising the JCOS and the IC. It is a composite TOE with the Security Card Operating System (COS) running on the Goodix GSE20 Security Chip with IC Dedicated Software. The GSE20 Security Chip and associated IC Dedicated Software are certified to Common Criteria EAL6+ [CC3], comparable to a smart card controller.

The COS provides the following functionality:

- Virtual Machine Software [10] and a Runtime Environment [8],
- Common Application Programming Interface Software [9],
- Application Programming Interface for HCI [18],
- GlobalPlatform (GP) Software[11] (Secure Element configuration [17] and Common Implementation Configuration [30], no security claimed),
- seRoot, implements OS Patch Update/Config Software. This component ensures that only Goodix Authorized updates may be applied,
- Proprietary Application Programming Interface Software (Extension API) [37], including OSCCA algorithms (SM2, SM3, SM4, no security claimed)[19][20][21],
- Proprietary Native Application, Yula, as a NFC Tag application (no security claimed),
- EDA (Event Driven Architecture) for task management,
- Kernel for managing system resources and communication between the IC and OS,
- Limited Mode, provide functions to defend against continuous attacks,
- Exception Handler, preserve TOE into secure state when physical attacks are detected,

With the purpose of implementing the functionalities related to GlobalPlatform software, in this ST are claimed SFRs iterated with /GP, /GP-ELF and /GP-KL which do not come from [[JCPP]] but they are inspired by [GPC_SE_PP]. The SFRs iterated as /GP add functionalities for the security policies defined in /GP-ELF and /GP-KL. These iterated SFRs, /GP-ELF and /GP-KL, add functionalities for ELF loading information flow control

policy, covering INSTALL (see [11] section 11.5) and LOAD (see [11] section 11.6) commands, and for Data & Key loading information control policy, covering PUT KEY (see [11] section 11.8) and STORE DATA (see [11] section 11.11) commands, respectively.

Figure 1 provides an overview of the TOE and its interfaces.

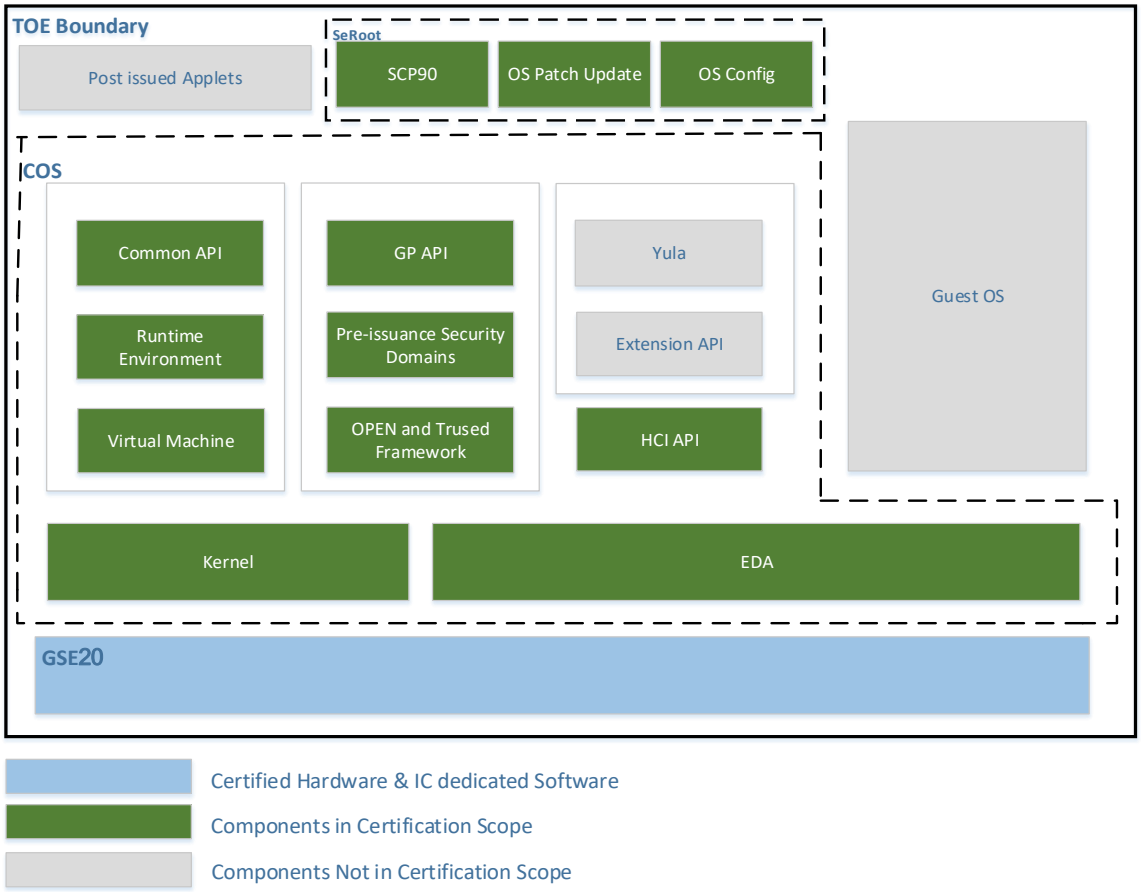

Figure 1 TOE Overview

1.3.2TOE Type and Usage

The TOE is a Smart Card Platform (IC and OS) along with the native applications and the Java Card System.

The TOE is not considered to be a final product. It is delivered to the end user and then, the end user will be able to load, install, instantiate and execute the off-card verified Javacard applets on the upper layer. The final TOE developers are expected to perform the development based on the following specifications:

- ICAO MRTD Doc 9303 [23]
- China financial integrated circuit card specification (JR/T 0025.1 2018, JR/T 0025.13 2018, JR/T 0025.14-2018) [25]
- Technical specifications on IC card for urban public transport ticket (JT/T 978.2-2023) [26]
- Strongbox [27]

- Car Connectivity Consortium Digital Key Release 3 [28]

The Security Card Operating System (COS) implements GlobalPlatform functionality allowing the installation of various applications, including but not limited to access control, mobile transaction, digital ID and digital car key.

1.3.3TOE Security Functionality

The TOE provides the following major security functionalities:

- GSE20 security chip provides cryptographic functions and security features to protect the circuits and its IC Dedicated Software from physical attacks, side channel attacks and perturbation attacks.
- Cryptographic algorithms and functionality:
  - TDES for encryption/decryption (CBC) and MAC generation and verification (3-key 3DES, Retail-MAC, CBC-MAC)
  - AES (Advanced Encryption Standard) for encryption/decryption (GCM, CBC, CCM, CFB, CTR), HMAC algorithms and MAC generation and verification (CMAC)
  - RSA and RSA CRT for encryption/decryption and signature generation and verification
  - RSA and RSA CRT key generation
  - ECC over GF(p) for signature generation and verification (ECDSA)
  - Random number generation conforming to class PTG.2 and DRG.3 of AIS 20/31 [6]
  - Key Derivation Function

The TOE implements more cryptographic mechanisms that are not in the evaluation scope because these cryptographic mechanisms are not conformant with SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms v1.3 [ACM]. This is further described in section 1.4.2.5.

- Java Card 3.1.0 functionality:
  - Java Card Virtual Machine for bytecode execution
  - Transient and persistent memory management for applets
  - Applet firewall protection
  - Access control rules between applets and the JCRE
  - Javacard wrapper layer for native implementations
  - Garbage Collection
  - Support for Extended Length APDUs

- Sensitive result, Sensitive array, array view
- Oneshot object

- GlobalPlatform 2.3.1 and GP admendments functionality:
  - Loading and installation of Java Card packages
  - Java CAP file deletion
  - Java applet deletion
  - Supplementary Security Domains (APSD and CASD) creation
  - Applet and Security Domain association
  - Key installation
  - Applet signature verification
  - CVM (PIN) Management
  - SCP 02 and SCP 03 secure channels
  - Delegated Management, DAP (RSA up to 4096 bits and ECC up to 512 bit)
  - Compliance to Common Implementation Configuration [30] (security not claimed)
  - Compliance to Secure Element configuration [17] (security not claimed)
  - GP framework and Amendment A, C, D, E.
- HCI communication functionality
  - HCI APIs for HCI communication
- Goodix proprietary functionality
  - EDA framework for task management
  - SeRoot for OS Patch update and OS configuration over SCP 90 secure channel (only available for an authorized entity)
  - Kernel for managing system resources and communication between IC and COS.
  - Yula NFC Tag application (security not claimed)
  - Proprietary Application Programming Interface Software (Extension API) [37] (security not claimed)

1.3.4Required non-TOE Hardware/Software/Firmware

The end users of the TOE use the TOE with the loaded applets as a SE. These users communicate with the TOE with SPI interfaces, ISO7816 and I2C. Therefore, the communication device supporting these interfaces is needed for using the TOE.

The administrators of SEs configure and update the TOE with seRoot, install additional applets or delete applets with CCM functionality. These users require the same equipment as end-users.

The developers of Java Card applets load and execute the applets on the TOE with the development tools and byte code verifier for the development.

1.4 TOE Description

1.4.1Physical scope of the TOE

The TOE consists of Security OS, IC hardware, IC Dedicated Software and guidance documentation.

Category	Component	Version	Delivery form
IC hardware	Security IC	A1	Module
IC Dedicated Software	IC Dedicated Software	v1	binary in ROM and Flash
IC Crypto Library	IC Crypto Library	v1.1	binary in ROM and Flash
COS framework	Runtime Environment Virtual Machine Common API HCI API	03.01.0000	binary in ROM and Flash
GlobalPlatform	GP API GP APDU		binary in ROM and Flash
Proprietary software	Yula NFC Tag application *		binary in ROM and Flash
	EDA framework		binary in ROM and Flash
	Kernel		binary in ROM and Flash
Proprietary Applet	seRoot (OS Patch Update, OS Configuration)		binary in ROM and Flash
User Manual	GEOP02 User Manual [32]	1.9	.pdf file
	GEOP seRoot User Manual[33]	0.7	.pdf file
	GEOP02 Preparative Procedures [34]	1.8	.pdf file
	GEOP02 Operational User Guidance [35]	1.7	.pdf file
	GEOP02 Security Guidance [36]	1.5	.pdf file

* This is the only component from the table which is not part of the evaluation scope. See detail software components in Section 1.4.2

Table 1 TOE physical scope

The security IC is delivered to the client as module with IC software, COS, which includes Proprietary software, and seRoot as proprietary applet in the ROM and Flash using a secure delivery method with security seals. The user manuals are delivered to the client with emails using PGP signed and encrypted packages.

The TOE can be identified by the TOE ID (see Table 2). The TOE ID can be obtained by using getVersion command (see [32]).

Data Element	Length (byte)	Value	Description
IC firmware version	2	0100	IC firmware version is 0100
OS Rom version	1	03	
Major Patch version	1	01	
Minor Patch version	2	0000	
RFU	32	N.A.	Internal info
CID	16 bytes	XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX	Chip ID, Different value for each chip
RFU	27	N.A.	Internal info

Table 2 TOE ID

1.4.2Logical scope of the TOE

The certification of this TOE is a composite certification. The certificate of the underlying hardware platform GSE20, which is part of this TOE, is re-used. In the following sections more detailed descriptions of the TOE components are provided. In the description 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.

1.4.2.1Security IC

The security IC, GSE20, is a high-security C040 System on Chip with ARM SC300 processor, AES, (T)DES cryptographic engines and a GRACE2 co-processor for RSA, ECC and OSCCA crypto operations. It contains RAM, ROM and Flash which protect the confidentiality and integrity of the stored data, MMU for memory protection, DRNG and TRNG for random number generation and other peripherals like DMA, I2C, ISO7816 and SPI. It also has an active shield and sensors for the detecting physical or environmental attacks. The security IC is certified according to Common Criteria EAL6+ [CC3].

1.4.2.2IC Dedicated Software

The IC Dedicated Software is certified in the scope of the security IC.

1.4.2.3Security OS and GlobalPlatform Software

Security OS consists of Native OS, JCVM, JCRE, JCAPI and GP framework. JCVM, JCRE, JCAPI and GP Software are implemented according to the Java Card Specification Version 3.1.0 [8][9][10] and Global Platform Specification v2.3.1 [11].

Global Platform Software consists of GP framework and Amendment A v1.2 [13], C v1.3 [14], D v1.2 [15], E v1.1 [16]. The following GP components are excluded from the certification:

- Secure Element configuration [17].
- Common Implementation Configuration [30].

Security OS components version can be identified by using the getVersion command (see [32]). This command returns the platform identification data, which includes the Chip ID, ROM version, Security OS version, Security OS patch version. These are the relevant return values for identification purposes of the TOE, but the command provides more information. GEOP02 version is a data string that allows to identify the Security OS component.

The specific versions of the components are described in [32].

1.4.2.4Proprietary Software

The TOE implements the proprietary software Kernel, EDA, seRoot.

The specific versions of the components are described in [32].

seRoot mainly provides OS Patch Update and OS Config. OS Patch Update is used to update TOE securely. And OS Config provides a method to set up the initial states, pre-personalization data, features configurations of the TOE securely.

EDA provides a task scheduler and task management functionalities, and it also separates resources from different tasks.

Kernel manages system resources and communication between the IC and OS.

The TOE provides a Limited Mode to prevent continuous physical attacks.

The TOE uses SCP90, a proprietary Secure Channel Protocol designed for GSE20, which ensures the security of information transmission between the terminal and the chip.

1.4.2.5Non-evaluated features

Yula NFC Tag Application and its Javacard API & Extension API are excluded from evaluation. Yula NFC Tag Application is not directly responsible for security. It primarily serves as an interface for communication between the NFC-enabled device and the tag. The actual security mechanisms are implemented at a lower level.

The TOE implements cryptographic mechanisms and proprietary interfaces for providing those mechanisms as a service (OSCCA algorithms). However, these mechanisms are out of the evaluation scope and their interfaces do not compromise the security of the TOE. Also, the hashing algorithms (SHA1, SHA-224, SHA-256, SHA-384, SHA-512) are out of the evaluation scope and their interfaces do not compromise the security of the TOE either.

The following list covers cryptographic mechanisms that are supported by the TOE but not conformant with SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms v1.3 [ACM]:

KDF

- ALG_KDF_DPI_MODE with any length within the specification limit [2]
- ALG_KDF_FEEDBACK_MODE with any length within the specification limit [2]
- ALG_KDF_COUNTER_MODE with any length within the specification limit [2]
- ALG_KDF_ICAO_MRTD with any length within the specification limit [23]

Symmetric encryption/decryption

- ALG_AES_ECB_ISO9797_M1, ALG_AES_ECB_ISO9797_M2 and ALG_AES_ECB_PKCS5 with cryptographic key sizes 128, 192 and 256 bits
- ALG_DES_ECB_ISO9797_M1, ALG_DES_ECB_ISO9797_M2, ALG_DES_ECB_NOPAD, ALG_DES_ECB_PKCS5 with cryptographic key sizes 112 and 168 bits
- ALG_AES_BLOCK_128_ECB_NOPAD with cryptographic key sizes 128, 192 and 256 bits
- ALG_RSA_NOPAD with any key length that is multiple of 64 from 512 to 1900 bits
- ALG_RSA_PKCS1 and ALG_RSA_PKCS1_OAEP with key length that is a multiple of 64 between 512 and 1900 bits
- ALG_DES_CBC_ISO9797_M1, ALG_DES_CBC_ISO9797_M2, ALG_DES_CBC_NOPAD, ALG_DES_CBC_PKCS5 with cryptographic key size 112 bits
- 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_MAC4_NOPAD, ALG_DES_MAC8_ISO9797_1_M1_ALG3, ALG_DES_MAC8_ISO9797_1_M2_ALG3, ALG_DES_MAC8_ISO9797_1_M1, ALG_DES_MAC8_ISO9797_1_M2, ALG_DES_MAC8_NOPAD, ALG_DES_MAC4_PKCS5, ALG_DES_MAC8_NOPAD with cryptographic key size 112 bits

Digital signature

- ALG_RSA_SHA_PKCS1, ALG_RSA_SHA_PKCS1_PSS, ALG_RSA_SHA_ISO9796 with key length that is a multiple of 64 between 512 and 1900 bits
- 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 with key length that is a multiple of 64 between 512 and 1900 bits
- ALG_ECDSA_SHA, ALG_ECDSA_SHA_224, ALG_ECDSA_SHA_256, ALG_ECDSA_SHA_384, ALG_ECDSA_SHA_512 with key sizes 112, 128, 160, 192 and 224 bits

Key establishment

- ALG_EC_SVDP_DH, ALG_EC_SVDP_DH_KDF, ALG_EC_SVDP_DHC, ALG_EC_SVDP_DHC_KDF, ALG_EC_SVDP_DHC_PLAIN, ALG_EC_SVDP_DH_PLAIN_XY, ALG_EC_SVDP_DH_PLAIN, ALG_EC_PACE_GM with key sizes 112, 128, 160, 192, 224, 256, 384, 512 and 521 bits

Key generation

- ECC Families SM, FIPS, NIST with key sizes 112, 128, 160, 192 and 224 bits and Brainpool with key sizes 112, 128, 160, 192 and 224 bits
- The ECC secp256k1, Brainpool-p256t, Brainpool-p384t1 and Brainpool-p512t.
- RSA-ND and RSA-CRT with key length that is a multiple of 64 between 512 and 1900 bits

These cryptographic mechanisms are not on the scope of evaluation because of the high assurance level of the TOE and due to the fact that these cryptographic mechanisms are vulnerable to high potential attacks. On the other hand, the cryptographic mechanisms that are implemented by the TOE and conformant with [ACM] are claimed in section 7.1.1.1.

1.4.3Interfaces of the TOE

1.4.3.1Electrical and Physical interface

These interfaces are provided by the certified security IC.

1.4.3.2Logical interface

The logical interface of the TOE is composed of the following:

- Javacard API interface [9]
- GP API interface [12]
- GP APDU command [11]
- HCI API interface
- seRoot APDU command [33]
- Bytecodes
- Yula NFC Tag API Interface [32] (security not claimed)
- Yula NFC command [32] (security not claimed)
- Extension API [37] (security not claimed)

1.4.4Form of Delivery

The Security OS is delivered embedded in the IC in wafer form to the applet developer according to phase 5 from [[JCPP]]. The delivery package will be sealed with secure tape. The delivery process will also be trackable with courier’s signature. This is in the scope of the IC certification.

SCP and COS keys are securely delivered based on pre-shared secret.

The user guidance and datasheet documents are delivered in electronic form to the user as encrypted and signed email attachment.

1.4.5TOE Life Cycle

The TOE development and production life cycle is scheduled in phases, which are defined in the Java Card Protection Profile [[JCPP]].

The Security OS is developed in Phase 1 “Security Embedded Software Development”. At the end of Phase 1, the TOE send the Security OS to Goodix hardware team, in a secure manner, to be programmed in Phase 3.

Phase 2 IC Development, Phase 3 IC Manufacturing as well as Phase 4 IC Packaging of this life cycle are evaluated during IC certification.

In Phase 2 IC Development of GSE20, access to sensitive design data of GSE20 is restricted to who are involved in the development of the product.

In Phase 3 IC Manufacturing, the wafer of GSE20 is produced and tested on wafers. The confidentiality and integrity of any design and configuration data in this phase will be ensured. This includes secure treatment and insertion of configuration data as well as manufacturing data, which are generated by Goodix.

In Phase 4 IC Packaging, the GSE20 is embedded into packages. The part of IC Dedicated Software is programmed into the Flash. And the TOE (or part of) can also be loaded to the user Flash area in this phase.

In Phase 5, the Composite Product Integrator, the Goodix Javacard Team, pre-personalize the Security OS and conduct tests in the same packaging and testing environment as Phase 4. Then the TOE is delivered to the client in a secure manner, which is evaluated during IC certification.

The TOE is personalized in Phase 6, if necessary. This is out of this certification scope.

In Phase 7, the TOE provides the full set of security functionalities to avoid abuse of the product by untrusted entities.

Note: User Applet development is outside the scope of this certification. Applets can be loaded into Flash memory. Applet loading into Flash memory can be done in phases 5, and 6. Applet loading in phase 7 is also allowed. This post-issuance loading of applets is allowed (except the native applets). The certification is only valid for platforms that return the platform identification data as stated in Table 1.

During Phases 1 to 3, the objectives for the environment 3 are covered by the developer’s security measures. During phases 4 to 7, the TOE protects itself with its own Security functions in the environment. But additional requirements for the environment must be followed (OE.Resp-Appl, OE.USE_DIAG).

The different sites where the TOE has been developed are presented in the following Table 3. In this table is related each site with its corresponding activity carried out and the phase of the TOE Life Cycle.

Site address	Phase of development	Activity carried out
Goodix Technology Co., Ltd. Central Creative Office Building (room 2104-2108), 33 Luoyu Road, Hongshan District, Wuhan, Hubei province, P.R. China	Phase 1 – Security IC Embedded Software Development (Java Card System) Phase 5 – Composite Product Integrator	TOE development and testing; Data server host; Product key management and shipment
Goodix Technology Co., Ltd. Room 502, 5rd floor of Puruan Building (west wing), No.2 Boyun Road, Pudong New Area, Shanghai, P.R. China	Phase 2 – Security IC development	TOE CP & FT program development TOE failure analysis Test samples and device management
STATS ChipPAC Semiconductor (Jiangyin) Col,Ltd. (JSCC) No.78 Changshan Road, Jiangyin, Jiangsu Province, P.R.China	Phase 3 – Security IC Manufacturing, Phase 4 – Security IC Packaging	Receiving, checking and storing wafers Wafer testing Wafer dicing Module packaging Module testing Module warehousing and dispatch Scraps collection and shipment
SJ Semiconductor Co.Ltd (SJSemi) 6# Dongsheng West Road Jiangyin City Jiangsu Province P.R. China	Phase 3 – Security IC Manufacturing, Phase 4 – Security IC Packaging	Receiving, checking and storing wafers Wafer dicing Module packaging Module testing Module warehousing and dispatch Scraps collection and shipment
Goodix Shenzhen finished-good warehouse (FGWH) 1st floor, Block A of Aerospace Micromotor building, No.7 Langshan #2 Road, Nanshan District, Shenzhen, Guangdong Province, P.R.China	Phase 2 –Security IC development	Shenzhen FGWH: Finished goods warehousing and delivery TOE Scrapping (the actual scrapping process is taken place at the contracted third-party premises) Defect module collection (for the failure analysis)

Table 3 Sites involved in TOE development

2 Conformance Claim

2.1 CC Conformance Claim

This Security Target (ST) and the TOE claim conformance to Common Criteria Version 3.1 Part 1 [CC1], Part2 [CC2] and Part 3 [CC3]:

- Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model, Version 3.1, Revision 5, April 2017, CCMB-2017-04-001
- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-003

Conformance of this ST is claimed for: Common Criteria Part 2 extended and Common Criteria Part 3 conformant.

2.2 PP Claim

The TOE is a composite one which relies on a certified IC and its dedicated software. This already-certified product claimed demonstrable conformance to PP0084 with their own SPD, objectives and SFRs. It is stated that the current ST will not discuss any information related to the certified IC. Refer to [31] for the IC ST.

The Security Target claims demonstrable conformance to the Java Card System Protection Profile – Open Configuration [[JCPP]]. The augmentation packages from Appendix 2 of [[JCPP]] “Key Derivation Functions”, “Sensitive Result” and “Sensitive Array” are claimed in this ST.

2.3 Package Claim

The assurance level for the TOE is EAL5 augmented with the components ALC_DVS.2 and AVA_VAN.5. The evaluation assurance level exceeds the requirement claimed by the [[JCPP]].

2.4 Conformance Claim Rationale

All sections of this Security Target, in which security problem definition, objectives and security requirements are defined, clearly state which of these items are taken from [[JCPP]] and which are added in this Security Target. Therefore, the rationales for the items from [[JCPP]] are not repeated here. Moreover, all additionally stated items in this Security Target do not contradict the items included from the [[JCPP]]. In case refinement or deletion of the items from the [[JCPP]], additional justification is

provided in the corresponding section of the ST. The operations done for the SFRs taken from [[JCPP]] are also clearly indicated.

The differences between this ST and the claimed Protection Profile are described in the following sections. These considerations show that the Security Target correctly claims demonstrable conformance to [[JCPP]].

2.4.1TOE Type Rationale

The TOE type as stated in Section 1.3.2 of this ST extends to the TOE type of the PP as stated in Section 2.1 of [[JCPP]] namely a Java Card platform, implementing the Java Card Specification Version 3.1.0 [8][9][10]. The TOE includes the IC HW in scope, security functionality attached to the Card Manager, OS management capabilities and continuous attacks countermeasures.

2.4.2Security Problem Definition Rationale

All the items of the security problem definition defined in Section 5 of [[JCPP]] are taken into this Security Target except that T.INSTALL and T.DELETION in [[JCPP]] are refined by T.UNAUTHORISED-CARD-MNGT which extends more threats related to card management. In addition, the following security problems are introduced in this Security Target. All the refined and introduced security problems are additions that [[JCPP]] allows.

T.COM-EXPLOIT is included to cover communication channels attacks.

T.LIFE-CYCLE is included to cover content management attacks.

All the previous are included due to the inclusion of the Card Manager in the scope of the TOE.

The threat T.UNAUTHORIZED_OS_MNGT is introduced for OS update and config which is additional functionality [[JCPP]] allows, as the final impact makes OS management capabilities more restrictive than the PP covers.

The threat T.EXCEPTION-COUNTER and T.LIMITED-MODE are included for the Limited Mode which is additional functionality [[JCPP]] allows as the additional threat guarantees that additional layers of protection against continuous attacks are implemented.

The assumption A.Process-Sec-IC and A.Resp-Appl are taken from the underlying certified secure IC [31] which are compliant to the Security IC PP [ICPP]. The assumptions A.Resp-Appl in this Security Target includes an application note to further clarify the application context which conforms to [ICPP]. These assumptions are allowed by [[JCPP]].

2.4.3SO and SOE Rationale

All the security objectives defined in Section 6 of [[JCPP]] are taken into this Security Target, except O.LOAD, O.INSTALL and O.DELETION are refined by O.CARD-MANAGEMENT. All the following introduced security objectives are additions to [[JCPP]].

OE.CARD-MANAGEMENT, OE.SCP.RECOVERY, OE.SCP.SUPPORT and OE.SCP.IC in [[JCPP]] Section 6.2 are replaced by O.CARD-MANAGEMENT, O.SCP.RECOVERY, O.SCP.SUPPORT and O.SCP.IC in this ST. O.SCP.RECOVERY, O.SCP.SUPPORT, and O.SCP.IC are objectives for the TOE as the Smart Card Platform belongs to the TOE for this evaluation. O.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.

The optional packages O.SENSITIVE_RESULTS_INTEG and O.SENSITIVE_ARRAYS_INTEG in [[JCPP]] Appendix 2 are included. Their rationales are defined in the PP.

O.AUTH-OS-MNGT is included for the OS configuration that [[JCPP]] allows since the objective claims OS secure management.

O.SECURE-LOAD-ACODE, O.SECURE-AC-ACTIVATION and O.TOE-IDENTIFICATION are included in accordance to [JIL_SRCL] to guarantee the secure updating functionality.

O.EXCEPTION-COUNTER and O.LIMITED-MODE are included for the Limited Mode functions. These contribute to harden the resistance of the TOE to continuous non-detected attacks.

O.DOMAIN-RIGHTS is included to ensure the security and integrity of the APSD keys by restricting access and modification rights exclusively to the AP.

O.COMM_AUTH, O.COMM_INTEGRITY and O.COMM_CONFIDENTIALITY are included to ensure a secure communication between the TOE and the origin of the card management.

The ST introduces the following additional security objectives for the environment: OE.Process_Sec_IC, OE.Resp-Appl,

OE.Process_Sec_IC, OE.Resp-Appl are from the Security IC [ICPP] 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 [ICPP]. These security objectives for the environment are allowed by [[JCPP]].

2.4.4Security Functional Requirement Statement

The Security Functional Requirements for the Java Card component are taken from the Java Card PP [[JCPP]] except for the following groups of exceptions:

The relationship between the ST and the Java Card PP is described hereafter. The relationship between SFRs uses the following notation:

- **Refinement (R)**: The element in the ST refines the corresponding [[JCPP]] element. New names are given between brackets and added to the list of elements.

- **Addition (A)**: The element is newly defined in the ST; it is not present in [[JCPP]] and does not affect it.

- **x**: The element is present in [[JCPP]].

All the mandatory SFRs of [[JCPP]] are relevant to the TOE as shown in the table below. All the operations performed on the Java Card SFRs are appropriate for the TOE.

TOE’s SFRs	[[JCPP]]	Relationship with [[JCPP]]
R: FDP_ITC.2/GP-ELF (Editorial Refinement)	x	FDP_ITC.2/Installer
R: FMT_SMR.1/GP (Editorial Refinement)	x	FMT_SMR.1/Installer
R: FPT_FLS.1/GP (Editorial Refinement)	x	FPT_FLS.1/Installer
R: FCO_NRO.2/GP (Editorial Refinement)	x	FCO_NRO.2/CM
R: FDP_IFC.2/GP-ELF (Editorial Refinement)	x	FDP_IFC.2/CM
R: FDP_IFF.1/GP-ELF (Editorial Refinement)	x	FDP_IFF.1/CM
R: FDP_UIT.1/GP (Editorial Refinement)	x	FDP_UIT.1/CM
R: FIA_UID.1/GP (Editorial Refinement)	x	FIA_UID.1/CM
R: FMT_MSA.1/GP (Editorial Refinement)	x	FMT_MSA.1/CM
R: FMT_MSA.3/GP (Editorial Refinement)	x	FMT_MSA.3/CM
R: FMT_SMF.1/GP (Editorial Refinement)	x	FMT_SMF.1/CM
R: FTP_ITC.1/GP (Editorial Refinement)	x	FTP_ITC.1/CM
FIA_UAU.1/GP		A
FIA_UAU.4/GP		A
FPT_TDC.1/GP		A
FDP_ITC.2/GP-KL		A
FDP_IFC.2/GP-KL		A
FDP_IFF.1/GP-KL		A
FDP_ACC.2/OSM		A
FDP_ACF.1/OSM		A
FDP_UIT.1/OSM		A
FMT_MSA.3/OSM		A
FMT_SMF.1/OSM		A
FTP_ITC.1/OSM		A
FPT_FLS.1/OSM		A
FDP_ACF.1/LM		A

FDP_ACC.2/LM		A
FMT_MSA.1/LM		A
FMT_MSA.3/LM		A
FMT_SMF.1/LM		A
FIA_UID.1/LM		A
FIA_UAU.1/LM		A
FPT_PHP.3		A
FCS_RNG.1/PTG.2		A
FCS_RNG.1/DRG.3		A
FPT_EMSEC.1		A

Table 4 SFRs' Consistency Statement

Application note: /GP, /GP-ELF and /GP-KL SFRs are inspired in [GPC_SE_PP]. However, no conformance is claimed to the referred PP.

FIA_UAU.1/GP and FIA_UAU.4/GP claim additional authentication capabilities of the TOE to guarantee it's secure management by authorized and authenticated users. FDP_ITC.2/GP-KL, FDP_IFC.2/GP-KL and FDP_IFF.1/GP-KL model the Key Loading policy, which is under the control of the Card Manager (within the scope of the TOE). FPT_TDC.1/GP is added to prevent misinterpretation of data issued through the INSTALL, LOAD, PUT KEY and STORE DATA commands sent to the card. Therefore, none of the additional SFRs contradict the statements and the SPD of the claimed [[JCPP]].

The rationale for all refined SFRs is included in the form of an Application Note for each in section 7. SFRs from [[JCPP]] not covered by the table above are kept as defined by the PP.

OSM stands for OS Management and covers the secured management capabilities of the OS including configuration and update. These management capabilities must be authenticated through a secure channel and are controlled under an access control policy. This additional feature has no impact on the [[JCPP]] coverage since the included SFRs do not contradict none of the statements from it.

LM stands for Limited Mode and covers under which circumstances is the mode triggered, what are the functions available once entered and who can restore to an operational mode. This additional feature has no impact on the [[JCPP]] coverage since the included SFRs do not contradict none of the statements from it.

FPT_PHP.3 is included to provide resistance to physical attacks covered by O.SCP.IC.

FCS_RNG.1/PTG.2 and FCS_RNG.1/DRG.3 are iterated to cover the guarantee for both physical and deterministic random number sources.

FPT_EMSEC.1 does not contradict none of the SFRs from the [JCPP] and increases physical resistance requirements provided by the IC by mitigating intelligible emanations.

3 Security Aspects

This chapter describes the main security issues of the Java Card System and its environment, security aspects, based on [[JCPP]]. All security aspects described in [[JCPP]] section 4 are applied. Additional security aspects are introduced in section 3.8 and 3.9.

3.1 Confidentiality

The security aspects #.CONFID-APPLI-DATA, #.CONFID-JCS-CODE and #.CONFID-JCS-DATA of stated in [[JCPP]] Section 4.1 are applied here as well.

3.2 Integrity

The security aspects #.INTEG-APPLI-CODE, #.INTEG-APPLI-DATA, #.INTEG-JCS-CODE, and #.INTEG-JCS-DATA in [[JCPP]] Section 4.2 are applied here as well. In addition, the following security aspect is introduced:

3.3 Unauthorized Executions

The security aspects #.EXE-APPLI-CODE, #.EXE-JCS-CODE, #.FIREWALL, and #.NATIVE stated in [[JCPP]] Section 4.3 are applied here as well.

3.4 Bytecode Verification

The security aspect #.VERIFICATION stated in [[JCPP]] Section 4.4 are applied here as well.

3.5 Card Management

The security aspect #.CARD-MANAGEMENT, #.INSTALL, #.SID, #.OBJ-DELETION and #.DELETION stated in [[JCPP]] Section 4.5 are applied here as well.

3.6 Services

The security aspects #.ALARM, #.OPERATE, #.RESOURCES, #.CIPHER, #.KEY-MNGT, #.PIN-MNGT, #.SCP and #TRANSACTION stated in [[JCPP]] Section 4.6 are applied here as well.

3.7 Miscellaneous

The security aspect #.INTEG-APPLI-DATA-PHYS in [[JCPP]] Appendix 2 are applied here as well.

3.8 OS Management

#.OSM The TOE allows only authorized entity to update or configure the Security OS. While performing OS update, the TOE ensures that only authenticated OS Patch can be installed with an atomic operation.

3.9 Limited Mode

#.LM If the Exception Counter reaches the limit, the TOE enters Limited Mode for performing a limited set of functions (e.g. reset the Exception Counter or read audit information.)

4 Security Problem Definition

This chapter describes the security problem definition of the TOE based on [[JCPP]]. All assets, threats, organizational security policy and assumptions defined in [[JCPP]] section 5 are applied. Additional assets are introduced in section 4.1.1 and 4.1.2.

4.1 Description of Assets

4.1.1User Data

The user data assets D.APP_CODE, D.APP_C_DATA, D.APP_KEYs and D.PIN described in Section 5.1.1 of [[JCPP]] are the assets of the TOE.

D.APP_I_DATA	Integrity sensitive data of the applications, like the data contained in an object, a static field of a package, a local variable of the currently executed method, the CVM security attributes (such as CVM value, CVM State, CVM Retry Limit, and CVM Retry Counter), or a position of the operand stack. To be protected from unauthorised modification
D.ISD_KEYS	Refinement of D.APP_KEYS of [[JCPP]]. ISD cryptographic keys needed to perform card management operations on the card. To be protected from unauthorised disclosure and modification.
D.APSD_KEYS	Refinement of D.APP_KEYS of [[JCPP]]. APSD cryptographic keys 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 unauthorised disclosure and modification.
D.CASD_KEYS	Refinement of D.APP_KEYS of [[JCPP]]. CASD cryptographic keys needed to establish Secure Channels with the CA and to decrypt confidential content for APSDs. To be protected from unauthorised disclosure and modification

4.1.2TSF Data

The TSF assets D.API_DATA, D.CRYPTO, D.JCS_CODE, D.JCS_DATA and D.SEC_DATA described in Section 5.1.2 of [[JCPP]] are the assets of the TOE. The TOE also has the following assets.

D.TOE_ID	TOE Identification Data for identifying the TOE. To be protected from unauthorized modification.
D.OS_IMAGE	The update image of the Security OS. To be protected from unauthorized disclosure and modification.
D.CONFIG_DATA	The OS configuration. To be protected from unauthorized disclosure and modification.
D.EXCEPTION_COUNTER	The exception counter used for attack detection. When a potential attack is detected the exception counter is updated up to a limit. Once its limit is reached, the TOE is put into the limited mode. To be protected from unauthorized modification.
D.GP_REGISTRY	The information resource for Card Content management. The GP Registry contains information for managing the card, as well as Executable Load Files, Applications, SD associations, privileges, Identifiers, life cycle states and memory resource quotas. To be protected from unauthorised modification.

4.2 Description of Threats

4.2.1Confidentiality

Since this Security Target claims demonstrable conformance to the [[JCPP]], the threats T.CONFID-APPLI-DATA, T.CONFID-JCS-CODE and T.CONFID-JCS-DATA described in Section 5.2.1 of [[JCPP]] are applied here as well.

4.2.2Integrity

Since this Security Target claims demonstrable conformance to the [[JCPP]], the threats T.INTEG-APPLI-CODE, T.INTEG-APPLI-CODE.LOAD, T.INTEG-APPLI-DATA, T.INTEG-APPLI-DATA.LOAD, T.INTEG-JCS-CODE and T.INTEG-JCS-DATA described in Section 5.2.2 of [[JCPP]] are applied here as well.

4.2.3Identity Usurpation

Since this Security Target claims demonstrable conformance to the [[JCPP]], the threats T.SID.1 and T.SID.2 described in Section 5.2.3 of [[JCPP]] are applied here as well.

4.2.4Unauthorized Execution

Since this Security Target claims demonstrable conformance to the [[JCPP]], the threats T.EXE-CODE.1, T.EXE-CODE.2 and T.NATIVE described in Section 5.2.4 of [[JCPP]] are applied here as well.

4.2.5Denial of Service

Since this Security Target claims demonstrable conformance to the [[JCPP]], the threat T.RESOURCES described in Section 5.2.5 of [[JCPP]] is applied here as well.

4.2.6Card Management

T.UNAUTHORISED-CARD-MNGT refines T.INSTALL and T.DELETION from [[JCPP]] and adds threats to card management.

T.UNAUTHORISED-CARD-MNGT Unauthorised Card Management

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

- Load of a package file
- Installation of a package file
- Extradition of a package file or an applet
- Personalisation of an applet or an SD
- Deletion of a package file or an applet
- Privileges update of an applet or an SD

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

The TOE has the following additional threats other than those defined in [[JCPP]].

T.COM-EXPLOIT Communication Channel Exploitation

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.

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 re-personalises the application). Directly threatened asset(s): D.APP_I_DATA, D.APP_C_DATA, and D.GP_REGISTRY

4.2.7Service

Since this Security Target claims demonstrable conformance to the [[JCPP]], the threat T.OBJ-DELETION described in Section 5.2.7 of [[JCPP]] is applied here as well.

4.2.8Miscellaneous

Since this Security Target claims demonstrable conformance to the [[JCPP]], the threat T.PHYSICAL described in Section 5.2.8 of [[JCPP]] is applied here as well.

In addition, this ST claims the augmented packages “Sensitive Array” and “Sensitive Result”, the threat T.PHYSICAL covers, from package Sensitive Array, the following sub-threat exploiting specifically the listed assets below:

- The attacker performs a physical manipulation to alter (part of) an application’s integrity-sensitive data. Directly threatened assets: D.APP_I_DATA, D.PIN, and D.APP_KEYs

Moreover, from package Sensitive Result, the threat T.PHYSICAL covers the following sub-threat exploiting specifically the listed assets below:

- The attacker performs a physical manipulation to alter (part of) an application’s integrity-sensitive data. Directly threatened assets: D.APP_I_DATA, D.PIN, and D.APP_KEYs

4.2.9OS Management

The TOE has the following OS management threats not defined in [[JCPP]].

T.UNAUTHORIZED_OS_MNGT Unauthorized OS Management

An attacker exploits the OS management functions to:

- Install a malicious update without detection
- modify/disclose OS Image or OS configuration commands
- Modify TOE ID

- Interrupt OS Patch update process

Directly threatened asset(s): D.OS_IMAGE, D.CONFIG_DATA, D.TOE_ID.

4.2.10 Limited Mode

The TOE has the following additional threats not defined in [[JCPP]].

T.EXCEPTION-COUNTER Exception Counter Manipulation

The limited mode is determined based on D.EXCEPTION_COUNTER. An attacker tries to manipulate the exception counter may cause the limited mode to exit without authentication.

Directly threatened asset(s): D.EXCEPTION_COUNTER.

T.LIMITED-MODE Limited Mode Manipulation

The attacker deploys a series of subtle physical attacks (such as precise fault injection, timing attacks, or voltage manipulation) aimed at the TOE. These attacks are designed to be stealthy enough to evade the TOE's primary attack detection mechanisms.

Directly threatened asset(s): D.APP_CODE, D.APP_I_DATA, D.PIN, and D.APP_KEY.

4.3 Organizational Security Policies

Since this Security Target claims demonstrable conformance to the [[JCPP]], the organizational security policy OSP.VERIFICATION described in Section 5.3 of [[JCPP]] is applied here as well.

4.4 Assumptions

Since this Security Target claims demonstrable conformance to the [[JCPP]], the assumptions A.CAP_FILE and A.VERIFICATION described in Section 5.4 of [[JCPP]] are applied here as well.

Note that the assumption A.DELETION from [[JCPP]] is excluded. The Card Manager of the TOE ensures the security of the applet deletion operation. Therefore the assumption is no longer relevant.

The following assumptions from the ST of security IC, GSE20 [31], are refined in this ST.

A.Process-Sec-IC Protection during Packaging, Finishing and Personalization

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. For a preliminary list of assets to be protected are:

1. the Security IC Embedded Software and its specifications, implementation and related documentation,
2. Pre-personalisation Data and Personalisation Data including specifications of formats and memory areas, test related data,
3. the user data of the Composite TOE and related documentation, and
4. material for software development support

A.Resp-Appl Treatment of user data of the Composite TOE

All user data of the Composite TOE are owned by Security IC Embedded Software. Therefore, it must be assumed that security relevant user data of the Composite TOE (especially cryptographic keys) are treated by the Security IC Embedded Software as defined for its specific application context.

Application Note: The trusted Applet developers shall well protect their user data. During the TOE usage, the terminal or system in interaction with the TOE, shall ensure the protection (integrity and confidentiality) of their own keys and user data by operational means and/or procedures.

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

The Application Provider (AP) must well protect the security of the application together with its security domain keys (D.APP_KEYs).

The AP must change its default security domain keys before performing any operation.

The Verification Authority (VA) must well protect the security of the application verification key and securely verify the applications to be loaded on the card with the verification key.

5 Security Objectives

5.1 Security Objectives for the TOE

The security objectives O.SID, O.FIREWALL, O.GLOBAL_ARRAYS_CONFID, O.GLOBAL_ARRAYS_INTEG, O.ARRAY_VIEWS_CONFID, O.ARRAY_VIEWS_INTEG, O.NATIVE, O.OPERATE, O.REALLOCATION, O.RESOURCES, O.ALARM, O.CIPHER, O.RNG, O.KEY-MNGT, O.PIN-MNGT, O.TRANSACTION and O.OBJ-DELETION defined in section 6.1 of [[JCPP]] are applied here.

The security objectives O.SENSITIVE_ARRAYS_INTEG, O.SENSITIVE_RESULTS_INTEG defined in [[JCPP]] Appendix 2 are applied here as well. O.CIPHER and O.KEY-MNGT do not need to be modified due to the inclusion of package “Key Derivation Functions”.

In addition, the Security Objectives described in the following sections are defined/refined for the TOE.

5.1.1Card Management

The security objective for the environment OE.CARD-MANAGEMENT defined in [[JCPP]] section 6.2 is replaced by O.CARD-MANAGEMENT defined here.

O.CARD-MANAGEMENT	Card Management The card manager as defined in [11] section 3.8 shall control the access to card management functions such as the installation, update, or deletion of applets. It shall also implement the Issuer’s policy on the card. The card manager is an application with specific rights (e.g. ISD), which is responsible for the administration of the SE. This component will in practice be tightly connected with the TOE, which in turn shall very likely rely on the card manager for the effective enforcing of some of its security functions. Typically, the card manager shall be in charge of the life cycle of the whole card, as well as that of the installed applications (applets). The card manager should prevent card content management operations (loading, installation, deletion) from being carried out, for instance, at invalid states of the card or by non-authorised actors. It shall also enforce security policies established by the Issuer.
O.DOMAIN-RIGHTS	Application Security Domain The Issuer shall not get access or change personalised APSD keys, which belong exclusively to the AP. Modification of an SD key set is restricted to the AP who owns the SD.
Application Note:	APs’ Security Domain keyset is used to establish a secure channel between the APs and the platform. The key sets are unknown to the Card Issuer. They must be changed before any operation on the security domain (OE.Resp-Appl).

5.1.2Secure Communication

O.COMM_AUTH Communications Authenticity

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

O.COMM_INTEGRITY Communications Integrity

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

O.COMM_CONFIDENTIALITY Communications Confidentiality

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

5.1.3Security IC

The Security Objectives for the environment OE.SCP.IC, OE.SCP.RECOVERY and OE.SCP.SUPPORT defined in [[JCPP]] Section 6.2 are replaced by the Security Objectives O.SCP.IC, O.SCP.RECOVERY and O.SCP.SUPPORT of the TOE.

O.SCP.IC IC Physical Protection

The SCP of the TOE shall provide security features against physical attacks which addresses the security aspect #.SCP (7).

O.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 of the shall allow the TOE software 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 #.SCP (1).

O.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 #.SCP

5.1.4OS Management

The TOE includes OS Management feature to fulfill the following objectives:

O.AUTH-OS-MNGT Authorized OS Management

The TOE shall ensure that only authorized entity can configure the OS

The following objectives are based on [JIL_SRCL].

O.SECURE-LOAD-ACODE Secure loading of the Additional Code

The TOE shall check an evidence of authenticity and integrity of the loaded Additional Code. The TOE enforces that only the allowed version of the Additional Code can be loaded. The TOE 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.

NOTE: Additional Code stands for D.OS_IMAGE

O.SECURE-AC-ACTIVATION Secure activation of the Additional Code

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 final TOE shall be completed before activation.
If the Atomic Activation is successful, then the resulting product is the updated TOE, otherwise (in case of interruption or incident which prevent the forming of the final TOE such as tearing, integrity violation, error case…), the initial TOE shall remain in its initial state or fail secure.

NOTE: Additional Code stands for D.OS_IMAGE

O.TOE-IDENTIFICATION Secure identification of the TOE by the user

The TOE’s identification data identifies the ROM and OS Patch version field. 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 final TOE’s identification data allows identifications of the ROM and OS Patch version field. The user shall be able to uniquely identify the

final TOE or initial TOE through ROM and OS Patch version field in TOE’s identification data.

NOTE: Additional Code stands for D.OS_IMAGE.

5.1.5Limited Mode

O.EXCEPTION-COUNTER Exception Counter

The TOE shall ensure that in the limited mode, the exception counter can only be reset using specified commands after authentication by the card issuer.

O.LIMITED-MODE Limited Mode

The TOE shall trigger exception mechanisms under unexpected physical behaviors, leading to increments in the exception counter. The TOE shall ensure that only limited set of commands are available when the TOE is put into Limited Mode and reject all other operations.

5.2 Security Objectives for the operational environment

The security objectives for the operation environment OE.CAP_FILE, OE.VERIFICATION and OE.CODE-EVIDENCE defined in [[JCPP]] section 6.2 are applied here as well. OE.CARD-MANAGEMENT, OE.SCP.IC, OE.SCP.RECOVERY and OE.SCP.SUPPORT in [[JCPP]] section 6.2 are replaced by O.CARD-MANAGEMENT, O.SCP.IC, O.SCP.RECOVERY and O.SCP.SUPPORT in this ST. In addition, the ST introduced the following SOEs with application notes as required by the IC platform.

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.

OE.Resp-Appl Treatment of user data of the Composite TOE

All user data of the Composite TOE are owned by Security IC Embedded Software. Therefore, it must be assumed that security relevant user data of the Composite TOE (especially cryptographic keys) are treated by the Security IC Embedded Software as defined for its specific application context.

Application Note: The trusted Applet developers shall well protect their user data. During the TOE usage, the terminal or system in interaction with the TOE, shall ensure the protection (integrity and confidentiality)

of their own keys and user data by operational means and/or procedures.

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

The Application Provider (AP) must well protect the security of the application together with its security domain keys (D.APP_KEYs).

The AP must change its default security domain keys before performing any operation.

The Verification Authority (VA) must well protect the security of the application verification key and securely verify the applications to be loaded on the card with the verification key.

5.3 Security Objectives Rationale

Section 6.3 in the [[JCPP]] provides a rationale how the assumptions, threats, and OSPs are addressed by the objectives that are specified in the [[JCPP]]. The rationales for OE.CARD-MANAGEMENT, OE.SCP.IC, OE.SCP.RECOVERY and OE.SCP.SUPPORT as defined in [[JCPP]] Section 6.3.4 remains valid for O.CARD-MANAGEMENT, O.SCP.IC, O.SCP.RECOVERY and O.SCP.SUPPORT in this ST.

The following table provide additional tracing from the assumptions, threats and OSPs to objectives introduced or modified by this ST.

Security Problem Definition	Security Objective Rationale
T.UNAUTHORISED-CARD-MNGT	O.CARD-MANAGEMENT controls the access to card management functions such as the loading, installation, extradition, or deletion of applets. O.COMM_AUTH prevents unauthorized users from initiating a malicious card management operation. O.COMM_INTEGRITY protects the integrity of the card management data while it is in transit to the card. O.COMM_CONFIDENTIALITY prevents the disclosure of encrypted data transiting to the card. O.DOMAIN-RIGHTS restricts the modification of an AP security domain keyset to the AP who owns it.
T.COM-EXPLOIT	O.COMM_AUTH prevents unauthorized users from initiating a malicious card management operation. O.COMM_INTEGRITY protects the integrity of the card management data while it is in transit to the card. O.COMM_CONFIDENTIALITY prevents the disclosure of encrypted data transiting to the card.

T.LIFE_CYCLE	O.CARD-MANAGEMENT that controls the access to card management functions such as the loading, installation, extradition, or deletion of applets. O.DOMAIN-RIGHTS that restricts 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.
T.UNAUTHORIZED_OS_MNGT	This threat is covered by enforcing authorized users to perform OS management operations and updates as stated in O.AUTH-OS-MNGT, O.SECURE-LOAD-ACODE, O.SECURE-AC-ACTIVATION, O.TOE-IDENTIFICATION.
T.EXCEPTION-COUNTER	This threat is covered by enforcing that only Card Issuer can reset the Exception Counter, as defined in O.EXCEPTION-COUNTER.
T.LIMITED-MODE	This threat is encountered by O.LIMITED-MODE. the physical nature of these attacks still triggers the TOE's exception mechanisms, leading to increments in the exception counter and entering the Limited Mode.
T.PHYSICAL	This threat is covered by entering into Limited Mode when D.EXCEPTION_COUNTER reaches its limit value, as described in O.LIMITED-MODE. Due to the fact that this ST claims the augmented packages "Sensitive Array" and "Sensitive Result", this threat is covered by O.SENSITIVE_ARRAYS_INTEG and O.SENSITIVE_RESULTS_INTEG as defined in [[JCPP]] In addition, this is also covered by the physical protections of the underlying platform as defined in O.SCP.IC.
A.Process-Sec-IC	Since OE.Process-Sec-IC requires the Composite Product Manufacturer to implement those measures assumed in A.Process-Sec-IC, the assumption is covered by this objective.
A.Resp-Appl	Since OE.Resp-Appl requires the Security IC Embedded Software to implement measures as assumed in A.Resp-Appl, the assumption is covered by the objective.

Table 5 Security Objective Rationale

6 Extended Components Definition

Two extended components defined and described in [[JCPP]] are applied here as well for the TOE and one extended component defined and described in the [BSI-PP-0055] is applied here as well for the TOE:

6.1 Definition of FCS_RNG

The family FCS_RNG of the class FCS Cryptographic Support is defined and described in the [[JCPP]].

FCS_RNG Generation of random numbers

Family behavior: This family defines quality requirements for the generation of random numbers which are intended to be use for cryptographic purposes.

Component levelling:

FCS_RNG.1 Generation of random numbers requires that random numbers meet a defined quality metric.

Management: FCS_RNG.1

There are no management activities foreseen.

Audit: FCS_RNG.1

There are no actions defined to be auditable.

FCS_RNG.1 Random number generation

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1 The TSF shall provide a [selection: physical, non-physical true, deterministic, hybrid physical, hybrid deterministic] random number generator [selection: DRG.2, DRG.3, DRG.4, PTG.2, PTG.3, NTG.1] [AIS20] [AIS31] that implements: [assignment: list of security capabilities].

FCS_RNG.1.2 The TSF shall provide random numbers that meet [assignment: a defined quality metric].

6.2 Definition of FPT_EMSEC

The family FPT_EMSEC TOE Emanation of the class FPT Protection of the TSF is defined and described in the [BSI-PP-0055].

Family behavior: This family defines requirements to mitigate intelligible emanations.

Component levelling:

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].

6.3 Cryptographic Key Derivation (FCS_CKM.5)

The family FCS_CKM.5 of the class FCS Cryptographic Support is defined and described in the [[JCPP]].

FCS_CKM Cryptographic Key derivation

Family behavior: This section describes a component of the family Cryptographic key management (FCS_CKM) for key derivation as process by which one or more keys are calculated from either a preshared key or a shared secret and other information. Key derivation is the deterministic repeatable process by which one or more keys are calculated from both a pre-shared key or shared secret, and other information, while key generation required by FCS_CKM.1 uses internal random numbers.

Component levelling:

The component FCS_CKM.5 is on the same level as the other components of the family FCS_CKM.

Management:	FCS_CKM.5 There are no management activities foreseen.
Audit:	FCS_CKM.5 The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST: a) Minimal: Success and failure of the activity. b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys).

FCS_CKM.5 Requires the TOE to provide key derivation.

FCS_CKM.5 Cryptographic key derivation

Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] and FCS_CKM.4 Cryptographic key destruction.

FCS_CKM.5.1	The TSF shall derive cryptographic keys [assignment: key type] from [assignment: input parameters] in accordance with a specified cryptographic key derivation algorithm [assignment: cryptographic key derivation algorithm] and specified cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

7 Security Requirements

This part of the Security Target defines the detailed security requirements that shall be satisfied by the TOE. The statement of TOE security requirements shall define the functional and assurance security requirements that the TOE needs to satisfy in order to meet the security objectives for the TOE. This chapter consists of the sections "Security Functional Requirements", "Security Assurance Requirements" and "Security Requirements Rationale".

The CC allows several operations to be performed on security requirements (on the component level); refinement, selection, assignment, and iteration are defined in paragraph 8.1 of the CC Part1 [CC1]. These operations are used in [[JCPP]] and in this Security Target, respectively.

The **refinement** operation is used to add details to requirements, and, thus, further restricts a requirement. Refinements of security requirements are denoted in such a way that added words are in **bold text** and removed/changed words are crossed out as ~~crossed out text~~.

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments having been made are denoted by showing as *italic text.*

The **selection** operation is used to select one or more options provided by [[JCPP]] or CC in stating a requirement. Selections having been made are denoted as *underlined italic*.

The **iteration** operation is used when a component is repeated with varying operations. It is denoted by showing brackets "/iteration indicator" and the iteration indicator after the slash.

Security functional requirements from the Protection Profile are applied to this Security Target. In compliance with Application Note 12 in the Protection Profile

7.1 Security Functional Requirements

This section states the security functional requirements for the TOE. For readability and for compatibility with previous versions, requirements are arranged into groups. The following groups defined [[JCPP]] Section 7.2 are applied here: Core with Logical Channels (CoreG_LC), Installation (InstG), Applet deletion (ADELG), Object deletion (ODELG) and Secure carrier (CarG).

"ELF Loading SFP" replaces "Package Loading SFP". Covers INSTALL and LOAD commands. Data & Key Loading information flow control SFP is included for loading of SD/Application keys and data through STORE DATA and PUT KEY commands.

All subjects (prefixed with an "S") defined in [[JCPP]] Section 7.2 are applied here: S.APPLET, S.BCV, S.CAD, S.JCRE, S.JCVM, S.LOCAL, S.MEMBER and S.CAP_FILE, except that the S.BCV defined in [[JCPP]] is refined as S.SD described in the following table together with the new subjects introduced in this ST.

S.ADEL and S.INSTALLER are parts of S.OPEN.

Subject	Description
S.SD	A GlobalPlatform SD representing an off-card entity on the card. This entity can be the Issuer, an Application Provider, the Controlling Authority, or the Validation Authority.
S.seRoot	S.SeRoot is the representative of the OS Administrator within the TOE who is responsible for verifying the signature of the additional code before authorizing its loading, installation and activation. seRoot can be selected when the TOE enters into Limited Mode.
S.OPEN	It represents the GP Environment (OPEN) on the card. The main responsibilities of the S.OPEN is to provide an API to applications, command dispatch, Application selection, (optional) logical channel management, Card Content management, memory management, and Life Cycle management.

Table 6 Subjects introduced in this ST

Objects (prefixed with an "O") defined in [[JCPP]] Section 7.2 are applied here: O.APPLET, O.CODE_CAP_FILE and O.JAVAOBJECT

Information (prefixed with an "I") defined in [[JCPP]] Section 7.2 are applied here: I.APDU and I.DATA

Security attributes linked to these subjects, objects and information defined in [[JCPP]] Section 7.2 are applied here: Active Applets, Applet Selection Status, Applet's version number, CAP File AID, Context, Currently Active Context, Dependent package AID, LC Selection Status, LifeTime, Owner, Package ID, Registered Applets, Resident CAP files, Resident Packages, Selected Applet Context, Sharing and Static References.

Operations (prefixed with "OP") defined in [[JCPP]] Section 7.2 are applied here: OP.ARRAY_ACCESS, OP.ARRAY_LENGTH, OP.ARRAY_T_ALOAD, OP.ARRAY_T_ASTORE, OP.ARRAY_AASTORE, OP.CREATE, OP.DELETE_APPLET, OP.DELETE_CAP_FILE, OP.DELETE_CAP_FILE_APPLET, OP.INSTANCE_FIELD, OP.INVK_VIRTUAL, OP.INVK_INTERFACE, OP.JAVA, OP.PUT, OP.THROW and OP.TYPE_ACCESS.

Table 7 lists all the security functional requirements of the TOE. The definition of those that are not modified from the claimed PP [[JCPP]] are not included in the document. Refer to the original PP.

SFR	Description	Modified/Added in ST
CoreG_LC Management Security Functional Requirements		
FDP_ACC.2/FIREWALL	Complete access control	No
FDP_ACF.1/FIREWALL	Security attribute based access control	No
FDP_IFC.1/JCVM	Subset information flow control	No

FDP_IFF.1/JCVM	Simple security attributes	Modified, see section 7.1.1.1
FDP_RIP.1/OBJECTS	Subset residual information protection	No
FMT_MSA.1/JCRE	Management of security attributes	No
FMT_MSA.1/JCVM	Management of security attributes	No
FMT_MSA.2/FIREWALL_JCVM	Secure security attributes	No
FMT_MSA.3/FIREWALL	Static attribute initialization	No
FMT_MSA.3/JCVM	Static attribute initialization	No
FMT_SMF.1	Specification of Management Functions	No
FMT_SMR.1	Security roles	No
FCS_CKM.1	Cryptographic key generation	Modified, see section 7.1.1.1
FCS_CKM.4	Cryptographic key destruction	Modified, see section 7.1.1.1
FCS_CKM.5	Cryptographic key derivation function	Modified, see section 7.1.1.1
FCS_COP.1	Cryptographic operation	Modified, see section 7.1.1.1
FDP_RIP.1/ABORT	Subset residual information protection	No
FDP_RIP.1/APDU	Subset residual information protection	No
FDP_RIP.1/GlobalArray	Subset residual information protection	No
FDP_RIP.1/bArray	Subset residual information protection	No
FDP_RIP.1/KEYS	Subset residual information protection	No
FDP_RIP.1/TRANSIENT	Subset residual information protection	No
FDP_ROL.1/FIREWALL	Basic rollback	No
FAU_ARP.1	Security alarms	Modified, see section 7.1.1.1
FDP_SDI.2/DATA	Stored data integrity monitoring and action	Modified, see section 7.1.1.1
FDP_SDI.2/ARRAY	Stored data integrity monitoring and action	No
FDP_SDI.2/RESULT	Stored data integrity monitoring and action	Modified, see section 7.1.1.6
FPR_UNO.1	Unobservability	Modified, see section 7.1.1.1
FPT_FLS.1	Failure with preservation of secure state	No
FPT_TDC.1	Inter-TSF basic TSF data consistency	Modified, see section 7.1.1.1

FIA_ATD.1/AID	User attribute definition	No
FIA_UID.2/AID	User identification before any action	No
FIA_USB.1/AID	User-subject binding	Modified, see section 7.1.1.1
FMT_MTD.1/JCRE	Management of TSF data	No
FMT_MTD.3/JCRE	Secure TSF data	No
InstG Security Functional Requirements		
FPT_RCV.3/Installer	Automated recovery without undue loss	Modified, see section 7.1.1.2
AdelG Security Functional Requirements		
FDP_ACC.2/ADEL	Complete access control	No
FDP_ACF.1/ADEL	Security attribute based access control	No
FDP_RIP.1/ADEL	Subset residual information protection	No
FMT_MSA.1/ADEL	Management of security attributes	No
FMT_MSA.3/ADEL	Static attribute initialization	No
FMT_SMF.1/ADEL	Specification of Management Functions	No
FMT_SMR.1/ADEL	Security roles	No
FPT_FLS.1/ADEL	Failure with preservation of secure state	No
OdelG Security Functional Requirements		
FDP_RIP.1/ODEL	Subset residual information protection	No
FPT_FLS.1/ODEL	Failure with preservation of secure state	No
ELF Loading Information Flow Control Policy		
FDP_IFC.2/GP-ELF	Complete information flow control	Refined, see section 7.1.1.3
FDP_IFF.1/GP-ELF	Simple security attributes	Refined, see section 7.1.1.3
FDP_ITC.2/GP-ELF	Import of user data with security attributes	Refined, see section 7.1.1.3
Data & Key Loading Information Flow Control Policy		
FDP_IFC.2/GP-KL	Complete information flow control	Added, see section 7.1.1.4
FDP_IFF.1/GP-KL	Simple security attributes	Added, see section 7.1.1.4
FDP_ITC.2/GP-KL	Import of user data with security attributes	Added, see section 7.1.1.4
GP Group		
FCO_NRO.2/GP	Enforced proof of origin	Refined, see section 7.1.1.5

FDP_UIT.1/GP	Data exchange integrity	Refined, see section 7.1.1.5
FMT_SMR.1/GP	Security roles	Refined, see section 7.1.1.5
FPT_FLS.1/GP	Failure with preservation of secure state	Refined, see section 7.1.1.5
FPT_TDC.1/GP	Inter-TSF basic TSF data consistency	Added, see section 7.1.1.5
FIA_UID.1/GP	Timing of identification	Refined, see section 7.1.1.5
FIA_UAU.1/GP	Timing of authentication	Added see section 7.1.1.5
FIA_UAU.4/GP	Single-use authentication mechanisms	Added, see section 7.1.1.5
FMT_MSA.1/GP	Management of security attributes	Refined, see section 7.1.1.5
FMT_MSA.3/GP	Static attribute initialization	Refined, see section 7.1.1.5
FMT_SMF.1/GP	Specification of Management Functions	Refined, see section 7.1.1.5
FTP_ITC.1/GP	Inter-TSF trusted channel	Refined, see section 7.1.1.5
OS Management Security Functional Requirements		
FDP_ACC.2/OSM	Complete information flow control	Added, see section 7.1.2.1
FDP_ACF.1/OSM	Simple security attributes	Added, see section 7.1.2.1
FDP_UIT.1/OSM	Data exchange integrity	Added, see section 7.1.2.1
FMT_MSA.3/OSM	Static attribute initialization	Added, see section 7.1.2.1
FMT_SMF.1/OSM	Specification of Management Functions	Added, see section 7.1.2.1
FTP_ITC.1/OSM	Inter-TSF trusted channel	Added, see section 7.1.2.1
FPT_FLS.1/OSM	Failure with preservation of secure state	Added, see section 7.1.2.1
Smart Card Platform Security Functional Requirements		
FCS_RNG.1/PTG.2	Random Number Generation (PTG.2)	Added, see section 7.1.3
FCS_RNG.1/DRG.3	Random Number Generation (class DRG.3)	Added, see section 7.1.3
FPT_EMSEC.1	TOE Emanation	Added, see section 7.1.3
FPT_PHP.3	Resistance to physical attack	Added, see section 7.1.3
Limited Mode Group		
FDP_ACF.1/LM	Security attribute based access control	Added see section 7.1.2.2
FDP_ACC.2/LM	Complete access control	Added see section 7.1.2.2
FMT_MSA.1/LM	Management of security attribute	Added see section 7.1.2.2
FMT_MSA.3/LM	Static attribute initialisation	Added see section 7.1.2.2

FMT_SMF.1/LM	Specification of Management Functions	Added see section 7.1.2.2
FIA_UID.1/LM	Timing of Identification	Added see section 7.1.2.2
FIA_UAU.1/LM	Timing of authentication	Added see section 7.1.2.2

Table 7 Security Functional Requirements from [[JCPP]]

7.1.1 Security Functional Requirements refined or modified in this Security Target

7.1.1.1CoreG_LC Group

The Core with Logical Channels SFRs from the [[JCPP]] are refined by the following SFRs.

FDP_IFF.1/JCVM	**Simple security attributes**
Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialization
FDP_IFF.1.1/JCVM	The TSF shall enforce the *JCVM information flow control SFP* based on the following types of subject and information security attributes: • *subject: S.JCVM* • *security attribute: Currently Active Context*
FDP_IFF.1.2/JCVM	The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: • *An operation OP.PUT(S1, S.MEMBER, I.DATA) is allowed if and only if the Currently Active Context is "Java Card Runtime Environment".* • *Any other OP.PUT operations are allowed regardless of the Currently Active Context.*
FDP_IFF.1.3/JCVM	The TSF shall enforce *no additional information flow control SFP rules.*
FDP_IFF.1.4/JCVM	The TSF shall explicitly authorise an information flow based on the following rules: *none*
FDP_IFF.1.5/JCVM	The TSF shall explicitly deny an information flow based on the following rules: *none*
Application note:	The storage of temporary Java Card Runtime Environment's objects references is runtime-enforced ([8], §6.2.8.1-3). It should be noticed that this policy essentially applies to the execution of bytecode. Native methods, the Java Card RE itself and possibly some API methods can be granted specific rights or limitations through the FDP_IFF.1.3 /JCVM to FDP_IFF.1.5/JCVM

elements. The way the Java Card virtual machine manages the transfer of values on the stack and local variables (returned values, uncaught exceptions) from and to internal registers is implementation dependent. For instance, a returned reference, depending on the implementation of the stack frame, may transit through an internal register prior to being pushed on the stack of the invoker. The returned bytecode would cause more than one OP.PUT operation under this scheme.

FCS_CKM.1 **Cryptographic key generation**

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/RSA The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *RSA-ND and RSA-CRT* and specified cryptographic key sizes*: any length that is multiple of 64 from 1900 to 4096 bits* (*) that meet the following: *FIPS PUB 186-5 [3].*

Application Note: The keys can be generated and diversified in accordance with [9] specification in classes KeyPair.
(*) The keys sizes from 1900 to 3000 are supported by the product but has legacy until December 2025 with [ACM]; and from 3000 to 4096 is supported by the product and recommended by [ACM].

FCS_CKM.1.1/ECC The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *ECC* and specified cryptographic key sizes *ECC: 256, 384, 512 and 521 bits* (*) that meet the following: *FIPS PUB 186-5 [3].*

Application Note: The keys can be generated and diversified in accordance with [9] specification in classes KeyPair.
(*) The following table shows a mapping between the ECC Families used with its corresponding key sizes. The ECC Families NIST, FR and Brainpool with key sizes from 256 to 521 are recommended by SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms v1.3 [ACM].

Size [bits]	ECC	Elliptic Curve Families
256	secp256r1 Brainpool-p256r1; frp256v1	NIST; Brainpool; ISO/IEC 15946-5
384	secp384r1, Brainpool-p384r1;	NIST; Brainpool;
512	Brainpool-p512r1	Brainpool
521	secp521r1	NIST

Table 8 Mapping between the ECC Families used with its corresponding key sizes.

FCS_CKM.4 **Cryptographic key destruction**

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.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting the keys with random numbers* that meets the following: *none*.

Application Note:

- The keys are reset as specified in [9] Key class, with the method clearKey(). Any access to a cleared key for ciphering or signing shall throw an exception.
- This component shall be instantiated according to the version of the Java Card API applicable to the security target and the implemented algorithms ([9]).

FCS_CKM.5 Cryptographic key derivation

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] and FCS_CKM.4 Cryptographic key destruction.

FCS_CKM.5.1 The TSF shall derive cryptographic keys *Table 9, column no. 2 'Key type'* from *Table 9, column no. 3 'Input parameters'* in accordance with a specified cryptographic key derivation algorithm *Table 9, column no. 4 'Algorithm and usage'* and specified cryptographic key sizes *Table 9, column no. 5 'Key size'* that meet the following: *Table 9, column no. 6 'Standard'*.

Iteration	Key type	Input parameters	Algorithm and usage	Key size	Standard
/KDF_TLS12	Secret key	Seed	ALG_PRF_TLS12	any length within the specification limit [5]	IETF RFC 5246
/KDF_ANSI_x963	Secret key	Counter	ALG_KDF_ANSI_X9_63	any length within the specification limit [29]	ANSI X9.63
/KDF_HKDF	Secret key	Salt	ALG_KDF_HKDF	any length within the specification limit [22]	IETF RFC 5869

Table 9. FCS_CKM.5.1 Key Derivation Functions

FCS_COP.1 Cryptographic Operation

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_COP.1.1/TDES	The TSF shall perform *decryption and encryption* in accordance with a specified cryptographic algorithm: • *ALG_DES_CBC_ISO9797_M1 (*)* • *ALG_DES_CBC_ISO9797_M2(*)* • *ALG_DES_CBC_NOPAD (*)* • *ALG_DES_CBC_PKCS5 (*)* and cryptographic key sizes *168 bits* that meet the following: *Java Card API specification* [9]
Application Note:	(*) The *CBC_MODE* is supported by the product but is legacy until December 2027 for key sizes of 168 bits with SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms v1.3 [ACM].
FCS_COP.1.1/DESMAC	The TSF shall perform *MAC generation and verification* in accordance with a specified cryptographic algorithm *TDES 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_MAC4_NOPAD (*)* • *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 (*)* • *ALG_DES_MAC4_PKCS5 (*)* • *ALG_DES_MAC8_PKCS5 (*)* and cryptographic key sizes *168 bits* that meet the following: *Java Card API specification [9].*
Application Note:	(*) The *MAC_CBC_MODE* is supported by the product but is legacy until December 2027 for key sizes of 168 bits with SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms v1.3 [ACM].
FCS_COP.1.1/AES	The TSF shall perform *decryption and encryption* in accordance with a specified cryptographic algorithm: • *ALG_AES_BLOCK_128_CBC_NOPAD* • *ALG_AES_CBC_ISO9797_M1* • *ALG_AES_CBC_ISO9797_M2* • *ALG_AES_CBC_PKCS5* • *ALG_AES_CFB* • *ALG_AES_CTR* • *ALG_AES_GCM* • *ALG_AES_CCM* and cryptographic key sizes *128, 192 and 256 bits* that meet the following: *for ALC_AES_GCM see FIPS 197 [4], NIST Special Publication 800-38D Recommendation for BlockCipher [1], for ALC_AES_CCM see NIST.SP800-38C, Recommendation for Block Cipher Modes of Operation - The CCM Mode for Authentication and Confidentiality [24], for the rest see Java Card API specification [9].*

FCS_COP.1.1/AES_MAC The TSF shall perform *CMAC generation and verification* in accordance with a specified cryptographic algorithm:
- *ALG_AES_CMAC_128*
- *ALG_AES_MAC_128_NOPAD*

and cryptographic key sizes *128, 192 and 256 bits* that meet the following: *see Java Card API specification [9].*

FCS_COP.1.1/RSA The TSF shall perform *decryption and encryption* in accordance with a specified cryptographic algorithm:
- *ALG_RSA_PKCS1(*)*
- *ALG_RSA_PKCS1_OAEP*

and cryptographic key sizes *any key length that is a multiple of 64 between 1900 and 4096 bits* (**) that meet the following: *Java Card API specification* [9].

Application Note: (*) The *PKCS1v1.5* is legacy until December 2030+ with SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms v1.3 [ACM].
(**) The keys sizes from 1900 to 3000 are supported by the product but has legacy until December 2025 with [ACM]; and from 3000 to 4096 is supported by the product and recommended by [ACM]

FCS_COP.1.1/RSASignature The TSF shall perform *digital signature generation and verification* in accordance with a specified cryptographic algorithm:
- *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*

and cryptographic key sizes *any key length that is a multiple of 64 between 1900 and 4096 bits* (**) that meet the following: *Java Card API specification [9].*

Application Note: (*) The *PKCS1v1.5* is legacy until December 2030+ with SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms v1.3 [ACM].
(**) The keys sizes from 1900 to 3000 are supported by the product but has legacy until December 2025 with [ACM]; and from 3000 to 4096 is supported by the product and recommended by [ACM]

FCS_COP.1.1/ECDSA The TSF shall perform *digital signature generation and verification* in accordance with a specified cryptographic algorithm:
- *ALG_ECDSA_SHA_224 (*)*
- *ALG_ECDSA_SHA_256*
- *ALG_ECDSA_SHA_384*
- *ALG_ECDSA_SHA_512*

and cryptographic key sizes *224, 256, 384, 512 and 521 bits* that meet the following: *Java Card API specification [9].*

Application Note: (*) The *SHA_224* is legacy until December 2025 with SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms v1.3 [ACM].

FCS_COP.1.1/DAP	The TSF shall perform *verification of the DAP signature attached to Executable Load Applications* in accordance with a specified cryptographic algorithm • *ALG_ECDSA_SHA_256* and cryptographic key sizes *512(EC_FP)* that meet the following: *GP Spec [11].*
FAU_ARP.1	**Security alarms**
Hierarchical to:	No other components.
Dependencies:	FAU_SAA.1 Potential violation analysis
FAU_ARP.1.1	The TSF shall take *one of the following actions:* • *throw an exception,* • *lock the card session (after a predefined number of resetted sessions the card shall switch to Limited Mode),* • *reinitialize the Java Card System and its data,* • *response with error code to S.CAD* • *reset session* upon detection of a potential security violation.
Refinement:	**The "potential security violation" stands for one of the following events:** **• CAP file inconsistency,** **• typing error in the operands of a bytecode,** **• applet life cycle inconsistency,** **• card tearing and power failure,** **• abort of a transaction in an unexpected context [9] and ([8], §7.6.2)** **• violation of the Firewall or JCVM SFPs,** **• unavailability of resources,** **• array overflow,** **• checksum mismatch of sensitive arrays** **• functionality of a not present Module is invoked** **• verification fails of Sensitive Result** **• Abnormal environmental condition** **• Card Manager Life Cycle inconsistency** **• General Fault Injection Detection** **• FLASH defects** **• Integrity protected persistent data inconsistency** **• Integrity protected transient data inconsistency** **• Logical Memory Access Violation** **• MMU window access violation** **• Times of try for PIN verification or SCP authentication reach the limit**

Application Note:

- The developer shall provide the exhaustive list of actual potential security violations the TOE reacts to. For instance, other runtime errors related to applet's failure like uncaught exceptions.
- The bytecode verification defines a large set of rules used to detect a "potential security violation". The actual monitoring of these "events" within the TOE only makes sense when the bytecode verification is performed on-card.

- Depending on the context of use and the required security level, there are cases where the card manager and the TOE must work in cooperation to detect and appropriately react in case of potential security violation. This behavior must be described in this component. It shall detail the nature of the feedback information provided to the card manager (like the identity of the offending application) and the conditions under which the feedback will occur (any occurrence of the java.lang.SecurityException exception).
- The "locking of the card session" may not appear in the policy of the card manager. Such measure should only be taken in case of severe violation detection; the same holds for the re-initialization of the Java Card System. Moreover, the locking should occur when "clean" re-initialization seems to be impossible.
- The locking may be implemented at the level of the Java Card System as a denial of service (through some systematic "fatal error" message or return value) that lasts up to the next "RESET" event, without affecting other components of the card (such as the card manager). Finally, because the installation of applets is a sensitive process, security alerts in this case should also be carefully considered herein.

FDP_SDI.2/DATA	**Stored data integrity monitoring and action**
Hierarchical to:	FDP_SDI.1 Stored data integrity monitoring
Dependencies:	No dependencies.
FDP_SDI.2.1/DATA	The TSF shall monitor user data stored in containers controlled by the TSF for *integrity errors* on all objects, based on the following attributes: *the following integrity protected data:* • *D.APP_KEYs: CRC32* • *D.PIN: CRC32* • *D.TOE_ID: protected by IC Flash EDC and checksums [31]*
FDP_SDI.2.2/DATA	Upon detection of a data integrity error, the TSF shall *reset the card session and do the attack velocity check*.
Application Note:	Although no such requirement is mandatory in the Java Card specification, at least an exception shall be raised upon integrity errors detection on cryptographic keys, PIN values and their associated security attributes. Even if all the objects cannot be monitored, cryptographic keys and PIN objects shall be considered with particular attention by ST authors as they play a key role in the overall security. It is also recommended to monitor integrity errors in the code of the native applications and Java Card applets. For integrity sensitive application, their data shall be monitored (D.APP_I_DATA): applications may need to protect information against unexpected modifications, and explicitly control whether a piece of information has been changed between two accesses. For example, maintaining the integrity of an electronic purse's balance is extremely important because this value represents real money. Its modification must be controlled, for illegal ones would denote an important failure of the payment system. A dedicated library could be implemented and made available to developers to achieve better security for specific objects,

following the same pattern that already exists in cryptographic APIs, for instance.

FPR_UNO.1	**Unobservability**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPR_UNO.1.1	The TSF shall ensure that *all users* are unable to observe the operation *all operations* on *D.APP_KEYs and D.PIN* by *another user*.

FPT_TDC.1	**Inter-TSF basic TSF data consistency**
Hierarchical to:	No other components.
Dependencies:	No dependencies
FPT_TDC.1.1	The TSF shall provide the capability to consistently interpret *the CAP files, the bytecode and its data arguments* when shared between the TSF and another trusted IT product.
FPT_TDC.1.2	The TSF shall use • *the rules defined in [10] specification,* • *the API tokens defined in the export files of reference implementation* when interpreting the TSF data from another trusted IT product.
Application Note:	Concerning the interpretation of data between the TOE and the underlying Java Card platform, it is assumed that the TOE is developed consistently with the SCP functions, including memory management, I/O functions and cryptographic functions.

FIA_USB.1/AID	**User-subject binding**
Hierarchical to:	No other components.
Dependencies:	FIA_ATD.1 User attribute definition.
FIA_USB.1.1/AID	The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: *CAP file AID.*
FIA_USB.1.2/AID	The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: *Each uploaded CAP file is associated with a unique CAP file AID.*
FIA_USB.1.3/AID	The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: *The initially assigned CAP file AID is unchangeable.*
Application Note:	The user is the applet and the subject is the S.CAP_FILE. The subject security attribute "Context" shall hold the user security attribute "package AID".

7.1.1.2 InstG Group

The installation SFR from the [[JCPP]] is refined by the following SFRs.

FPT_RCV.3/INSTALLER	**Automated recovery without undue loss**
Hierarchical to:	FPT_RCV.2 Automated recovery.
Dependencies:	AGD_OPE.1 Operational user guidance.
FPT_RCV.3.1/Installer	When automated recovery from *none* is not possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

FPT_RCV.3.2/Installer For *a failure during load/installation of a CAP file/applet and deletion of a CAP file/applet/object*, the TSF shall ensure the return of the TOE to a secure state using automated procedures.

FPT_RCV.3.3/Installer The functions provided by the TSF to recover from failure or service discontinuity shall ensure that the secure initial state is restored without exceeding *0%* for loss of TSF data or objects under the control of the TSF.

FPT_RCV.3.4/Installer The TSF shall provide the capability to determine the objects that were or were not capable of being recovered.

Application Note:

- FPT_RCV.3.1/Installer: This element is not within the scope of the Java Card specification, which only mandates the behavior of the Java Card System in good working order. Further details on the "maintenance mode" shall be provided in specific implementations. The following is an excerpt from [CC2], p296: In this maintenance mode normal operation might be impossible or severely restricted, as otherwise insecure situations might occur. Typically, only authorised users should be allowed access to this mode but the real details of who can access this mode is a function of FMT: Security management. If FMT: Security management does not put any controls on who can access this mode, then it may be acceptable to allow any user to restore the system if the TOE enters such a state. However, in practice, this is probably not desirable as the user restoring the system has an opportunity to configure the TOE in such a way as to violate the SFRs.
- FPT_RCV.3.2/Installer:
  - Should the installer fail during loading/installation of a package/applet, it has to revert to a "consistent and secure state". The Java Card RE has some clean up duties as well; see [8], §11.1.5 for possible scenarios. Precise behavior is left to implementers. This component shall include among the listed failures the deletion of a package/applet. See ([8], §11.3.4) for possible scenarios. Precise behavior is left to implementers.
  - Other events such as the unexpected tearing of the card, power loss, and so on, are partially handled by the underlying hardware platform (see [ICPP]) and, from the TOE's side, by events "that clear transient objects" and transactional features. See FPT_FLS.1.1, FDP_RIP.1/TRANSIENT, FDP_RIP.1/ABORT and FDP_ROL.1/FIREWALL.
- FPT_RCV.3.3/Installer: The quantification is implementation dependent, but some facts can be recalled here. First, the SCP ensures the atomicity of updates for fields and objects, and a power-failure during a transaction or the normal runtime does not create the loss of otherwise-permanent data, in the sense that memory on a smart card is essentially persistent with this respect (Flash). Data stored on the RAM and subject to such failure is intended to have a limited lifetime anyway (runtime data on the stack, transient objects' contents). According to this, the loss of data within the TSF scope should be limited to the same restrictions of the transaction mechanism.

7.1.1.3ELF Loading Information Flow Control Policy

FDP_IFC.2/GP-ELF **Complete information flow control**

Hierarchical to: FDP_IFC.1 Subset information flow control.

Dependencies:	FDP_IFF.1 Simple security attributes
FDP_IFC.2.1/GP-ELF	The TSF shall enforce the *ELF Loading information flow control SFP* on • *Subjects: S.SD, S.CAD, S.OPEN* • *Information: APDU commands INSTALL and LOAD, GP APIs for loading and installing CAP files*
FDP_IFC.2.2/GP-ELF	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.
Application Note:	The subject S.SD can be the ISD, an APSD, or the CASD. GlobalPlatform's card content management APDU commands and API methods are described in [11] Chapter 11 and Appendix A.1, respectively.

FDP_IFF.1/GP-ELF	**Simple security attributes**
Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation
FDP_IFF.1.1/GP-ELF	The TSF shall enforce the *ELF Loading information flow control SFP* based on the following types of subject and information security attributes: • *Subjects: S.SD, S.CAD, S.OPEN* • *Information: I.APDU(Installation Application, Card Management Commands)* • *Security Attributes: Card Life Cycle State information, Privileges data, and the protection security levels of messages.*
FDP_IFF.1.2/GP-ELF	The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: • *S.SD implements one or more Secure Channel Protocols, namely SCP02, SCP03, each with a complete Secure Channel Key Set.* • *S.SD has all of the cryptographic keys required by its privileges.* • *On receipt of INSTALL or LOAD commands, S.OPEN checks that the card Life Cycle State is not CARD_LOCKED or TERMINATED.* • *S.OPEN accepts an Executable Load File only if its integrity and authenticity has been verified.*
FDP_IFF.1.3/GP-ELF	The TSF shall enforce the *none*.
FDP_IFF.1.4/GP-ELF	The TSF shall explicitly authorise an information flow based on the following rules: *none*.
FDP_IFF.1.5/GP-ELF	The TSF shall explicitly deny an information flow based on the following rules: • *S.OPEN fails to verify the integrity and request verification of the authenticity for Executable Load Files.* • *S.OPEN fails to verify the Card Life Cycle state.* • *S.OPEN fails to verify the SD privileges.* • *S.SD fails to verify the security level applied to protect INSTALL or LOAD commands.*

- *S.SD fails to set the security level (integrity and/or confidentiality) to apply to the next incoming command and/or next outgoing response.*
- *S.SD fails to unwrap INSTALL or LOAD commands.*

Application Note: This SFR refines and replaces FDP_IFF.1/CM of [[JCPP]].
APDUs belongs to the policy ELF Loading information flow control SFP are described in the following references:
-For INSTALL, see [11] section 11.5.
-For LOAD, see [11] section 11.6.
The INSTALL and LOAD commands must only be issued within a Secure Channel Session and the levels of security for these commands depend on the security level defined in the EXTERNAL AUTHENTICATE command.
The Minimum-Security Level of INSTALL and LOAD is 'AUTHENTICATED'.
For instance, Security attributes that can be used in FDP_IFF.1.1/GP-ELF are the authorisation status per Card Life Cycle State information, Privileges data, and the protection security levels of messages: Entity authentication, Integrity and Data Origin authentication, Confidentiality.
For more details about the rules to be applied to each role of INSTALL command, refer to [11] sections 9.3 and 3.4.

FDP_ITC.2/GP-ELF **Import of user data with security attributes**

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/GP-ELF The TSF shall enforce the *ELF Loading information flow control SFP* when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.2.2/GP-ELF The TSF shall use the security attributes associated with the imported user data.

FDP_ITC.2.3/GP-ELF 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/GP-ELF 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/GP-ELF The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE:

- *Java Card rules defined in [10] and [8]: ELF loading is allowed only if, for each dependent ELF, its AID attribute is equal to a resident ELF AID attribute, the major (minor) Version attribute associated to the dependent ELF is less than or equal to the major (minor) Version attribute associated to the resident ELF.*

Application Note:	This SFR corresponds to FDP_ITC.2/Installer of [[JCPP]]. Java Card rules are defined in [10] sections 4.4 and 4.5, [8] section 11. The TSF shall use the INSTALL data format and the LOAD data format when interpreting the user data from outside the TOE.

7.1.1.4Data & Key Loading Information Flow Control Policy

FDP_IFC.2/GP-KL	**Complete information flow control**
Hierarchical to:	FDP_IFC.1 Subset information flow control.
Dependencies:	FDP_IFF.1 Simple security attributes
FDP_IFC.2.1/GP-KL	The TSF shall enforce the *Data & Key Loading information flow control SFP* on • *Subjects: S.SD, S.CAD, S.OPEN* • *Information: GP APDU commands STORE DATA and PUT KEY, GP APIs for loading and storing data and keys* and all operations that cause that information to flow to and from subjects covered by the SFP.
FDP_IFC.2.2/GP-KL	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.
Application Note:	GlobalPlatform's card content management APDU commands and API methods are described in [11] Chapter 11 and Appendix A.1, respectively. The subject S.SD can be the ISD, an APSD, or the CASD.

FDP_IFF.1/GP-KL	**Simple security attributes**
Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation
FDP_IFF.1.1/GP-KL	The TSF shall enforce the *Data & Key Loading information flow control SFP* based on the following types of subject and information security attributes: • *Subjects: S.SD, S.CAD, S.OPEN* • *Information: I.APDU(Installation Application, Card Management Commands)* • *Security Attributes: authorisation status per Card Life Cycle State information, Privileges data, and the protection security levels of messages.*
FDP_IFF.1.2/GP-KL	The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: • *S.SD implements one or more Secure Channel Protocols, namely SCP02, SCP03, each with a complete Secure Channel Key Set.* • *S.SD has all of the cryptographic keys required by its privileges.* • *An Application accepts a message only if it comes from the S.SD it belongs to.* • *On receipt of a request to forward STORE DATA or PUT KEY commands to an Application, S.OPEN checks that the card Life Cycle State is not CARD_LOCKED or TERMINATED.*

- *On receipt of a request to forward STORE DATA or PUT KEY commands to an Application, S.OPEN checks that the requesting S.SD has no restrictions for personalisation.*
- *S.SD unwraps STORE DATA or PUT KEY according to the Current Security Level of the current Secure Channel Session and prior to the command being forwarded to the targeted Application or SD.*

FDP_IFF.1.3/GP-KL The TSF shall enforce the *none*.

FDP_IFF.1.4/GP-KL The TSF shall explicitly authorise an information flow based on the following rules: *none*.

FDP_IFF.1.5/GP-KL The TSF shall explicitly deny an information flow based on the following rules:

- *S.OPEN fails to verify the Card Life Cycle, Application, and SD Life Cycle states.*
- *S.OPEN fails to verify the privileges belong to an SD or an Application.*
- *S.SD fails to unwrap STORE DATA or PUT KEY.*
- *S.SD fails to verify the security level applied to protect APDU commands.*
- *S.SD fails to set the security level (integrity and/or confidentiality) to apply to the next incoming command and/or next outgoing response.*

Application Note: APDUs belongs to the policy Data & Key Loading information flow control SFP are described in the following references:
For PUT KEY, see [11] section 11.8.
For STORE DATA, see [11] section 11.11. The PUT KEY and STORE DATA commands must only be issued within a Secure Channel Session and the levels of security for these commands depend on the security level defined in the EXTERNAL AUTHENTICATE command.
The Minimum-Security Level of PUT KEY and STORE DATA is 'AUTHENTICATED'. For instance, Security attributes that can be used in FDP_IFF.1.1/GP-KL are the authorisation status per Card Life Cycle State information, Privileges data, and the protection security levels of messages: Entity authentication, Integrity and Data Origin authentication, Confidentiality. For more details about Key Access Conditions, Data and Key Management, refer to [11] sections 7.5.2 and 7.6.

FDP_ITC.2/GP-KL **Import of user data with security attributes**

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/GP-KL The TSF shall enforce the *Data & Key Loading information flow control SFP* when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.2.2/GP-KL The TSF shall use the security attributes associated with the imported user data.

FDP_ITC.2.3/GP-KL	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/GP-KL	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/GP-KL	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: *The algorithms and key sizes of the imported keys shall be supported by the Card.*
Application Note:	The algorithms and key sizes of the imported keys shall be supported by the Card as specified in [11] Appendices B and C. PUT KEY and STORE DATA are described in [11] sections 11.8 and 11.11.

7.1.1.5GP Group

The card management SFRs from the [[JCPP]] are refined by the following SFRs.

FPT_FLS.1/GP	**Failure with preservation of secure state**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_FLS.1.1/GP	The TSF shall preserve a secure state when the following types of failures occur: • *S.OPEN fails to load/install an Executable Load File / Application instance.* • *S.SD fails to load SD/Application data and keys.* • *S.OPEN fails to verify/change the Card Life Cycle, Application, and SD Life Cycle states.* • *S.OPEN fails to verify the privileges belong to an SD or an Application.* • *S.SD fails to verify the security level applied to protect APDU commands.*
Application Note:	This SFR extends FPT_FLS.1/Installer of [[JCPP]] to include the failures that may occur during the loading of SD/Application keys and data. Refer to [8] section 11.1.5 and [11] sections 11.5, 11.6, 11.8, 11.11 for additional details.

FCO_NRO.2/GP	**Enforced proof of origin**
Hierarchical to:	FCO_NRO.1 Selective proof of origin.
Dependencies:	FIA_UID.1 Timing of identification.
FCO_NRO.2.1/GP	The TSF shall ~~enforce the generation of evidence of origin for transmitted application packages~~ **be able to generate an evidence of origin** at all times **for 'Executable Load Files, SD/Application data and keys' received from the off-card entity (originator of transmitted data) that communicates with the card.**

FCO_NRO.2.2/GP	The TSF shall be able to relate the ~~[assignment: list of attributes] of the originator of the information, and the [assignment: list of information fields] of the information to which the evidence applies.~~ **'Executable Load Files, SD/Application data and keys' to the card with associated security attributes (the identity of the originator, the destination) such that the evidence of origin can be verified.**
FCO_NRO.2.3/GP	The TSF shall provide a capability to verify the evidence of origin of information to the *off-card entity (recipient of the evidence of origin) who requested that verification* given *no limitation*.
Application Note:	This SFR extends FCO_NRO.2/CM of [[JCPP]] to cover the SD/Application data and keys transmitted and loaded to the card via STORE DATA and PUT KEY commands. 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.

FPT_TDC.1/GP	**Inter-TSF basic TSF data consistency**
Hierarchical to:	No other components.
Dependencies:	No dependencies
FPT_TDC.1.1/GP	The TSF shall provide the capability to consistently interpret *ELFs, SD/Application data and keys, data used to implement a Secure Channel* when shared between the TSF and another trusted IT product.
FPT_TDC.1.2/GP	The TSF shall use *the list of interpretation rules to be applied by the TSF when processing the INSTALL, LOAD, PUT KEY, and STORE DATA commands sent to the card* when interpreting the TSF data from another trusted IT product.
Application Note:	The list of interpretation rules to be applied by the TSF when processing the INSTALL, LOAD, PUT KEY and STORE DATA commands sent to the card are defined in [11] sections 11.5, 11.6, 11.8, and 11.11.

FDP_UIT.1/GP	**Data exchange integrity**
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/GP	The TSF shall enforce the *ELF Loading information flow control SFP and Data & Key Loading information flow control SFP* to *receive* user data in a manner protected from *modification, deletion, insertion and replay* errors.
FDP_UIT.1.2/GP	The TSF shall be able to determine on receipt of user data, whether *modification, deletion, insertion, replay* has occurred.
Application Note:	This SFR extends FDP_UIT.1/CM of [[JCPP]] to cover the integrity protection of SD/Application data and keys. This SFR applies where APDU command and response integrity protection is required. For instance: INSTALL and LOAD commands.

FIA_UID.1/GP	**Timing of Identification**
Hierarchical to:	No other components.
Dependencies:	No dependencies.

FIA_UID.1.1/GP	The TSF shall allow • *application selection* • *secure channel initialization* • *requesting TOE identification data* on behalf of the user to be performed before the user is identified.
FIA_UID.1.2/GP	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.
Application Note:	User are the roles defined in the component FMT_SMR.1/GP.

FIA_UAU.1/GP	**Timing of authentication**
Hierarchical to:	No other components.
Dependencies:	FIA_UID.1 Timing of identification
FIA_UAU.1.1/GP	The TSF shall allow *the TSF mediated actions listed in FIA_UID.1/GP* on behalf of the user to be performed before the user is authenticated.
FIA_UAU.1.2/GP	The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.4/GP	**Single-use authentication mechanisms**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_UAU.4.1/GP	The TSF shall prevent reuse of authentication data related to *the authentication mechanism used to create a secure communication channel*.

FMT_MSA.1/GP	**Management of security attributes**
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/GP	The TSF shall enforce the *ELF Loading information flow control SFP and Data & Key Loading information flow control SFP* to restrict the ability to *Operations in the table below* the security attributes *Security Attributes in the table below* to *Authorized Roles in the table below.*

Operations	Security Attributes: Card Life Cycle State	Authorized Roles
DELETE Executable Load File	OP_READY, INITIALIZED, SECURED	ISD, AM SD, DM SD
DELETE Executable Load File and Application(s)	OP_READY, INITIALIZED, SECURED	ISD, AM SD, DM SD
DELETE Application	OP_READY, INITIALIZED, SECURED	ISD, AM SD, DM SD
DELETE Key	OP_READY, INITIALIZED, SECURED	ISD, AM SD, DM SD, SD
INSTALL	OP_READY, INITIALIZED, SECURED	ISD, AM SD, DM SD
INSTALL [for personalization]	OP_READY, INITIALIZED, SECURED	ISD, AM SD, DM SD, SD
LOAD	OP_READY, INITIALIZED, SECURED	ISD, AM SD, DM SD
PUT KEY	OP_READY, INITIALIZED, SECURED	ISD, AM SD, DM SD, SD
SELECT	OP_READY, INITIALIZED, SECURED, CARD_LOCKED (with privilege)	ISD, AM SD, DM SD, SD (with privilege)

SET STATUS	OP_READY, INITIALIZED, SECURED, CARD_LOCKED	ISD, AM SD, DM SD, SD
STORE DATA	OP_READY, INITIALIZED, SECURED	ISD, AM SD, DM SD, SD
GET DATA	OP_READY, INITIALIZED, SECURED, CARD_LOCKED, TERMINATED	ISD, AM SD, DM SD, SD
GET STATUS	OP_READY, INITIALIZED, SECURED, CARD_LOCKED	ISD, AM SD, DM SD, SD

Operations	Card Life Cycle State	Minimum Security Level	Authorized Identified Roles with Privileges
INITIALIZE UPDATE	OP_READY, INITIALIZED, SECURED, CARD_LOCKED	None	ISD, AM SD, DM SD, SD
EXTERNAL AUTHENTICATE	OP_READY, INITIALIZED, SECURED, CARD_LOCKED	C-MAC	ISD, AM SD, DM SD, SD

Legend:
ISD: Issuer Security Domain
AM SD: Security Domain with Authorised Management privilege
DM SD: Security Domain with Delegated Management privilege
SD: Other Security Domain

Application Note: This SFR refines FMT_MSA.1/CM of [[JCPP]]. It is extended to cover Data and Key loading Policy. The authorised identified roles could are off-card entities as defined in FMT_SMR.1/GP

FMT_MSA.3/GP **Static attribute initialisation**

Hierarchical to: No other components.

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

FMT_MSA.3.1/GP The TSF shall enforce the *ELF Loading information flow control SFP and Data & Key Loading information flow control SFP* to provide *restrictive* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/GP The TSF shall allow the *Card Issuer and Application Provider* to specify alternative initial values to override the default values when an object or information is created.

Application Note: This SFR refines FMT_MSA.3/CM of [[JCPP]]. It is extended to cover Data and Key loading Policy. The authorised identified roles could be off-card or on-card entities as defined in FMT_SMR.1/GP.

FMT_SMF.1/GP **Specification of Management Functions**

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1/GP The TSF shall be capable of performing the following management functions **specified in [11]**:

- *Card and Application Security Management as defined in [11] Life Cycle, Privileges, Application/SD Locking and Unlocking, Card Locking and Unlocking, Card Termination, Application Status interrogation, Card Status Interrogation, command dispatch, Operational Velocity Checking, and Tracing and Event Logging.*

- *Management functions (Secure Channel Initiation/Operation/Termination) related to SCPs as defined in [11].*

Application Note: This SFR corresponds to FMT_SMF.1/CM of [[JCPP]], applied to card content management operations.
Management functions related to SCPs are defined in [11] Chapter 10.

FMT_SMR.1/GP **Security roles**

Hierarchical to: No other components.
Dependencies: FIA_UID.1 Timing of identification
FMT_SMR.1.1/GP The TSF shall maintain the roles

- *On-card: S.OPEN, S.SD (e.g. ISD, APSD), seRoot.*
- *Off-card: Issuer, Users (e.g. VA, AP) owning SDs and OS Administrators.*

FMT_SMR.1.2/GP The TSF shall be able to associate users with roles.
Application Note: This SFR corresponds to FMT_SMR.1/Installer and FMT_SMR.1/CM of [[JCPP]], applied to roles involved in card content management operations (this is why it has been renamed). Moreover, the seRoot role (on-card) represents the functionality within the TOE that can be exercised by the OS Administrator role (off-card). Accordingly, both are, for all practical purposes, the same user role.

FTP_ITC.1/GP **Inter-TSF trusted channel**

Hierarchical to: No other components.
Dependencies: No dependencies
FTP_ITC.1.1/GP 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/GP The TSF shall permit *another trusted IT product* to initiate communication via the trusted channel.
FTP_ITC.1.3/GP The TSF shall initiate communication via the trusted channel for

- *APDU commands sent to the card within a Secure Channel Session.*
- *When loading/installing a new ELF on the card.*
- *When transmitting and loading sensitive data to the card using STORE DATA or PUT KEY commands.*
- *When deleting ELFs, Applications, or Keys.*

Application Note: This SFR corresponds to FTP_ITC.1/CM of [[JCPP]], applied where APDU command and response integrity and/or confidentiality protection through a Secure Channel are required

7.1.1.6Package Sensitive Results

The TOE implements the optional package “Sensitive Results” from [[JCPP]] Appendix 2.

FDP_SDI.2/RESULT	**Integrity_Sensitive_Result**
Hierarchical to:	FDP_SDI.1 Stored data integrity monitoring
Dependencies:	No dependencies.
FDP_SDI.2.1/RESULT	The TSF shall monitor user data stored in containers controlled by the TSF for *integrity errors* on all objects, based on the following attributes: *sensitive API result stored in the javacardx.security.SensitiveResult class*.
FDP_SDI.2.2/RESULT	Upon detection of a data integrity error, the TSF shall *throw an exception*.

7.1.2 Security Functional Requirements introduced in this ST

7.1.2.1OS Management Group

The TOE implements the following OS management SFRs.

FDP_ACC.2/OSM	**Complete access control**
Hierarchical to:	FDP_ACC.1 Subset access control
Dependencies:	FDP_ACF.1 Security attribute-based access control
FDP_ACC.2.1/OSM	The TSF shall enforce the *OS Management access control SFP* on *the following list of subjects, objects and operations:* *Subjects: S.seRoot* *Objects: D.OS_IMAGE, D.CONFIG_DATA.* *Operations: update OS Patch Image and configure OS functionality and commands.* and all operations among subjects and objects covered by the SFP.
FDP_ACC.2.2/OSM	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.

FDP_ACF.1/OSM	**Security attribute based access control**
Hierarchical to:	No other components
Dependencies:	FDP_ACC.1 Subset access control, FMT_MSA.3 Static attribute

FDP_ACF.1.1/OSM	The TSF shall enforce the *OS Management access control SFP* to objects based on the following:
	• *Security attributes:*
	  ○ *D.OS_IMAGE's Identification data*
	  ○ *D.OS_IMAGE's signature*

FDP_ACF.1.2/OSM	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: • *The operations "update OS Patch Image" and "configure OS functionality and commands" are only allowed if the identified and authenticated user role is S.seRoot* • *The operation "update OS Patch Image" is allowed if the verification of D.OS_IMAGE's signature result is successful.*

- *The operation "update OS Patch Image" is allowed if the comparison between the Identification data of both the TOE and the D.OS_IMAGE's demonstrates that the operation can be performed.*

FDP_ACF.1.3/OSM The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *none*.

FDP_ACF.1.4/OSM The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *if the identified user role is S.seRoot all other operations besides "update OS Patch Image" and "configure OS functionality and commands" are denied*.

Application Note: Identification data verification is necessary to ensure that the D.OS_IMAGE code is actually targeting the TOE and that its version is compatible with the TOE version.

FDP_UIT.1/OSM **Data exchange integrity**

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/OSM The TSF shall enforce the *OS Management access control SFP* to *receive* user data in a manner protected from *modification, deletion, insertion and replay* errors.

FDP_UIT.1.2/OSM The TSF shall be able to determine on receipt of user data, whether *modification, deletion, insertion, replay* **of the OS image or configuration commands** has occurred.

FMT_MSA.3/OSM **Static attribute initialisation**

Hierarchical to: No other components.

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

FMT_MSA.3.1/OSM The TSF shall enforce the *OS Management access control SFP* to provide *restrictive* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/OSM The TSF shall allow the *S.seRoot* to specify alternative initial values to override the default values when an object or information is created.

Application Note: The D.OS_IMAGE signature verification status must be set to "Fail" by default, therefore preventing any updates from being installed until the D.OS_IMAGE signature and its identification data is actually successfully verified by the TOE

FMT_SMF.1/OSM **Specification of Management Functions**

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1/OSM The TSF shall be capable of performing the following management functions:

- *update OS Patch Image*
- *configure OS functionality and commands*

FTP_ITC.1/OSM **Inter-TSF trusted channel**

Hierarchical to: No other components.

Dependencies: No dependencies

FTP_ITC.1.1/OSM	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/OSM	The TSF shall permit **the CAD placed in the OS Administrator secured environment** to initiate communication via the trusted channel.
FTP_ITC.1.3/OSM	The TSF shall initiate communication via the trusted channel for - *update OS Patch Image* - *configure OS functionality and commands*

FPT_FLS.1/OSM	**Failure with preservation of secure state**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_FLS.1.1/OSM	The TSF shall preserve a secure state when the following types of failures occur: *interruption or incident which prevents the forming of the updated TOE.*

7.1.2.2Limited Mode Group

The SFRs for Limited Mode are provided here.

FDP_ACC.2/LM	**Complete access control**
Hierarchical to:	FDP_ACC.1 Subset access control
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.2.1/LM	The TSF shall enforce the *Limited Mode access control SFP* on: • *subject: S.SD* • *object: D.EXCEPTION_COUNTER and all objects* and all operations among subjects and objects covered by the SFP.
FDP_ACC.2.2/LM	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.

FDP_ACF.1/LM	**Security attribute based access control**
Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1/LM	The TSF shall enforce the *Limited Mode access control SFP* to objects based on the following: • *subject: S.SD* • *object: D.EXCEPTION_COUNTER and all objects.* • *Attribute: D.EXCEPTION COUNTER and ISD*
FDP_ACF.1.2/LM	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: • *The D.EXCEPTION_COUNTER can be reset by S.SD with security attribute ISD after authentication.*
FDP_ACF.1.3/LM	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *none*

FDP_ACF.1.4/LM	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *Deny all operations over all objects except those covered in FMT_SMF.1/LM if the D.EXCEPTION_COUNTER has reached the limit.*

FMT_MSA.1/LM Management of security attribute

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/LM	The TSF shall enforce the *Limited Mode access control policy* to restrict the ability to *reset* the security attributes: • *D.EXCEPTION_COUNTER,* to • *S.SD with security attribute ISD,*

FMT_MSA.3/LM	**Static attribute initialisation**
Hierarchical to:	No other components.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.3.1/LM	The TSF shall enforce the *Limited Mode access control policy* to provide *restrictive* default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/LM	The TSF shall allow the *nobody* to specify alternative initial values to override the default values when an object or information is created.

FMT_SMF.1/LM	**Specification of Management Functions**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FMT_SMF.1.1/LM	The TSF shall be capable of performing the following management functions: • *reset D.EXCEPTION_COUNTER,* • *select ISD* • *get TOE version* • *select seRoot*

FIA_UID.1/LM Timing of Identification

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_UID.1.1/LM	The TSF shall allow following operations on behalf of the user to be performed before the user is identified. • *select ISD* • *get TOE version* • *select seRoot*
FIA_UID.1.2/LM	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.1/LM	**Timing of authentication**
Hierarchical to:	No other components.
Dependencies:	FIA_UID.1 Timing of identification

FIA_UAU.1.1/LM	The TSF shall allow *select ISD, get TOE version, select seRoot* on behalf of the user to be performed before the user is authenticated.
FIA_UAU.1.2/LM	The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

7.1.3Further Security Functional Requirements from the Smart Card Platform

The TOE has the following functionality provided by the underlying hardware platform [31].

FPT_PHP.3	**Resistance to physical attack**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_PHP.3.1	The TSF shall resist *physical manipulation and physical probing* to the *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:	If a physical attack is detected, an alarm is triggered and the chip will reset or generate an interrupt.

FCS_RNG.1/PTG.2	**Random Number Generation (PTG.2)**
Hierarchical to:	No other components
Dependencies:	No dependencies
FCS_RNG.1.1/PTG.2	The TSF shall provide a physical random number generator that implements: *(PTG.2.1) A total failure test detects a total failure of entropy source immediately when the RNG has started. When a total failure is detected, no random numbers will be output.* *(PTG.2.2) If a total failure of the entropy source occurs while the RNG is being operated, the RNG prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source.* *(PTG.2.3) The online test shall detect non-tolerable statistical defects of the raw random number sequence (i) immediately when the RNG has started, and (ii) while the RNG is being operated. The TSF must not output any random numbers before the power-up online test has finished successfully or when a defect has been detected.* *(PTG.2.4) The online test procedure shall be effective to detect non-tolerable weaknesses of the random numbers soon.*

	(PTG.2.5) The online test procedure checks the quality of the raw random number sequence. It is triggered externally. The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time.
FCS_RNG.1.2/PTG.2	The TSF shall provide **numbers of 32 bits** that meet: *(PTG.2.6) Test procedure A does not distinguish the internal random numbers from output sequences of an ideal RNG.* *(PTG.2.7) The average Shannon entropy per internal random bit exceeds 0.997.*
FCS_RNG.1/DRG.3	**Random Number Generation (Class DRG.3)**
Hierarchical to:	No other components
Dependencies:	No dependencies
FCS_RNG.1.1/DRG.3	The TSF shall provide a deterministic random number generator that implements: *(DRG.3.1) If initialized with a random seed using a PTRNG of class PTG.2 as random source, the internal state of the RNG shall have at least 112 bits entropy.* *Note: The seed is provided by a certified PTG.2 physical TRNG with guaranteed 7.976 bit of entropy per byte.* *(DRG.3.2) The RNG provides forward secrecy.* *(DRG.3.3) The RNG provides backward secrecy even if the current internal state is known.*
FCS_RNG.1.2/DRG.3	The TSF shall provide random numbers that meet: *(DRG.3.4) The RNG, initialized with a random seed from a PTRNG of class PTG.2, generates output for which 2^{48} 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.*
FPT_EMSEC.1	**TOE Emanation**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_EMSEC.1.1	The TOE shall not emit *variations in power consumption or timing during TOE execution* in excess of *non-meaningful information* enabling access to *TSF data: D.CRYPTO* and *User data: D.PIN, D.APP_KEYs*.
FPT_EMSEC.1.2	The TOE shall ensure *the unauthorized users* are unable to use the following interface *contact PINs or chip surfaces* to gain access to *TSF data D.CRYPTO* and *User data D.PIN, D.APP_KEYS*.

7.2 Security Assurance Requirements

The evaluation assurance level is EAL5 augmented with ALC_DVS.2 and AVA_VAN.5. In the following 9, the security assurance requirements are given.

Aspect	Acronym	Description

Development	ADV_ARC.1	Security Architecture design
	ADV_FSP.5	Functional specification
	ADV_IMP.1	Implementation representation
	ADV_INT.2	TSF internals
	ADV_TDS.4	TOE design
Guidance Documents	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
Life-Cycle Support	ALC_CMC.4	CM capabilities
	ALC_CMS.5	CM scope
	ALC_DEL.1	Delivery procedures
	ALC_DVS.2	Development security
	ALC_LCD.1	Life-cycle definition
	ALC_TAT.2	Tools and techniques
Security Target Evaluation	ASE_CCL.1	Conformance claims
	ASE_ECD.1	Extended components definition
	ASE_INT.1	ST introduction
	ASE_OBJ.2	Security objectives
	ASE_REQ.2	Derived security requirements
	ASE_SPD.1	Security problem definition
	ASE_TSS.1	TOE summary specification
Tests	ATE_COV.2	Analysis of coverage

	ATE_DPT.3	Depth
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability Assessment	AVA_VAN.5	Advanced methodical vulnerability testing

Table 10: Assurance components

7.3 Security Requirements Rationale

7.3.1 Rationale for Security Functional Requirements

The SFR rationales for the SOs and SFRs provided in [[JCPP]] Section 7.4.1 and 7.4.2 are applicable for this ST as well.

The rationales for the SOs and SFRs not mentioned in [[JCPP]] are provided below which shows how the security functional requirements are combined to meet the security objectives.

Objective	TOE Security Functional Requirements
O.CARD-MANAGEMENT	FDP_UIT.1/GP ensures the integrity of card management operations. FDP_ITC.2/GP-ELF enforces the ELF loading information flow policy when importing ELFs. FDP_ITC.2/GP-KL enforces the Data & Key information flow policy when importing keys and data. FPT_FLS.1/GP requires the card to preserve a secure state when failures occur during loading/installing/deleting an Executable File / application instance. FDP_IFC.2/GP-ELF, FDP_IFF.1/GP-ELF, FDP_IFC.2/GP-KL, FDP_IFF.1/GP-KL enforce the information flow control policy for managing, authenticating and protecting the Card management commands and responses between off-card and on-card entities.

	FIA_UID.1/GP, FIA_UAU.1/GP and FIA_UAU.4/GP ensure appropriate identification and authentication mechanisms. In addition, they specify the actions that can be performed before authenticating the origin of the APDU commands that the card receives. FCO_NRO.2/GP enforces the evidence of the origin during the loading of Executable Load Files, SD/Application data and keys. FPR_UNO.1 enforces the unobservability of the imported keys and the encryption, decryption, signature generation and verification cryptographic mechanisms on SD/Application keys. FPT_TDC.1/GP specifies requirements for preventing any possible misinterpretation of the Security Domain keys used to implement a Secure Channel when they are loaded from the off-card entity. FTP_ITC.1/GP requires a trusted channel for authenticating the card management commands and for securely protecting (authenticity, integrity, and/or confidentiality) the loading of ELF/data. FMT_MSA.1/GP and FMT_MSA.3/GP specify security attributes enabling to: • ensure the authenticity, integrity, and/or confidentiality of card management commands; • enforce the TOE Life cycle management and transitions. FMT_SMF.1/GP enforces the card management operations (Loading, Installation, etc.), the privileges and transitions. It specifies the actions protecting the card management commands.

	FMT_SMR.1/GP maintains the roles S.OPEN, Card Issuer, Application provider. In addition, it maintains the roles Application Provider/Controlling Authority and specifies the authorised identified roles enabling to send and authenticate card management commands for which the integrity, authenticity, and/or confidentiality have to be ensured. FPT_RCV.3/Installer ensures safe recovery from failure.
O.DOMAIN-RIGHTS	FDP_IFC.2/GP-ELF, FDP_IFF.1/GP-ELF, FDP_IFC.2/GP-KL, FDP_IFF.1/GP-KL enforce the ELF, data, and keys loading information flow control policy for managing, authenticating, and protecting the Card management commands and responses between off-card and on-card entities. FIA_UID.1/GP, FIA_UAU.1/GP, and FIA_UAU.4/GP ensure appropriate identification and authentication mechanisms. In addition, they specify the actions that can be performed before authenticating the origin of the APDU commands that the card receives. FTP_ITC.1/GP requires a trusted channel for authenticating the card management commands and for securely protecting (authenticity, integrity, and/or confidentiality) the loading of ELF/data. FCO_NRO.2/GP enforces the evidence of the origin during the loading of Executable Load Files, SD/Application data, and keys. FMT_MSA.1/GP and FMT_MSA.3/GP specify security attributes enabling to:

	• ensure the authenticity, integrity, and/or confidentiality of card management commands; • enforce the TOE Life cycle management and transitions. FMT_SMF.1/GP enforces the card management operations (Loading, Installation, etc.), the privileges, and transitions. It specifies the actions protecting the card management commands. FMT_SMR.1/GP maintains the roles S.OPEN, Card Issuer, Application provider. In addition, it maintains the roles Application Provider/Controlling Authority and specifies the authorised identified roles enabling to send and authenticate card management commands for which the integrity, authenticity, and/or confidentiality have to be ensured.
O.SCP.IC	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, FPT_EMSEC.1 contributes to the coverage of the objective by ensuring leakage resistant implementations of the unobservable operations. FPT_PHP.3 contributes to the coverage of the objective by preventing bypassing, deactivation or changing of other security features.
O.SCP.RECOVERY	FPT_FLS.1 contributes to the coverage of the objective by preserving a secure state after failure.
O.SCP.SUPPORT	FCS_RNG.1/PTG.2, FCS_RNG.1/DRG.3, FCS_COP.1, FCS_CKM.1, FCS_CKM.4 and FCS_CKM.5 contribute to meet the objective.
O.RNG	FCS_RNG.1/PTG.2 and FCS_RNG.1/DRG.3 contribute to the objective by providing true and pseudo random number generators.
O.AUTH-OS-MNGT	FDP_ACC.2/OSM, FDP_ACF.1/OSM, FDP_UIT.1/OSM, FIA_UID.1/GP, FMT_MSA.3/OSM, FMT_SMF.1/OSM, FMT_SMR.1/GP, FTP_ITC.1/OSM, FPT_FLS.1/OSM

	Contributes to meet this security objective by enforcing authorized OS Management.
O.SECURE-LOAD-CODE	This security objective specifies that the TOE shall check the authenticity and integrity in the additional code to be loaded. This is covered by FDP_ACC.2/OSM, FDP_ACF.1/OSM, FDP_UIT.1/OSM, FIA_UID.1/GP, FMT_MSA.3/OSM, FMT_SMF.1/OSM, FMT_SMR.1/GP, FTP_ITC.1/OSM, FPT_FLS.1/OSM that define the access control policies for the code update and OS management.
O.SECURE-AC-ACTIVATION	This security objective specifies that the activation of the additional code and update of the Identification Data shall be performed at the same time in an atomic way. This is covered by FDP_ACC.2/OSM, FDP_ACF.1/OSM, FDP_UIT.1/OSM, FIA_UID.1/GP, FMT_MSA.3/OSM, FMT_SMF.1/OSM, FMT_SMR.1/GP, FTP_ITC.1/OSM, FPT_FLS.1/OSM that define the access control policies.
O.TOE-IDENTIFICATION	This security objective specifies that the TOE shall decrypt the additional code prior installation. This is covered by FDP_ACC.2/OSM, FDP_ACF.1/OSM, FDP_UIT.1/OSM, FIA_UID.1/GP, FMT_MSA.3/OSM, FMT_SMF.1/OSM, FMT_SMR.1/GP, FTP_ITC.1/OSM, FPT_FLS.1/OSM
O.EXCEPTION-COUNTER	FMT_SMR.1/GP Contributes to cover the objective by defining the security role ISD. FMT_MSA.3/LM Contributes to cover the objective by restricting the initial value of the Exception Counter and allowing nobody to change the initial value. FMT_MSA.1/LM Contributes to cover the objective by only allowing the ISD to modify the Exception Counter. FIA_UAU.1/LM Contributes to cover the objective by requiring authentication before resetting the Exception Counter. FIA_UID.1/LM Contributes to cover the objective by requiring identification before resetting the Exception Counter.
O.LIMITED-MODE	FMT_SMR.1/GP Contributes to cover the objective by defining the security role ISD. FDP_ACC.2/LM Contributes to the coverage of the objective by defining the subject of the Limited Mode access control SFP.

	FDP_ACF.1/LM Contributes to cover the objective by controlling access to objects for all operations. FMT_SMF.1/LM Contributes to cover the objective by defining the management functions of the Limited mode. FIA_UAU.1/LM Contributes to cover the objective by requiring authentication before resetting the Exception Counter. FIA_UID.1/LM Contributes to cover the objective by requiring identification before resetting the Exception Counter.

Table 11: Rational for Additional Security Functional Requirements in the ST

7.3.2 Dependencies of Security Functional Requirements

The analysis of the dependency of the SFRs, including the refined SFRs identified in Section 7.1.1 of this ST, in [[JCPP]] Section 7.4.3.1 is valid for this ST as well.

The dependencies of the SFRs introduced in Section 7.1.1, 7.1.2 and 7.1.3, not analyzed in [[JCPP]] Section 7.4.3.1, are further analyzed below.

Security Functional Requirement	Dependencies	Fulfilled by security requirements
ELF Loading Information Flow Control		
FDP_ITC.2/GP-ELF	(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_IFC.2/GP-ELF, FTP_ITC.1/GP, FPT_TDC.1/GP
FDP_IFC.2/GP-ELF	FDP_IFF.1 Simple security attributes	FDP_IFF.1/GP-ELF
FDP_IFF.1/GP-ELF	FDP_IFC.1 Subset information flow control, FMT_MSA.3 Static attribute initialization	FDP_IFC.2/GP-ELF, FMT_MSA.3/GP
Data & Key Loading Information Flow Control		
FDP_ITC.2/GP-KL	(FDP_ACC.1 Subset access control, or FDP_IFC.1	FDP_IFC.2/GP-KL, FTP_ITC.1/GP, FPT_TDC.1/GP

	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_IFC.2/GP-KL	FDP_IFF.1 Simple security attributes	FDP_IFF.1/GP-KL
FDP_IFF.1/GP-KL	FDP_IFC.1 Subset information flow control, FMT_MSA.3 Static attribute initialization	FDP_IFC.2/GP-KL, FMT_MSA.3/GP
GP Group		
FPT_TDC.1/GP	No Dependencies	No Dependencies
FIA_UAU.1/GP	FIA_UID.1 Timing of identification	FIA_UID.1/GP
FIA_UAU.4/GP	No Dependencies	No Dependencies
FPT_FLS.1/GP	No Dependencies	No Dependencies
FCO_NRO.2/GP	FIA_UID.1 Timing of identification	FIA_UID.1/GP
FDP_UIT.1/GP	(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_IFC.2/GP-ELF, FDP_IFC.2/GP-KL, FTP_ITC.1/GP
FIA_UID.1/GP	No Dependencies	No Dependencies
FMT_SMF.1/GP	No Dependencies	No Dependencies
FMT_SMR.1/GP	FIA_UID.1 Timing of identification	FIA_UID.1/GP
FTP_ITC.1/GP	No Dependencies	No Dependencies
FMT_MSA.1/GP	(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/GP-ELF, FDP_IFC.2/GP-KL, FMT_SMR.1/GP, FMT_SMF.1/GP
FMT_MSA.3/GP	FMT_MSA.1 Management of security attributes, FMT_SMR.1 Security roles	FMT_MSA.1/GP, FMT_SMR.1/GP
Limited Mode Security Functional Requirements		

FDP_ACF.1/LM	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.2/LM FMT_MSA.3/LM
FDP_ACC.2/LM	FDP_ACF.1 Security attribute based access control	FDP_ACF.1/LM
FMT_MSA.1/LM	[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/LM FMT_SMR.1/GP FMT_SMF.1/LM
FMT_MSA.3/LM	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1/LM FMT_SMR.1 dependency not met since no associated roles are required
FMT_SMF.1/LM	No dependencies	N/A
FIA_UID.1/LM	No dependencies	N/A
FIA_UAU.1/LM	FIA_UID.1 Timing of identification	FIA_UID.1/LM
OS Management Security Functional Requirements		
FDP_ACC.2/OSM	FDP_ACF.1 Security attribute-based access control	FDP_ACF.1/OSM
FDP_ACF.1/OSM	FDP_ACC.1 Subset access control	FDP_ACC.2/OSM
	FMT_MSA.3 Static attribute initialisation	FMT_MSA.3/OSM
FIA_UID.1/GP	No dependencies	N/A
FMT_MSA.3/OSM	FMT_SMR.1 Security roles FMT_MSA.1 Management of security attributes	FMT_SMR.1/GP
FMT_SMF.1/OSM	No dependencies	N/A
FDP_UIT.1/OSM	FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control	FDP_ACC.2/OSM

	FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path	FTP_ITC.1/OSM
FTP_ITC.1/OSM	No dependencies	N/A
FPT_FLS.1/OSM	No dependencies	N/A
Security Functional Requirements from the Smart Card Platform		
FPT_PHP.3	No dependencies	N/A
FCS_RNG.1/PTG.2	No dependencies	N/A
FCS_RNG.1/DRG.3	No dependencies	N/A
FPT_EMSEC.1	No dependencies	N/A

Table 12: Dependency for SFRs introduced in this ST

The dependency FMT_MSA.1 of FMT_MSA.3/OSM is discarded as no management capabilities are provided for OSM security attributes.

7.3.3 Rationale for Security Assurance Requirements

The chosen assurance level EAL5 and the augmentation with the requirements ALC_DVS.2 and AVA_VAN.5 were chosen in order to meet the assurance expectations explained in the following paragraphs. In Table 10, the different assurance levels are shown as well as the augmentations. The augmentations are in compliance with the Protection Profile.

An assurance level of EAL5 with the augmentation AVA_VAN.5 and ALC_DVS.2 has been done, exceeding the requirement claimed by the [[JCPP]]. This evaluation assurance package was selected because EAL5 represents a meaningful increase in assurance from EAL4 by requiring semiformal design descriptions, a more structured and analyzable structure, and improved mechanisms and procedures that provide confidence that the TOE will not be tampered.

ALC_DVS.2 Sufficiency of security measures

Development security is concerned with physical, procedural, personnel and other technical measures that may be used in the development environment to protect the TOE.

In the particular case of a Security IC the TOE is developed and produced within a complex and distributed industrial process which must especially be protected. Details about the implementation, (e.g. from design, test and development tools as well as Initialization Data) may make such attacks easier. Therefore, in the case of a Security IC, maintaining the confidentiality of the design is very important.

This assurance component is a higher hierarchical component to EAL5 (which only requires ALC_DVS.1). ALC_DVS.2 has no dependencies.

AVA_VAN.5 Advanced methodical vulnerability analysis

Due to the intended use of the TOE, it must be shown to be highly resistant to penetration attacks. This assurance requirement is achieved by the AVA_VAN.5 component.

Independent vulnerability analysis is based on highly detailed technical information. The main intent of the evaluator analysis is to determine that the TOE is resistant to penetration attacks performed by an attacker possessing high attack potential.

AVA_VAN.5 has dependencies to ADV_ARC.1 “Security architecture description”, ADV_FSP.5 “Security enforcing functional specification”, ADV_TDS.4 “Basic modular design”, ADV_IMP.1 “Implementation representation of the TSF”, AGD_OPE.1 “Operational user guidance”, and AGD_PRE.1 “Preparative procedures”.

All these dependencies are satisfied by EAL5.

It has to be assumed that attackers with high attack potential try to attack Security ICs like smart cards used for digital signature applications or payment systems. Therefore, specifically AVA_VAN.5 was chosen in order to assure that even these attackers cannot successfully attack the TOE.

8 IC Composition rationale

8.1 Common Criteria rationale

Assurance level of the Platform-TOE is EAL6.

Assurance level of the composite-TOE is EAL5 augmented with ALC_DVS.2 and AVA_VAN.5.

Assurance level claimed in the composite-ST is consistent with the assurance level claimed in the Platform-ST.

8.2 Compatibility between Security Objectives (TOE and IC)

IC Objectives	Rationale	Link to the composite-TOE
O.Leak-Inherent	Covered by both IC and current evaluation.	O.SCP.IC O.SCP.SUPPORT
O.Phys-Probing	Covered by both IC and current evaluation.	O.SCP.IC O.SCP.SUPPORT
O.Malfunction	Covered by both IC and current evaluation.	O.OPERATE
O.Phys-Manipulation	Covered by both IC and current evaluation.	O.SCP.IC O.SCP.SUPPORT
O.Leak-Forced	Covered by both IC and current evaluation.	O.SCP.IC O.SCP.SUPPORT
O.Abuse-Func	Covered by both IC and current evaluation.	O.SCP.SUPPORT
O.Identification	Covered by both IC and current evaluation.	O.TOE-IDENTIFICATION
O.RND	Covered by both IC and current evaluation.	O.RNG
O.TDES	Covered by both IC and current evaluation.	O.CIPHER
O.AES	Covered by both IC and current evaluation.	O.CIPHER
O.KDF	Covered by both IC and current evaluation.	O.CIPHER
O.Mem-Access	Covered by both IC and current evaluation.	O.SCP.SUPPORT
O.SFR-Access	Covered by the IC evaluation.	-
O.RSA	Covered by both IC and current evaluation.	O.CIPHER
O.ECC	Covered by both IC and current evaluation.	O.CIPHER
O.X25519	Covered by the IC evaluation.	-
O.SHA256	Covered by the IC evaluation.	-
O.HMAC	Covered by the IC evaluation.	-
O.CRC	Covered by the IC evaluation.	-

Table 13: Compatibility between Security Objectives (TOE and IC)

8.3 Compatibility between SFRs (TOE and IC)

IC SFRs are separated in the following groups as defined in [7]:

- IP_SFR: irrelevant IC SFR not being used by the current TOE.
- RP_SFR-SERV: relevant IC SFR being used by the current TOE to implement a security service with associated TSFI.

- RP_SFR-MECH: relevant IC SFR being used by the current evaluation because its security properties providing protection attacks to the TOE.

IC SFR	Rationale	Link to composite-TOE
FRU_FLT.2	RP_SFR-SERV	FAU_ARP.1
FPT_FLS.1	RP_SFR-SERV	FPT_FLS.1
FMT_LIM.1	IP_SFR	The Composite TOE does not use Platform's test features after Platform delivered.
FMT_LIM.2	IP_SFR	The Composite TOE does not use Platform's test features after Platform delivered.
FAU_SAS.1	RP_SFR-SERV	FDP_ACF.1/OSM FMT_SMF.1/LM FIA_UID.1/GP FIA_UID.1/LM FIA_UAU.1/GP FIA_UAU.1/LM
FDP_SDC.1	RP_SFR-MECH	SM.CL_INVK
FDP_SDI.2	RP_SFR-SERV	FDP_SDI.2/DATA FDP_SDI.2/ARRAY FDP_SDI.2/RESULT
FPT_PHP.3	RP_SFR-SERV	FPT_PHP.3
FDP_ITT.1	RP_SFR-MECH	SM.CL_INVK. TOE invokes the CL APIs to perform cryptographic calculations. These APIs prevent the disclosure of sensitive data of the Composite TOE when it is transmitted between memory, CPU and cryptographic co-processor.
FPT_ITT.1	RP_SFR-MECH	SM.CL_INVK. TOE invokes the CL APIs to perform cryptographic calculations. These APIs prevent the disclosure of sensitive data of the Composite TOE when it is transmitted between memory, CPU and cryptographic co-processor.
FDP_IFC.1	RP_SFR-SERV	SM.CL_INVK
FCS_RNG.1/PTG.2	RP_SFR-SERV	FCS_RNG.1/PTG.2
FCS_COP.1/TDES	RP_SFR-SERV	FCS_COP.1.1/TDES FCS_COP.1.1/DESMAC
FCS_COP.1/AES	RP_SFR-SERV	FCS_COP.1.1/AES

		FCS_COP.1.1/AES_MAC
FCS_CKM.4/TDES	RP_SFR-SERV	FCS_CKM.4.1
FCS_CKM.4/AES	RP_SFR-SERV	FCS_CKM.4.1
FCS_RNG.1/DRG.3	RP_SFR-SERV	FCS_RNG.1/DRG.3
FCS_COP.1/RSA	RP_SFR-SERV	FCS_COP.1.1/RSA FCS_COP.1.1/RSASignature FCS_COP.1.1/DAP
FCS_COP.1/ECDSA	RP_SFR-SERV	FCS_COP.1.1/ECDSA FCS_COP.1.1/DAP
FCS_COP.1/ECDH	IP_SFR	
FCS_CKM.1/RSA	RP_SFR-SERV	FCS_CKM.1.1/RSA
FCS_CKM.1/ECC	RP_SFR-SERV	FCS_CKM.1.1/ECC
FCS_CKM.5/KDF	RP_SFR-SERV	FCS_CKM.5
FCS_CKM.4/CL	RP_SFR-MECH	FCS_CKM.4
FMT_SMF.1	RP_SFR-MECH	SM.MMU The Composite TOE is running in CPU privileged level. To set the MMU, The Composite TOE invokes the configuration function in the HAL of the Platform.
FDP_ACC.1	RP_SFR-MECH	SM.MMU The register access for MMU setting conforms to the Memory and Register Access Control Policy of the Platform.
FDP_ACF.1	RP_SFR-MECH	SM.MMU The register access for MMU setting conforms to the Memory and Register Access Control Policy of the Platform.
FMT_MSA.3	RP_SFR-MECH	SM.MMU The MMU doesn’t allow any user to set the default value to the Platform’s security attributes for its Memory Access Control Policy
FMT_MSA.1	RP_SFR-MECH	SM.MMU The Composite TOE is running in in CPU privileged level. The register access for MMU setting conforms to the Memory and Register Access Control Policy of the Platform.
FCS_COP.1/X25519	IP_SFR	
FCS_COP.1/SHA256	RP_SFR-SERV	FCS_COP.1.1/DAP

		FCS_COP.1.1/RSASignature FCS_COP.1.1/ECDSA
FCS_COP.1/HMAC	RP_SFR-SERV	FCS_COP.1.1/RSA FCS_COP.1.1/RSASignature FCS_COP.1.1/ECDSA
FCS_COP.1/CRC	RP_SFR-SERV	FDP_SDI.2.1/DATA FDP_SDI.2.1/RESULT FDP_SDI.2.1/ARRAY
FCS_CKM.1/X25519	IP_SFR	

Table 14: Compatibility between SFRs (TOE and IC).

Application note: FCS_COP.1/HMAC do not directly provide services to the COS for the linked SFRs. However, the SHA primitives, upon which the HMAC relies, were included within the scope of the IC evaluation and assessed under the HMAC testing requirements.

8.4 Compatibility between security objectives for the environment (TOE and IC)

IC OEs are separated in the following groups as defined in [7]:

- **IrOE**: The objectives for the environment being not relevant for the Composite-ST.
- **CfOE**: The objectives for the environment being fulfilled by the Composite-ST automatically.
- **SgOE**: The remaining Objectives for the environment of the Platform-ST belonging neither to the IrOE nor CfOE. Exactly this group makes up the significant objectives for the environment for the Composite-ST, which shall be addressed in the Composite-ST.

IC OEs	Rationale	Group	Link to the composite-TOE
OE.Resp-Appl	This objective for the environment ensures that the TOE will not disclose security relevant user data of the Composite TOE to unauthorised users or processes when communicating with a terminal. It is covered by the current evaluation.	SgOE	OE.Resp-Appl
OE.Process-Sec-IC	This objective for the environment ensures that the TOE should be maintained confidentiality and integrity of the TOE and of its manufacturing and test data using security procedures during delivery. It is covered by the current evaluation.	SgOE	OE.Process_Sec_IC

Table 15: Compatibility between security objectives for the environment (TOE and IC).

9 TOE Summary Specification

9.1 Security Functionality of the TOE

The TOE Security Functionality (TSF) is composed of Security Functions (SF) and Security Mechanisms (SM). They together fulfill the security functional requirements (SFR) for the TOE.

The Security Functions and Security Mechanisms related to SFRs of the TOE are summarized in Table 16 and described in section 9.2.

Security Function / Security Mechanism	Name	Name Fulfilled SFR
SF.JCVM	Java Card Virtual Machine	FDP_IFC.1/JCVM FDP_IFF.1/JCVM FMT_MSA.1/JCVM FMT_MSA.1/JCRE FMT_MSA.3/JCVM FMT_SMR.1 FMT_SMF.1 FTP_ITC.1/GP FDP_ROL.1/FIREWALL FDP_ACF.1/FIREWALL FDP_ACC.2/FIREWALL FMT_MSA.2/FIREWALL_JCVM FMT_MSA.3/FIREWALL FIA_UID.2/AID FAU_ARP.1 FPT_FLS.1 FDP_RIP.1/ABORT
SF.GP_CCM	GlobalPlatform Management	FCO_NRO.2/GP FDP_IFF.1/GP-ELF FDP_IFC.2/GP-ELF FDP_ITC.2/GP-ELF FDP_IFC.2/GP-KL FDP_IFF.1/GP-KL FDP_ITC.2/GP-KL FPT_TDC.1/GP FDP_UIT.1/GP FIA_UID.1/GP FMT_MSA.1/GP FMT_MSA.3/GP FMT_SMR.1/GP FMT_SMF.1/GP FTP_ITC.1/GP FIA_UAU.1/GP FIA_UAU.4/GP

		FPT_FLS.1/GP FPT_TDC.1 FIA_ATD.1/AID FIA_UID.2/AID FIA_USB.1/AID FDP_ACC.2/ADEL FDP_ACF.1/ADEL FDP_RIP.1/ADEL FDP_RIP.1/bArray FMT_SMF.1/ADEL FMT_MSA.1/ADEL FMT_MSA.3/ADEL FMT_SMR.1/ADEL FPT_FLS.1/ADEL FMT_MTD.1/JCRE FMT_MTD.3/JCRE FPT_RCV.3/INSTALLER FCS_COP.1 FAU_ARP.1 FPT_FLS.1
SF.CRYPTO	Cryptographic Functionality	FCS_CKM.1 FCS_CKM.4 FCS_CKM.5 FCS_COP.1 FDP_RIP.1/KEYS
SF.RNG	Random Number Generator	FCS_RNG.1/PTG.2 FCS_RNG.1/DRG.3
SF.KEY_STORAGE	Secure Key Storage	FCS_CKM.1 FCS_CKM.4 FDP_SDI.2/DATA FAU_ARP.1 FPT_FLS.1 FPR_UNO.1 FDP_RIP.1/KEYS
SF.LIMITED_MODE	Limited Mode	FDP_ACC.2/LM FDP_ACF.1/LM FMT_MSA.1/LM FMT_MSA.3/LM FMT_SMF.1/LM FIA_UID.1/LM FIA_UAU.1/LM
SF.OS_MANAGEMENT	Operating System Management	FDP_ACC.2/OSM FDP_ACF.1/OSM FIA_UID.1/GP FIA_UAU.1/GP FMT_MSA.3/OSM

		FMT_SMF.1/OSM FMT_SMR.1/GP FDP_UIT.1/OSM FTP_ITC.1/OSM FPT_FLS.1/OSM
SF.OBJ_MNG	Java Object Management	FDP_RIP.1/OBJECTS FDP_RIP.1/ODEL FPT_FLS.1/ODEL FAU_ARP.1 FPT_FLS.1
SF.TRANSIENT_MEM	Memory Management	FDP_RIP.1/TRANSIENT FIA_ATD.1/AID FDP_RIP.1/APDU FDP_RIP.1/bArray FDP_RIP.1/GlobalArray
SF.PERS_MEM	Persistent Memory Management	FAU_ARP.1 FPT_FLS.1 FDP_ROL.1/FIREWALL FDP_RIP.1/ABORT
SF.SENS_ARRAY	Data Error Detection	FAU_ARP.1 FPT_FLS.1 FDP_SDI.2/ARRAY
SF.EXCP_HANDLE	Hardware Protection and Error Handling	FAU_ARP.1 FPT_FLS.1 FPT_PHP.3
SF.PIN	PIN Management	FDP_SDI.2/DATA FPR_UNO.1
SF.SCA	Side-Channel Protection	FPR_UNO.1 FPT_EMSEC.1
SF.SENS_RES	Sensitive Result	FAU_ARP.1 FPT_FLS.1 FDP_SDI.2/RESULT

Table 16 Security Functions/Mechanisms of the TOE

9.2 Security Functions

9.2.1SF.JCVM

SF.JCVM provides the bytecode interpreter and the firewall to execute the bytecodes correctly to access the java objects under the proper access control according to the

specifications [8], [9] and [10]. This fulfills the SFRs FDP_IFC.1/JCVM, FDP_IFF.1/JCVM, FMT_MSA.1/JCVM, FMT_MSA.1/JCRE, FMT_MSA.3/JCVM, FMT_SMR.1, FMT_SMF.1, FDP_ROL.1/FIREWALL, FDP_ACF.1/FIREWALL, FDP_ACC.2/FIREWALL and FIA_UID.2/AID. SF.JCVM supports FAU_ARP.1, FPT_FLS.1 by throwing Java Exceptions according to specifications.

All values for security attributes are initialized and assigned by the system itself which fulfills FMT_MSA.2/FIREWALL_JCVM and FMT_MSA.3/FIREWALL.

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.

9.2.2SF.GP_CCM

SF.GP provides the card content management functionality and prevent users who are not authorized or have no respective rights to do it. It also provides a secure communication channel for sensitive data exchange to prevent from tampering and disclosure according the GlobalPlatform Specification [11] and GlobalPlatform Amendments A[13], D[15] and E[16].

9.2.3SF.CRYPTO

SF.CRYPTO provides key creation, key management, key derivation, key deletion and cryptographic functionality against state-of-the-art attacks, including side-channel analysis. It provides the API in accordance to the Java Card API Specification [9].

9.2.4SF.RNG

SF.RNG provides random number generation functions TRNG and DRNG, which conform to class PTG.2 and DRG.3 classes in AIS 20/31[6].

9.2.5SF.KEY_STORAGE

SF.KEY_STORAGE provides a secure data storage for keys. Cryptographic keys are stored with integrity protection.

9.2.6SF.LIMITED_MODE

SF.LIMITED_MODE prevents the TOE from further attack by providing a Limited Mode which TOE will enter in case that a determined amount of potentially malicious physical events are detected. In this mode, only limited functionality is available.

In Limited Mode, only commands to select ISD or seRoot, authenticate, and get TOE version are allowed. All other operations return error codes. This fulfills SFRs FDP_ACC.2/LM, FDP_ACF.1/LM, FMT_MSA.1/LM, FMT_MSA.3, FMT_SMF.1.

After ISD identification and authentication D.EXCEPTION_COUNTER can be reset to default value (64). This is covered by FIA_UID.1/LM and FIA_FIA_UAU.1/LM.

All operations and objects are covered by this access control SFP if Limited mode is triggered. FDP_ACF.1/LM models a Limited Mode access control SFP over all objects. In this case, after authentication, the D. EXCEPTION_COUNTER can be reset by S.SD, and all

operations except those claimed at the beginning of this section are denied if D.EXCEPTION_COUNTER reaches the limit.

9.2.7SF.OS_MANAGEMENT

SF.OS_MANAGEMENT models the method to update and configure the TOE securely for an OS Administrator, which can be an end user. It prevents the updating from unexpected update packages. The installation and activation of update packages is only allowed after its correct signature verification. Update packages are signed by Goodix using whether ALG_AES_MAC_128_NOPAD or ALG_ECDSA_SHA256. The TOE supports both signature verification mechanisms as described in FCS_COP.1.1/AES_MAC and FCS_COP.1.1/ECDSA.

After authentication (FIA_UID.1/GP and FIA_UAU.1/GP) and establishment of a SCP90 channel between the CAD and the TOE (FTP_ITC.1/OSM), the Image Sequence Number of the OS update is checked to guarantee that is larger than the current one to protect from replay as modeled by the access control policy FDP_ACC.2/OSM, FDP_ACF.1/OSM and FDP_UIT.1/OSM. Only seRoot acting on behalf of an OS Administrator is capable of performing such OS updates and configuration management.

After authentication, the integrity of D.OS_IMAGE is checked (FDP_UIT.1/OSM) and Image Sequence Number is updated after successfully update of OS Patch Update.

The TOE preserves a secure state after failure (FPT_FLS.1/OSM) not leading the update to be carried out.

9.2.8SF.OBJ_MNG

SF.OBJ provides the creation and deletion of java objects under the proper memory resource management and access right control according to the Java Card Runtime Environment Specification [8]. SF.OBJ throws Java Exceptions in case object creation error.

9.2.9SF.TRANSIENT_MEM

SF.TRANSIENT_MEM provides memory deletion for transient arrays, global arrays, and logical channels according to the Java Card Runtime Environment Specification [8].

9.2.10 SF.PERS_MEM

SF.PERS_MEM provides atomic write operations and transaction management according to the Java Card Runtime Environment Specification[8].

9.2.11 SF.SENS_ARRAY

SF.SENS_ARRAY defines a type of array with a checksum of its content. Applications can use it to check its integrity before access it for Java arrays [9]. The API throws Java exceptions in case the checksum in invalid.

9.2.12 SF.EXCP_HANDLE

SF.EXCP_HANDLE stops the current execution of TOE instructions immediately since any security exception is detected. That is to prevent TOE from working incorrectly risking disclosure of sensitive data or manipulation of TOE behaviors. It also prevents unlimited brute trying on TOE from attackers.

9.2.13 SF.PIN

SF.PIN provides an authentication method based on PIN to applets to identify and verify the users securely, which prevent TOE from the disclosure of PIN value and malicious trying brutally.

9.2.14 SF.SCA

SF.SCA provides side-channel protection function for timing attack, SPA, DPA, EMA and DEMA to prevent keys and PINs leakage while processing them.

9.2.15 SF.SENS_RES

SF.SENS_RES provides applications to check whether a method executes correctly so as to prevent some critical operations or variables are manipulated or bypassed.

10 Bibliography

10.1 Standards

[CC1]	Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model, Version 3.1, Revision 5, April 2017, CCMB-2017-04-001
[CC2]	Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002
[CC3]	Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-003
[ICPP]	Security IC Platform Protection Profile, Version 1.0, 13th Jan. 2014, registered and certified by Bundesamtfür Sicherheitinder Informationstechnik (BSI) under the reference BSI-PP-0084
[JCPP]	Java Card System - Open Configuration Protection Profile, version 3.1.0 (Apr. 2020), published by oracle, Inc. (bsi-cc-pp-0099-2020)
[BSI-PP-0055]	Protection Profile Machine Readable Travel Document with "ICAO Application", Basic Access Control(BAC PP), certified under reference BSI-CC-PP-0055-009, Version 1.10, BSI-CC-PP-0055
[GPC_SE_PP]	GlobalPlatform Technology Secure Element Protection Profile Version 0.0.0.21 (Target v1.0). January 2020. Document Reference: GPC_SPE_174.
[ACM]	SOG-IS Crypto Evaluation Scheme - Agreed Cryptographic Mechanisms, version 1.3 February 2023
[JIL_SRCL]	Joint Interpretation Library - Security requirements for post-delivery code loading - Version 1.0, February 2016
[1]	NIST SP 800-38D: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007, Morris Dworkin, National Institute of Standards and Technology

[2]	NIST SP 800-108: Recommendation for Key Derivation Using Pseudorandom Functions, revised, October 2009
[3]	FIPS PUB 186-5-2023: Digital Signature Standard, Federal Information Processing Standards Publication, 2023, February, National Institute of Standards and Technology
[4]	FIPS PUB 197-2001: ADVANCED ENCRYPTION STANDARD (AES), Federal Information Processing Standards Publication 197, November 26, 2001, U.S. Department of Commerce/National Institute of Standards and Technology
[5]	IETF RFC 5246: The Transport Layer Security (TLS) Protocol, version 1.2, August 2008
[6]	Anwendungshinweise und Interpretationen zum Schema (AIS), Funktionalitätsklassen und Evaluationsmethodologie für deterministische Zufallszahlengeneratoren, Version 3, 15.05.2013, Bundesamt für Sicherheit in der Informationstechnik (BSI)
[7]	JIL-Composite product evaluation for Smart Cards and similar devices, Version 1.5.1, May 2018-
[8]	Runtime Environment Specification, Java Card™ Platform, Version 3.1, Classic Edition, 2019-11
[9]	Application Programming Interface, Java Card™ Platform, v3.1 Classic Edition, 2019-11
[10]	Virtual Machine Specification, Java Card™ Platform, v3.1 Classic Edition, 2019-11
[11]	GlobalPlatform Card Specification v2.3.1, 2018-03
[12]	Java Card API and Export File for Card Specification v2.2.1 (org.globalplatform) v1.6, 2014-03
[13]	GlobalPlatform Card Specification 2.3 Amendment A v1.2 - Confidential Card Content Management, 2019-07

[14]	GlobalPlatform Card Specification 2.3 Amendment C v1.3 – Contactless Services, 2019-07
[15]	GlobalPlatform Card Specification 2.3 Amendment D v1.2 – Secure Channel Protocol '03', 2020-04
[16]	GlobalPlatform Card Specification 2.3 Amendment E v1.1 – Security Upgrade for Card Content Management, 2016-10
[17]	GlobalPlatform Technology Secure Element Configuration v2.0, 2018-08
[18]	ETSI TS 102 705 UICC Application Programming Interface for Java Card™ for Contactless Applications V13.0.0 (2019-05)
[19]	GM_T SM2-2012 Elliptic Curve Public Key Cryptography
[20]	GM_T SM3-2012 Cryptographic Hash Algorithm
[21]	GM_T SM4-2012 Block Cipher Algorithm
[22]	IETF RFC 5869: HMAC-based Extract-and-Expand Key Derivation Function (HKDF)
[23]	ICAO MRTD Doc 9303(Machine Readable Travel Documents Part 11: Security Mechanisms for MRTDs Eighth Edition, 2021)
[24]	NIST.SP800-38C: Recommendation for Block Cipher Modes of Operation – The CCM Mode for Authentication and Confidentiality
[25]	China financial integrated circuit card specification (JR/T 0025.1 2018, JR/T 0025.13 2018, JR/T 0025.14-2018)
[26]	Technical specifications on IC card for urban public transport ticket (JT/T 978.2-2023)
[27]	Strongbox Keymaster HAL, version 1.0, Google, November 2018

[28]	Digital key release 3.0, Version 3.0, Car Connectivity Consortium, October 2021.
[29]	ANSI X9.63: Public Key Cryptography For The Financial Services Industry: Key Agreement and Key Transport Using Elliptic Curve Cryptography, December 2011, American National Standards Institute
[30]	GlobalPlatform Card Common Implementation Configuration v2.1, 2018-07

10.2 Developer Documents

[31]	Security Target of Security Chip GSE20 Series with IC Dedicated Software V1.10, 19 June 2024, Shenzhen Goodix Technology Co., Ltd.
[32]	GEOP02 User Manual V1.9, 2025
[33]	GEOP seRoot User Manual, V0.7, 2025
[34]	GEOP02 Preparative Procedures V1.8, 2025
[35]	GEOP02 Operational User Guidance V1.7, 2025
[36]	GEOP02 Security Guidance V1.5, 2025
[37]	Goodix API Specification v2.0, 2024

11 Legal and Contact Information

Trademarks and Permissions

GOODiX and other Goodix trademarks are trademarks of Shenzhen Goodix Technology Co., Ltd. All other trademarks and trade names mentioned in this document are the property of their respective holders.

Disclaimer

Information contained in this document is intended for your convenience only and is subject to change without prior notice. It is your responsibility to ensure its application complies with technical specifications.

Shenzhen Goodix Technology Co., Ltd. (hereafter referred to as "Goodix") makes no representation or guarantee for this information, express or implied, oral or written, statutory or otherwise, including but not limited to representation or guarantee for its application, quality, performance, merchantability or fitness for a particular purpose. Goodix shall assume no responsibility for this information and relevant consequences arising out of the use of such information.

Without written consent of Goodix, it is prohibited to use Goodix products as critical components in any life support system. Under the protection of Goodix intellectual property rights, no license may be transferred implicitly or by any other means.

Shenzhen Goodix Technology Co., Ltd.

Headquarters: 2F. & 13F., Tower B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China

TEL: +86-755-33338828 FAX: +86-755-33338099

Website: http://www.goodix.com