Thales Mobile Digital Key v2.5-15E0 Security Target Public version

REVISION HISTORY

Version	State	Date	Author	Description of the modifications
1.4p	Final	06/05/2025	THALES DIS	Created from evaluated ST (v1.4)

CONTENTS

1. REFERENCE DOCUMENTS

1.1. EXTERNAL REFERENCES [ER]

[ISO]	ISO references
[ISO 7816]	Identification cards – Integrated circuit(s) cards with contacts - Books 1 to 9
[ISO 14443]	Identification cards — Contactless integrated circuit(s) cards — Proximity cards — Books 1 to 4
[ISO/IEC 10116]	ISO/IEC 10116 Information Technology - Security techniques, Modes of operation for an n-bit block cipher, 2017
[ISO/IEC 14888-2]	ISO/IEC 14888-2 Information technology – Security techniques, Digital signatures with appendix – Part 2: Integer factorization based mechanisms, 2008
[ISO/IEC 18033-3]	ISO/IEC 18033-3 Information technology - Security techniques, Encryption algorithms - Part 3: Block ciphers, 2010
[ISO/IEC 9797-1]	ISO/IEC 9797-1 Information Technology - Security techniques, Message Authentication Codes (MACs), Part 1: Mechanisms using a block cipher, , 2011
[ISO/IEC 9797-2:2011]	ISO/IEC 9797-2 Information Technology - Security techniques, Message Authentication Codes (MACs), Part 2: Mechanisms using a dedicated hash-function, 2011
[Javacard]	**Javacard references**
[JCRE310]	Java Card 3 Platform - Runtime Environment Specification, Classic Edition Version 3.1.0, November 2019
[JCVM310]	Java Card 3 Platform - Virtual Machine Specification, Classic Edition Version 3.1.0, November 2019
[JCAPI310]	Java Card 3 Platform - Java Card API, Classic Edition Version 3.1.0, November 2019
[GP]	**Global Platform references**
[GPCS]	GlobalPlatform Technology - Card Specification v2.3.1, March 2018 Reference: GPC_SPE_034
[Amd A]	GlobalPlatform Card - Confidential Card Content Management Card Specification v2.3 – Amendment A v1.2 Reference: GPC_SPE_007
[Amd B]	GlobalPlatform Technology - Remote Application Management over HTTP Card Specification v2.3 – Amendment B v1.2 Reference: GPC_SPE_011
[Amd C]	GlobalPlatform Card – Contactless services Card Specification v2.3 – Amendment C v1.3 Reference: GPC_SPE_025
[Amd D]	GlobalPlatform Card Technology - Secure Channel Protocol '03' Card Specification v2.3 – Amendment D v1.2 Reference: GPC_SPE_014
[Amd E]	GlobalPlatform Card Technology - Security Upgrade for Card Content Management Card Specification v2.3 – Amendment E v1.1 Reference: GPC_SPE_042
[Amd F]	GlobalPlatform Card - Secure Channel Protocol '11' Card Specification v2.3 – Amendment F v1.3 Reference: GPC_SPE_093
[Amd H]	GlobalPlatform Card - Executable Load File Upgrade

	Card Specification v2.3 – Amendment H v1.1 Reference: GPC_SPE_120
[CIC]	Common Implementation Configuration v2.1 Reference: GPC_GUI_080
[SE_CFG]	GlobalPlatform Secure Element Configuration v2.0 Reference: GPC_GUI_049
[PF]	GlobalPlatform Card - Privacy Framework v1.0.1 Reference: GPC_SPE_100
[CC]	**Common Criteria references**
[CC-1]	Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model CCMB-2017-04-001, Version 3.1 Revision 5, April 2017.
[CC-2]	Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements CCMB-2017-04-002, Version 3.1 Revision 5, April 2017.
[CC-3]	Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Components CCMB-2017-04-003, Version 3.1 Revision 5, April 2017.
[CCDB]	Common Criteria Supporting Document, Mandatory Technical Document – Composite product evaluation for Smart Cards and similar devices Ref: CCDB-2012-04-001, Version 1.2, April 2012.
[PP-JCS]	Java Card System Protection Profile – Open Configuration BSI-CC-PP-0099-V2-2020, Version 3.1, April 2020
[PP-GP]	GlobalPlatform Technology - Secure Element Protection Profile Ref: GPC_SPE_174, Version 1.0
[PP-0084]	Security IC Platform Protection Profile with augmentation Packages Ref: BSI-CC-PP-0084-2014
[ST_IC]	ST54L A02 Security Target for composition Ref: SMD_ST54L_ST_22_001, Revision A02.1, February 2024
[CCC]	**CCC references**
[CCC-TS-101]	Car Connectivity Consortium, Digital Key Release 3, Technical Specification Ref: CCC-TS-101, V 1.1.3
[PP-CCC-CP-023]	Car Connectivity Consortium Protection Profile of the Digital Key Applet Ref: CCC-CP-023 v1.0
[NIST]	**Other references**
[NIST SP 800-38B]	NIST, SP800-38B Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, May 2005
[NIST-SP-800-56A]	Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography - SP 800-56A Rev. 3 - April 2018
[FISP140-3]	Security Requirements for Cryptographic Modules - FIPS PUB 140-3 Federal Information Processing Standards Publication (Supersedes FIPS PUB 140-2)
[FIPS 180-4]	NIST, Secure Hash, Standard (SHS), 2012
[FIPS PUB 197]	Federal Information Processing Standards Publication 197 (FIPS PUB 197), Advanced Encryption Standard (AES), 2001
[NIST SP 800-38A]	NIST, SP800-38A Recommendation for Block Cipher Modes of Operation:

	Methods and Techniques
[FIPS PUB 186-4]	NIST, Digital Signature Standard (DSS), 2013
[OTHERS]	**Other references**
[AIS31]	A proposal for: Functionality classes for random number generators, version 2.0, 18.09.2011, Bundesamt für Sicherheit in der Informationstechnik
[BSI TR-03111]	Technical Guideline BSI TR-03111 - Elliptic Curve Cryptography - Version 2.10- Date: 2018-06-01
[RFC 5869]	IETF - HMAC-based Extract-and-Expand Key Derivation Function (HKDF) May 2010
[ANSI X9.62]	The Elliptic Curve Digital Signature Algorithm (ECDSA), American National Standard for Financial Services, November 16, 2005

1.2. INTERNAL REFERENCES [IR]

[AGD]	**TOE GUIDANCE DOCUMENTATION**
[AGD]	Thales Mobile Digital Key v2.5-15E0 – AGD_PRE Ref: D1627744 Thales Mobile Digital Key v2.5-15E0 – AGD_OPE Ref: D1627742 Thales Mobile Digital Key - 2.5 – User Guide Ref: D1634064 PLATFORM_IdentificationConfigurability For Connected eSE 5.3.4 v1.1 Ref: D1611592
[Others]	**Other internal references**
[ST_PF]	Connected eSE 5.3.4 Platform Security Target Ref: D1582688, v 1.4, July 2024

2. SECURITY TARGET INTRODUCTION

2.1. SECURITY TARGET IDENTIFICATION

Title :	Thales Mobile Digital Key v2.5-15E0 Security Target
Version :	1.4p
ST Reference :	D1623707
Origin :	Thales
IT Security Evaluation scheme :	Thales SIX

2.2. TOE IDENTIFICATION

Product Name :	Thales Mobile Digital Key
Security Controllers :	ST54L A02
TOE Name :	Thales Mobile Digital Key
TOE Version :	2.5-15E0
TOE documentation :	Guidance [AGD]
Composition elements:	
Composite TOE identifier:	Connected eSE 5.3.4 v1.1
Composite TOE Version:	V1.1

The TOE identification is provided in particular by a dedicated command GET DATA.

Please refer to [AGD] for more details.

Note that the TOE name used in the rest of the document is mDKA v2.5-15E0.

2.3. TOE OVERVIEW

2.3.1. TOE TYPE

The Target of Evaluation (TOE) type is a Javacard digital key applet built on an eSE/eUICC open platform implementing the Javacard and GlobalPlatform standards.

The TOE is implementing the [CCC-TS-101] specification.

The mDKA v2.5-15E0 product addresses the consumers electronics mobile market.

For the present evaluation, the TOE is the mDKA v2.5 Application and the underlying platform which supports its functionality. Therefore, the TOE boundaries encompass:

- **The hardware layer composed of the ST54L Rev C integrated circuit.**
- **The Connected eSE 5.3.4 v1.1 platform, which is the operating system of the product.**
- **The Thales mDKA v2.5 Applet Software**
- **The associated guidance documentation [AGD]**.

2.3.2. USAGE AND MAJOR SECURITY FEATURES OF THE TOE

2.3.2.1. USAGE

The TOE enables mobile devices to store, authenticate, and share Digital Keys for vehicles in a secure, privacy preserving way that works everywhere, even when the mobile device's battery is low or when mobile device or vehicle don't have Internet connectivity.

The product is able to communicate over [ISO 7816] (T=0, T=1) and SWP contact protocols. Inserted in an NFC-enabled mobile device, it allows communication with a terminal using the standard [ISO 14443] communication protocol.

Built upon an eSE/eUICC open platform implementing the Javacard and GlobalPlatform standards, the application software implements following services:

- **Owner Pairing:** The Owner device's role is identifying the device hosting the owner key for a given vehicle. This enables any mobile device that meets the technology and security requirements of Digital key to be paired as an owner device with a vehicle. Each vehicle may have only one owner device; an owner device has full authority over the paired vehicle and all associated Digital Keys. A given device can host several owner keys (in case someone owns multiple cars) but for a given car there is a single owner device.
- **Vehicle Access/Engine start:** Digital Key may be used to access a vehicle, start the engine, mobilize the vehicle, or authorize any other operation by placing a mobile device near an NFC reader, without requiring you to interact with a user interface of the mobile device (e.g., an app). In order for this operation to take place, the vehicle and the device SHALL be mutually authenticated first, and the vehicle verifies that the mobile device's Digital Key authorizes the requested operation. Mobile devices may also perform user authentication by requesting the user to insert passcode or biometric authentication mechanism. The limited operational range of NFC prevents attackers from tricking the vehicle into thinking that your mobile device is nearby when it's not. Examples of these operations includes Unlocking of the doors, Starting the engine, etc.
- **Sharing Digital Keys:** The devices which can use Digital Keys can be Owner device as well as Friend device. There is no limit to the number of friend devices with Digital Keys for a given

vehicle, but friend devices may not share access with other friend devices. An owner device shares a Digital Key with a friend device by sending a sharing link to the friend device (e.g., via SMS). When the Digital Key is accepted (e.g., by tapping the sharing link), the friend device creates a Digital Key with the appropriate parameters (vehicle, entitlements, etc.), the Digital Key framework establishes a secure communications channel between the two devices, through which the owner device signs (approves) the friend device's digital key (public key), and necessary signatures (approvals) are obtained from cloud services (e.g., Vehicle OEM Servers). To ensure that the shared Digital Key is usable only by the intended recipient, the owner may optionally provide them with one or more sharing passwords and/or PINs communicated on a different channel than the sharing link.

- **Termination/Suspension of Digital Keys:** This feature enables the user to terminate their digital key or to suspend it during various situations such as selling of the vehicle, the mobile device being stolen/lost, a security breach on the mobile device, or even when the owner decides not to share the keys anymore with a friend. Digital Keys may be terminated or suspended at any time. Termination is permanent and requires the sharing of a new Digital Key to restore access, while suspension is temporary and simply disables a Digital Key until it is resumed.

2.3.2.2. MAJOR SECURITY FEATURES OF THE TOE

The security features of the TOE are detailed in [CCC-TS-101] and listed below:

- Secure Owner Pairing
- Secure Standard Transaction
- Secure Fast Transaction
- Secure Check Presence Transaction
- Secure DK Sharing
- Key Termination & Suspension
- Secure Applet Management

2.3.3. AVAILABLE NON-TOE HARDWARE/SOFTWARE/FIRMWARE

The Figure 1 depicts the regions including TOE, Non-TOE and the Device region. The region inside red discrete lines (---) shows the TOE region, the one with blue discrete lines (---) shows the Non-TOE region and the region inside ash colored box depicts the Device region.

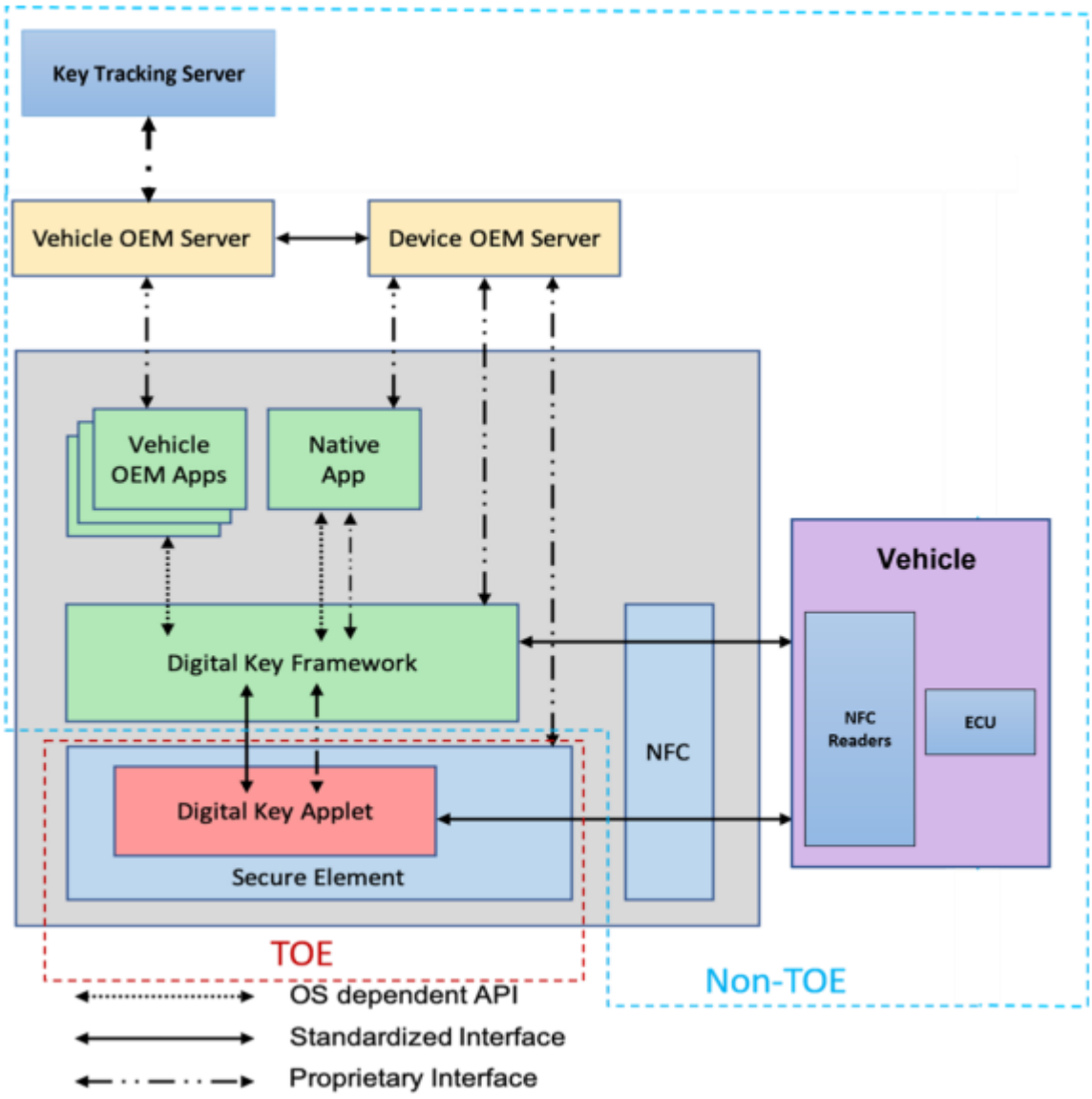

Figure 1: TOE and non-TOE components in the DK system

The non-TOE components depicted in Figure 1 are described here:

Digital Key Framework:

- Implements main features: owner pairing, Digital Key sharing and management.
- Provides common Digital Key service functionality via a set of OS-specific APIs for Vehicle OEM apps.

Vehicle OEM App:

- The Vehicle OEM app is optional. The main features of the app are supported natively by the device.
- May support the same features as the native app plus Vehicle OEM-specific features

Native App:

- Provides device-native UI such as Digital Key creation, Digital Key termination and deletion, Digital Key enable/disable, etc.
- Displays a list of all issued owner/friend Digital Keys.

Device OS

- The Operating system of the underlying device on which the Vehicle OEM Apps, Native Apps and Digital Key Framework reside.

DK Applet EE

- Execution Environment (SE Operating System and underlying platform, or equivalent) for the DK Applet.

Vehicle – ECU

- ECU of the Vehicle, performing the security functions for managing access and starting the Vehicle.

Vehicle

- Determine if the owner/friend device is eligible for the Digital Key service before allowing owner pairing or accepting a friend key shared by the owner device.
- Verify authenticity of the device

Vehicle NFC Readers

- Communicate with the owner device for owner pairing and Digital Key transactions (lock/unlock, engine start, etc.).
- Communicate with the friend device for Digital Key transactions.

Vehicle OEM Server

- Backend for external management of the Vehicles.
- Host owner account that links to the owner's vehicle(s)
- Manage Digital Key service subscriptions.
- Provide necessary attestations to the vehicle (when online) so that shared friend Digital Keys are accepted by the vehicle in the first friend transaction.
- Manage a secure channel to the vehicle.

Key Tracking Server (KTS)

- Record relevant data to be able to assign a tracked Digital Key for a vehicle to a device. The KTS is likely to be managed by the Vehicle OEM

Devices

- Can take on the role of an owner device and friend device.
- Support contactless transactions to lock/unlock vehicle and start the engine.
- Support configurable user authentication (e.g., passcode)

Owner Device

- Implement main features: transaction, owner pairing, Digital Key sharing (sender) and Digital Key termination.
- Store necessary certificates for owner pairing and Digital Key sharing
- Terminate Shared Keys

Friend Device

- Implement main features: transaction, Digital Key sharing (receiver), key termination.
- Store necessary certificates for Digital Key sharing
- Send termination attestation to Vehicle OEM Server

Device OEM Server

- Load and install the Digital Key instance of the Digital Key applet (if necessary)
- Provide and update necessary certificates in the device

2.4. TOE DESCRIPTION

2.4.1. ARCHITECTURE OF mDKA V2.5-15E0

The high-level architecture of mDKA v2.5-15E0 on ST54L A02 can be represented by Figure 2. In this figure, the elements in **blue** are configurable.

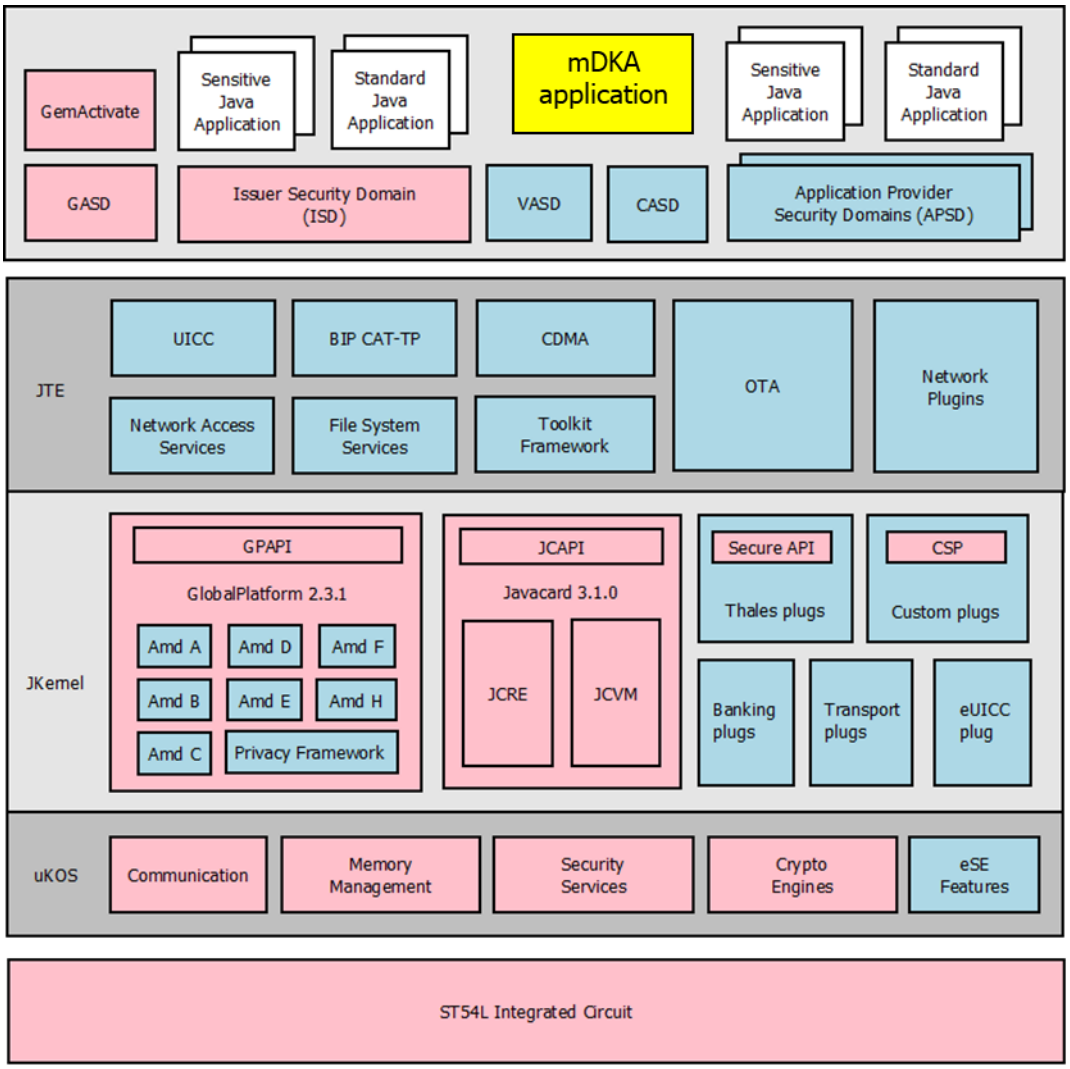

Figure 2 : mDKA v2.5-15E0 architecture

The architecture can be decomposed in three layers:

- The hardware layer composed of the ST54L A02 integrated circuit.
- The Connected eSE 5.3.4 v1.1 platform, which is the operating system of the product.
- The application layer, encompassing:
  - standard and sensitive applications and therefore mDKA v2.5 Application
  - the security domains (ISD, GASD, VASD, CASD and APSDs).

The Connected eSE 5.3.4 v1.1 platform implements two major industry standards:

- Oracle's Java Card 3.1.0 [Javacard], which consists of the Java Card 3.1.0 Virtual Machine, Java Card 3.1.0 Runtime Environment and the Java Card 3.1.0 Application Programming Interface.
- Global Platform 2.3.1 [GP], SE Configuration.

Additionally, other applets – not determined at the moment of the present evaluation – may be loaded before issuance.

2.4.2.TOE BOUNDARIES

2.4.2.1. TOE PHYSICAL BOUNDARIES

The TOE physical boundaries encompass:

- The ST54L A02 IC which is a tamper-proof chip in Wafer Level Chip Scale Package (WLCSP) format, which can be soldered in any device PCB.
- The Thales Connected eSE 5.3.4 v1.1 platform.
- The Thales mDKA v2.5 Application

Any other item is outside the scope of the evaluation.

2.4.2.2. TOE LOGICAL BOUNDARIES

The TOE logical boundaries are delimited (dash line in **red**) in Figure 3.

In this figure, the TSF components have been put in **yellow** color. The other components (in white color) do not participate to the TOE security.

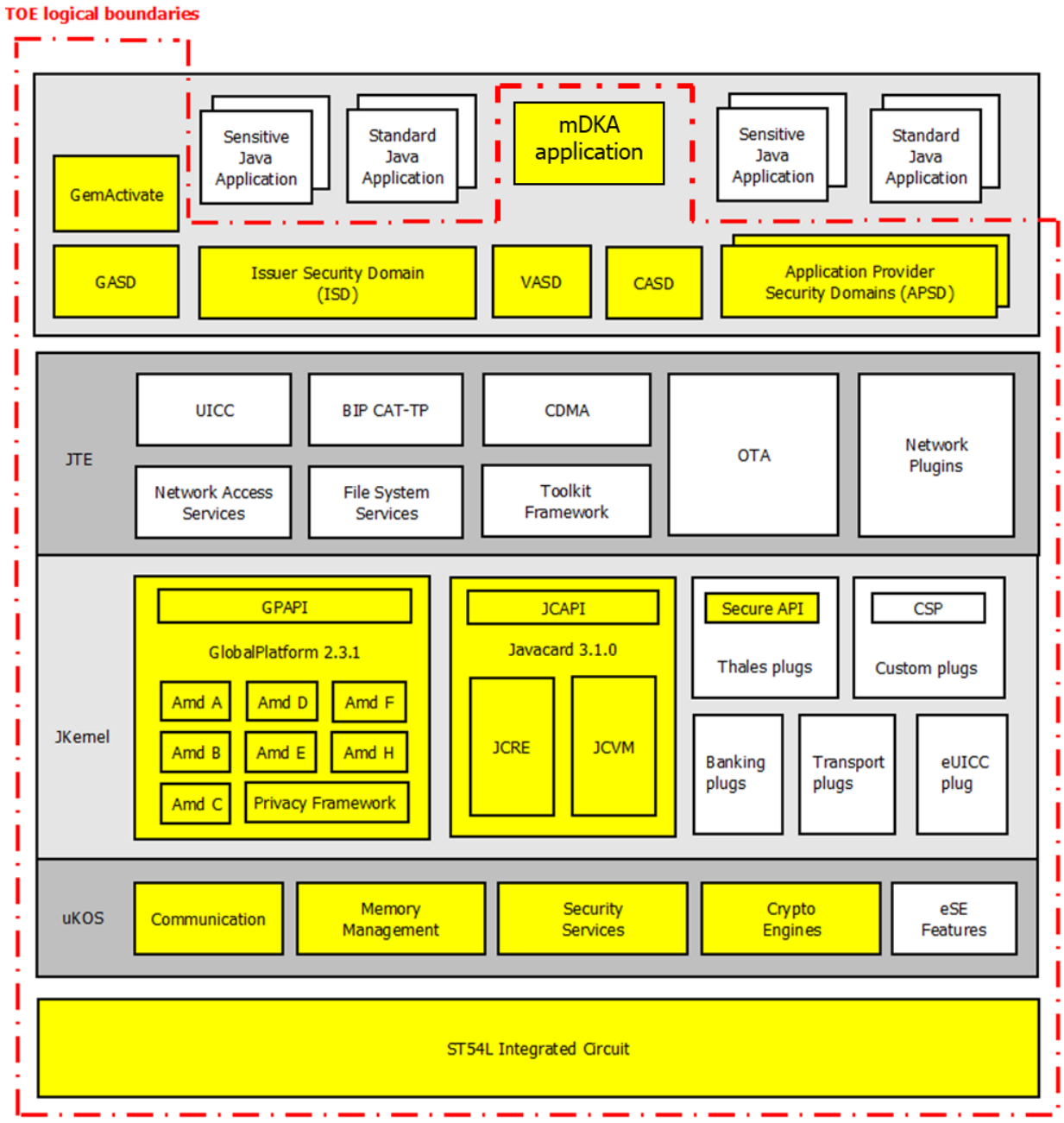

Figure 3 : TOE logical boundaries

2.5. TOE LIFE-CYCLE

The product and TOE life cycle is composed of 7 phases which are described in Table 2. The table also mentions the actor(s) involved in each phase, as well as the associated location(s).

The loading of the Connected eSE 5.3.4 v1.1 software and mDKA v2.5 application occurs during phase 5, after which the IC loading service is locked and no more available. The TOE delivery point - which determines the boundary between the ALC and AGD Common Criteria assurance classes - is put at the end of phase 5, as illustrated in Figure 4.

As described, at the end of phase 6 STMicroelectronics delivers personalized product to the Original Equipment Manufacturer (OEM). At this stage, the TOE is entirely built and protects itself through the security mechanisms implemented in the operating system and the underlying IC.

PHASE	DESIGNATION	DESCRIPTION / COMMENTS		ACTOR	LOCATION
1	Connected eSE 5.3.4 v1.1 software And mDKA v2.5 application development	Connected eSE 5.3.4 v1.1 platform development	Platform development & tests	**Thales** DIS MCS R&D team - secure environment -	**Thales Singapore** site
				Thales SL Crypto team - secure environment -	**Thales Singapore** site
		Patch development	Patch development and tests	**Thales** DIS MCS R&D and SL Crypto teams - secure environment -	**Thales Singapore** site
		mDKA v2.5 application	Applet development & tests	**Thales** DIS MCS R&D team - secure environment -	**Thales Singapore** site
		Basic and secure applets development	Applet development & tests	**Thales or any other accredited Application Provider (AP)** - secure environment -	**Thales Singapore** site or **Application Providers' development sites**
		Industrialization	Production scripts and tools development for phase 5.	**Thales** Product Engineering Team - secure environment -	**Thales Gémenos** site **Thales Singapore** site
			Personalization scripts development for phase 6.	**Thales** CPC team - secure environment -	**Thales Tczew** site
			Personalization Data Generation Secure delivery of Connected eSE 5.3.4 v1.1 platform to STMicroelectronics, together with scripts and personalization data.	**Thales** Data Generation team - secure environment -	**Thales Pont-Audemer** site

2	IC development	Development of the ST54L A02 IC and associated tools.	**STMicroelectronics** - Secure environment -	Development site(s) stated in the ST54L A02 CC certificate
3	IC manufacturing	Manufacturing of virgin ST54L A02 integrated circuits.	**STMicroelectronics** - Secure environment -	Manufacturing site(s) stated in the ST54L A02 CC certificate
4	IC packaging	IC packaging & testing.	**STMicroelectronics** - Secure environment -	Packaging site(s) stated in the ST54L A02 CC certificate
5	Composite Product integration	Loading of the Connected eSE 5.3.4 v1.1software (platform and pre-issuance applications). Pre-personalization and Testing.	**STMicroelectronics** - Secure environment -	Production site(s) stated in the ST54L A02 CC certificate
6	Personalization	Personalization and final tests.	**STMicroelectronics** - Secure environment -	Personalization site(s) stated in the ST54L A02 CC certificate
7	End-usage	End-usage for the Original Equipment Manufacturer (OEM) and accredited business partners (Application Providers). The OEM, who is the issuer of the Connected eSE 5.3.4 v1.1 product, is responsible for the secure element administration during the end-usage phase and the end of life process. The OEM also grants administration privileges to Application Providers on their respective Security Domains (APSD). Applets may be loaded onto the chip, and OS updates may also be triggered at this stage.	**Original Equipment Manufacturer and accredited business partners (Application Providers)**	Field
		End-usage for mobile phone holder The end-user accesses the OEM related services and performs secure transactions with his mobile phone, thanks to the Connected eSE 5.3.4 v1.1 secure element hosting the sensitive applications and related assets.	**Mobile phone holder**	Field

Table 1: Product and TOE life-cycle phases

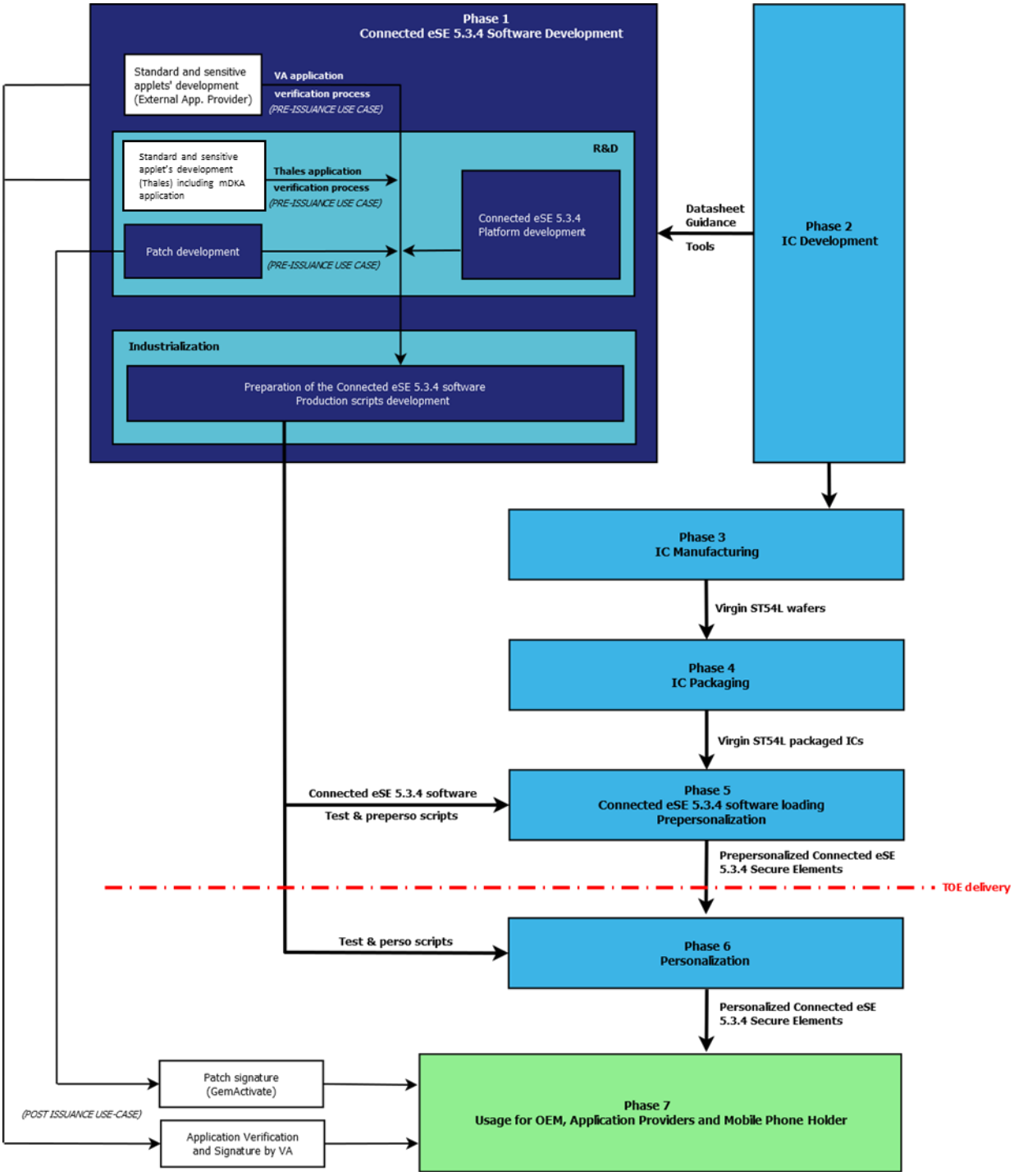

Figure 4: Product and TOE life-cycle

2.5.1.INVOLVED THALES-DIS SITES

- **Development and Project Management**
  - Singapore (Singapore)
    - Platform, Applet, and Patch Development
    - Cryptographic library development
  - La Ciotat (France)
    - Security architecture
    - CCC Project management
  - Meudon (France)
    - CCC Project management

- **Industrialization / Manufacturing**
  - Gémenos (France)
  - Tczew (Poland)
  - Pont-Audemer (France)
  - Singapore

2.5.2.TOE ACTORS

The TOE users (in the CC meaning, i.e. after TOE delivery) are described hereunder:

- ***The Original Equipment Manufacturer (OEM)***, who is the issuer of the Connected eSE 5.3.4 v1.1 secure element and owner of the TOE. The TOE authorizes the OEM, once authenticated, to manage the loading, instantiation or deletion of applications. This is *Device OEM server. Vehicle OEM Server* manage all back-end activities.
- ***The Application Providers (AP)*** are entities or institutions responsible for their applications and associated services. It may be for example a financial institution (a bank) or a transport operator.
- ***The Verification Authority (VA)***, trusted third party represented on the TOE, acts on behalf of the OEM and is responsible for the verification of applications signatures (DAP) during the loading process. These applications shall be validated for the standard applications or certified for the secure ones.
- ***End-user*****,** who holds the TOE in the usage phase (phase 7). The card is personalized with his or her identification and secrets. It could be either device's Owner or device's Friend.

2.5.3.TOE DELIVERY

The TOE is delivered as a whole package with the Platform (Connected eSE 5.3.4 v1.1 platform).It is ciphered by Thales Trust Center and delivered from Thales Data Processing Configuration development site (Tczew) to Thales Manufacturing site (Pont-Audemer) via Thales PDM tool.

It is then securely sent from Thales Manufacturing site to STMicroelectronics using Thales Allynis Connect platform (Thales' secure platform for data transfer with external parties).

STMicroelectronics is in charge of the Connected eSE 5.3.4 v1.1 platform including mDKA v2.5 application loading, pre-personalization and personalization in its own premises and proceeds to the delivery of the product directly to customers.

The different guides accompanying the TOE and parts of the TOE are the ones specified in [AGD]. They are delivered by Thales Technical representative, in form of electronic documents (*.PDF), via secure email (PGP ciphered).

ITEM TYPE	ITEM	REFERENCE/VERSION	FORM OF DELIVERY
Software	mDKA v2.5-15E0	Refer to chapter §2.2	Enciphered TOE via Allynis Connect (Thales secure transmission tool)
Document	[AGD]	Refer to chapter §1.2	Electronic document (PDF) via secure email

3. CONFORMANCE CLAIMS

3.1. COMMON CRITERIA VERSION

This ST conforms to CC Version 3.1 Revision 5 [CC-1][CC-2][CC-3].

3.2. CONFORMANCE TO CC PART 2 AND 3

This ST is CC part 2 extended with the FCS_RNG.1 components. All the other SFRs have been drawn from the catalogue of requirements in CC part 2 [CC-2].

This ST is CC part 3 conformant. It means that all SARs in that ST are based only upon assurance components in CC part 3 [CC-3].

3.3. ASSURANCE PACKAGE CONFORMANCE

This ST conforms to the assurance package EAL4 augmented by ALC_DVS.2, AVA_VAN.5 and ALC_FLR.1.

3.4. EVALUATION TYPE

This is a composite evaluation, which relies on the Connected eSE 5.3.4 v1.1 platform certificate and evaluation results:

- Certification done under the ANSSI scheme
- Certification report ANSSI-CC-2024/33
- Security Target [ST_PF] conformant to GPSE Protection Profile [PP-GP]
- Common criteria version: 3.1 Revision 5
- Assurance level: EAL4+ (ALC_DVS.2, AVA_VAN.5 and ALC_FLR.1 augmentation)

Consequently, the composite product evaluation (i.e. the present evaluation) includes the additional composition tasks defined in the CC supporting document “Composite product evaluation for smart cards and similar devices” [CCDB].

Connected eSE 5.3.4 v1.1 platform evaluation is a composite evaluation, which relies on the ST54L A02 chip certificate and evaluation results:

- Certification done under the NSCIB scheme
- Certification report NSCIB-CC-2300182-01
- Security Target [ST_IC] strictly conformant to IC Protection Profile [PP/0084]
- Common criteria version: 3.1 Revision 5
- Assurance level: EAL6+ (ALC_FLR.2 augmentation)

3.5. PROTECTION PROFILE CONFORMANCE CLAIMS

This Security Target claims strict conformance to Car Connectivity Consortium Protection Profile of the Digital Key Applet [PP-CCC-CP-023].

As strict conformance is required by the [PP-CCC-CP-023], no conformance claim rationale is required.

4. SECURITY PROBLEM DEFINITION

4.1. ASSETS

Assets are entities that the owner of the TOE presumably places value upon. Assets are expected to be directly protected by the TOE.

The following assets are divided in two groups. The first one contains the data created by and for the user (User data) and the second one includes the data created by and for the TOE (TSF data).

ASSETS (USER DATA)	DESCRIPTION	SENSITIVITY (C, I, A, P)[1]
D.OWNER_DATA	Information related to the owner or friend like phone number, location, device usage or other (Personally Identifiable Information) PII on the device side.	C, A
D.KEY_OPTIONS	Options to the keys like access rights, key validity and other fields which are not specified in more detail.	C, I, A
D.MESSAGES_EXCHANGES	Data and commands exchanged between the components of the systems in plain form.	C, I, A

Table 2 User Data Assets, Description and Sensitivity

ASSETS (TSF DATA)	DESCRIPTION	SENSITIVITY (C, I, A, P)[2]
D.KCMAC_KEY	The D.KCMAC_KEY is a derived symmetric key used to calculate cryptograms. It is part of owner DK secret / Friend DK secret. This key is used for secure channel opening during the fast transactions.	C, I, A
D.IMMOTOKEN	Vehicle cryptographic material that is provisioned by some vehicles (confidential mailbox) at the DK creation and that might be requested back during the fast or standard transaction to allow engine start.	C, I, A
D.SECRET_SHARED_KEY	A shared symmetric key generated on both the vehicle and the device sides during owner pairing (standard transaction) using key agreement method (Kdh). Kdh is a shared key computed using Diffie-Hellman according to [BSI TR-03111] Section 4.3 indications.	C, I, A
D.GP_CODE	The code of the GlobalPlatform framework on the secure element.	I, A

[1] C = Confidentiality, I = Integrity, P = Privacy, A = Availability
[2] C = Confidentiality, I = Integrity, P = Privacy, A = Availability

ASSETS (TSF DATA)	DESCRIPTION	SENSITIVITY (C, I, A, P)[2]
D.SE_MNGT_DATA	The data of the secure element management environment, like for instance, the identifiers, the privileges, life cycle states, the memory resource quotas of applets and security domains.	I, A
D.DK_APPLET_CODE	The source code of the DK Applet.	I, A
D.LONG_TERM_KEY	Symmetric long-term key that is used to derive encryption and MAC session keys. It is stored in NVM on both vehicle and device sides.	C, I, A
D.APPLET_ROOT_KEY	The root of trust of the SE storage that is used to bind the DK Applet and keys to the SE. (SE_root_SK/PK).	C, I, A
D.SESSION_KEYS	Temporary key material used to protect data in the DK communication protocol. This includes Kenc, Kmac and Krmac.	C, I, A
D.SEC_ATTRIBUTES	The runtime security data including all identifiers, context of execution.	C, I, A
D.OWNER_DK_SECRET	General term for Owner DK secret and/or Friend DK secret and/or IMMOTOKEN. **Application Note 1:** • **Public keys are mutually exchanged through pairing of the owner device to the vehicle. The owner can then authorize the use of Digital Keys by friends and family members, by signing their public keys. DK secret corresponds to the private key associated to these public keys. KCMAC is a symmetric key derived from the symmetric long-term key according to [RFC 5869] Section 2.** • **There is one Digital Key per vehicle. During owner pairing, all Digital Key elements are provided by the vehicle and transferred to the device.**	C, I, A
D.OWNER_DK_DATA	Information attached to the digital key concept except the DK secret. E.g. mailbox data, public DK key	I, A
D.DK_API_DATA	Data of the DK Applet API, such as the contents of its private fields.	C, I, A
D.RNG	Generated random numbers **Application Note 2: In addition to the confidentiality and integrity properties, unpredictability, sufficient entropy, and forward secrecy are to be considered for this asset.**	C, I, A

Table 3 : TSF Data Assets, Description and Sensitivity

The following table is intended to highlight where each cryptographic key could be potentially stored.

CRYPTOGRAPHIC KEYS	STORAGE

CRYPTOGRAPHIC KEYS	STORAGE
D.KCMAC_KEY	DK Applet within SE & Vehicle-ECU module
D.IMMOTOKEN	DK Applet within SE & Vehicle-ECU module
D.SECRET_SHARED_KEY	DK Applet within SE & Vehicle-ECU module
D.LONG_TERM_KEY	DK Applet within SE & Vehicle-ECU module
D.APPLET_ROOT_KEY	SE
D.SESSION_KEYS	DK Applet within SE & Vehicle-ECU module
D.OWNER_DK_SECRET	DK Applet within SE

Table 4 : Storage of cryptographic keys

4.2. THREATS

This section introduces the threats to the assets against which specific protection within the TOE or its environment is required. The threats that are specific to the Operational Environment are mapped to respective OE.Security Objectives inside Table 10. Several groups of threats are distinguished according to the means used in the attack. The classification is also inspired by the components of the TOE that are supposed to counter each threat.

THREATS	DESCRIPTION	COVERED ASSETS
T.DK_PHYSICAL	An attacker, with physical access to the TOE, may attempt to access the DK sensitive assets when it is stored. These physical access threats may involve attacks, which attempt to access the device through external hardware ports, impersonate the user authentication mechanisms, through its user interface, and also through direct and possibly destructive access to its storage media.	All
T.UNAUTHORIZED_SE_MNG	The attacker performs unauthorized secure element management operations (for instance impersonates one of the actors represented on the secure element) in order to take benefit of the privileges or services granted to this actor on the secure element such as fraudulent: • load of a package file installation of a	D.SE_MNGT_DATA D.DK_APPLET_CODE

THREATS	DESCRIPTION	COVERED ASSETS
	package file • extradition of a package file or an applet • personalization of an applet or a Security Domain • deletion of a package file or an applet • privileges update of an applet or a Security Domain Directly threatened	
T.LIFE_CYCLE	An attacker accesses to an application outside of its expected availability range thus violating irreversible life cycle phases of the application (for instance, an attacker repersonalizes the application).	All TSF data
T.IT_DISCLOSURE	Attacker get unauthorized access to the Immobilizer Token during storage, deletion or processing	D.IMMOTOKEN
T.DK_DISCLOSURE	Attacker is predicting (lack of randomness) or recovering the Digital Key from a Device in order to provision it on his own device and enter the paired Vehicle. The attacker could target any event in the key lifetime (creation/use/access/storage) and particularly events that require the secret material to be transferred from one memory location to another.	All TSF data
T.FLAW_SW	Attacker loads a malicious or exploitable code into the software of a component of the DK ecosystem in order to change the behaviour of the DK feature, to attempt to exfiltrate restricted data, or to gain additional privilege into	D.DK_APPLET_CODE D.APPLET_ROOT_KEY

THREATS	DESCRIPTION	COVERED ASSETS
	the system of the component. For instance, a malicious DK Sharing request is produced to the attacker's device.	
T.NON_REVOKED	An attacker could benefit of preventing the processing of a revocation request in order to keep using a provisioned DK (on a stolen device for instance) to access and steal a Vehicle. For instance, revocation would be issued in order to prevent a component (unit or family or maker or SW version…) from participating in the DK ecosystem when it has shown vulnerabilities. If the other components are not verifying the revocation status of presented certificates, then it would be possible for an attacker to use the vulnerable component to perform its attacks.	D.GP_CODE D.SE_MNGT_DATA D.DK_APPLET_CODE D.LONG_TERM_KEY D.APPLET_ROOT_KEY D.SEC_ATTRIBUTES D.DK_API_DATA
T.DATA_BREACH	Data breach could also take place during the execution of a transaction using NFC communication. The attacker might be able to obtain confidential data as well as shared keys.	D.OWNER_DK_SECRET D.OWNER_DK_DATA D.OWNER_DATA
T.KTS_DATA_LEAK	Unnecessary confidential information of the device's user is sent to the Key Tracking Server, leaking the Vehicle owner’s confidential data.	D.OWNER_DK_DATA D.OWNER_DATA
T.RETRIEVE_SECRET_SHARED-KEY_DKA	Attackers retrieve the previous Secret shared Key generated (Standard or Fast transaction) **Application Note 3 :** • **Dump NVM SE memory (Physical**	D.SECRET_SHARED_KEY

THREATS	DESCRIPTION	COVERED ASSETS
	attacks, Logical attacks (Malware) or Combine attacks) • Exploit chip power consumption or electromagnetic radiation leakages (Side channel attacks) • Flipping a bit allowing to bypass a security mechanism and providing access to memory (Fault injection attacks) If the attack succeeds, then the attacker can open a secure channel during fast transactions with the owner/friend Device to retrieve the immo token and data to lock/unlock. Next the attacker can open new secure channel to lock/unlock the Vehicle.	
T.RETRIEVE_KCMAC_KEY_DKA	Attackers retrieve Kcmac: Application Note 4 : • Dump the NVM SE memory If attack succeed, then the attacker can open secure channel during fast transactions with the owner/friend Device to retrieve the immo token and data to lock/unlock. Next the attacker can open new secure channel to lock/unlock the Vehicle.	D.KCMAC_KEY
T.RETRIEVE_SESSION_KEYS_DKA	Attackers retrieve Kenc / Kmac / Krmac: Application Note 5: • Dump the NVM SE memory An attacker would need to perform the attack for each session during transactions (standard or fast). Further attacks may be needed to lock/unlock	D.SESSION_KEYS

THREATS	DESCRIPTION	COVERED ASSETS
	the Vehicle.	
T.UPDATE_KEY_OPTIONS	Attackers modify the keys options to give all access to lock/unlock and Engine Start	D.KEY_OPTIONS
.UNAUTHORIZED_ACCESS _DK_ASSET	Attackers access crypto primitives using DK applet assets (secret shared key, …) Application Note 6: • Through relay attacks via rogue DK applet in the Device.	D.KCMAC_KEY D.IMMOTOKEN D.SECRET_SHARED_KE Y D.LONG_TERM_KEY D.APPLET_ROOT_KEY D.SESSION_KEYS
T.DEVICE_THEFT	An attacker may attempt to steal a user's device and use it to access or start the vehicle.	All
T.CA_KEY_LEAK	An attacker may attempt to steal the private key used by the device or vehicle root key. An attacker could use this key to generate fraudulent attestations.	D.APPLET_ROOT_KEY
T.RADIO_SNIFF	An attacker may attempt to sniff the traffic between a device and a vehicle during an exchange.	D.SESSION_KEYS D.SECRET_SHARED_KE Y D.SEC_ATTRIBUTES D.OWNER_DK_SECRET D.OWNER_DK_DATA D.OWNER_DATA D.DK_API_DATA D.MESSAGES_EXCHANG ES
T.RADIO_MITM	An attacker may attempt to gain a MITM presence between a device and a vehicle during an exchange and might be able to modify the keys being shared.	D.SESSION_KEYS D.KCMAC_KEY D.IMMOTOKEN D.SECRET_SHARED_KE Y
T.PROTOCOL_DOWNGRADE	An attacker may attempt to downgrade the protocol to an older version that has known weaknesses.	D.KCMAC_KEY D.IMMOTOKEN D.SECRET_SHARED_KE Y D.SE_MNGT_DATA D.SESSION_KEYS
T.SIGN_COMPROMISE_VERI FY_COMPROMISE	Attacker manipulates creation and validation of electronic signatures	D.IMMOTOKEN D.DK_APPLET_CODE D.APPLET_ROOT_KEY

THREATS	DESCRIPTION	COVERED ASSETS
		D.SESSION_KEYS D.OWNER_DK_SECRET
T.DENIAL_OF_LEGITIMATE_ DELETIONS	An attacker prevents a legitimate DK deletion request from the user or a backend system.	D.DK_API_DATA
T.DK_SK_MODIFICATIONS	An attacker modifies DK secret keys in memory.	D.KCMAC_KEY D.IMMOTOKEN D.SECRET_SHARED_KE Y D.DK_APPLET_CODE D.LONG_TERM_KEY D.APPLET_ROOT_KEY
T.RADIO_RELAY_TRANSAC TION	An attacker may try to relay a transaction with radio equipment.	D.SESSION_KEYS D.SEC_ATTRIBUTES D.RNG
T.INTERNET_CONNECTIVITY _DOS	A device OEM may attempt to prevent or otherwise limit the use of a DK from a competing device OEM	D.SEC_ATTRIBUTES D.OWNER_DATA D.DK_API_DATA D.KEY_OPTIONS
T.TIME_CHANGE	An attacker may attempt to change the time on the device or vehicle in order to enable a currently invalid key or disable a currently valid key.	D.KEY_OPTIONS
T.REPLAY_TRANSACTION	An attacker may try to replay an observed NFC transaction to the vehicle	D.SESSION_KEYS D.SEC_ATTRIBUTES D.RNG
T.UNAUTHORIZED_KEY_SH ARING	Two or more collaborating adversaries share (copy) access credentials without the vehicle owner's consent or knowledge. E.g. • resale of car access credentials (e.g. rental/fleet car) • {regulatory, user} ban evasion • use of uncertified applications and/or devices	D.KCMAC_KEY D.IMMOTOKEN D.SECRET_SHARED_KE Y D.LONG_TERM_KEY D.APPLET_ROOT_KEY D.OWNER_DK_SECRET
T.INSTANCE_CA_DISCLOSU RE	An attacker is extracting the Instance CA from the secure storage of a Device or signs its own Instance CA with a	D.OWNER_DK_DATA D.KEY_OPTIONS

THREATS	DESCRIPTION	COVERED ASSETS
	valid signature in order to provision digital keys which are not located in a certified DK Applet / DK Applet EE. DKs created by this attacker may not be secure and prone to various attacks.	

Table 5 Threats, Description and Covered Assets

4.3. ORGANISATIONAL SECURITY POLICIES

This section describes the organizational security policies to be enforced with respect to the TOE environment. Rules to which both the TOE and its human environment SHALL comply when addressing security needs related to the DK Applet.

ORGANIZATIONAL SECURITY POLICIES	DESCRIPTION
OSP.APPS_VALIDATION	The applications SHALL be associated with a digital signature and it SHALL be validated by a validation authority before loading it into the TOE.
OSP.OEM_SERVERS	A security policy SHALL be defined in order to ensure the security of the applications being stored on the OEM servers These policies can include access control policy, regular verification of integrity & encryption, isolation, etc. Site inspections SHALL also take place in order to ensure that the policies have been enforced as per the definitions inside the server security guidance documents.
OSP.KTS_SERVER	Policies SHALL be implemented for the data handled by the KTS server in order to ensure that unnecessary confidential data of the user may not be shared to the KTS server preventing data leakage.
OSP.OS_DOWNGRADE	A policy SHALL be put in place explaining the OS version with which the DK applet is compatible with and also ensure that the downgraded version of the OS is not in use.
OSP.SECURE_KEY_RETRIEVE	The implemented policies SHALL ensure that the key retrieval takes place in a secure manner (secure channel) leaving the attacker from accessing the keys during a transaction.
OSP.KEY_SHARE	A policy SHALL be enforced in the servers ensuring that key sharing and distribution takes place in a secure manner.

ORGANIZATIONAL SECURITY POLICIES	DESCRIPTION
OSP.KEY_OPTIONS	The key-options SHALL be secured against unauthorised modifications/access.
OSP.SERVER_COMMUNICATION	The communication channels established between the servers SHALL be secure. **Application Note 7:** • **Sensitive data elements, where applicable, SHALL be protected with additional encryption protocols.** • **Server APIs SHALL be supported only over https with mutual authentication, i.e., 2-Way TLS.**
OSP.PROTOCOL_FAILURE	A policy SHALL be defined in order to notify in case of a protocol failure and ensure continuity of working.
OSP.DOS_DETECT	A mechanism SHALL be implemented in order to detect the DOS attacks.
OSP.CERTIFICATE	The confidentiality & integrity of the certificate is protected & verified before installation/usage.
OSP.PKI_POLICY	A PKI policy SHALL be implemented which covers the secure management of PKI signature keys and secure operation of the PKI instances.

Table 6 Organizational Security Policies description

4.4. ASSUMPTIONS

The following assumption concerns the product operational environment, after the TOE delivery.

ASSUMPTIONS	DESCRIPTION
A.USER_AUTHENTICATION	The device provides robust user authentication mechanisms to identify the DK user for performing any authorized actions such as deletion of keys, keys sharing etc.
A.USER_PRIVACY_CONSENT	Privacy consent SHALL be asked to the user before sharing any private data of the user to the server/other devices.
A.CERTIFICATION_REVOCATION_SET	Upon request of revocation of any certificate part of a digital key certificate chain (vehicle side or device side), the ecosystem/Certificate authority informs the relevant parties

ASSUMPTIONS	DESCRIPTION
	concerning the revocation.
A.OEM_ADMIN	Administrators of the OEM server are trusted people. They have been well trained to use and administrate the server securely. They are well aware of the sensitivity of the assets the server deals with and also the responsibilities they have to carry out.
A.PRODUCTION_ENV	The production environment SHALL be trusted and secure (prevents attacks from internal attackers).
A.DEVICE_OEM	The Device OEM is a trusted actor who has full control on the content of the SE. It is the responsibility of the Device OEM to ensure that the DK Applet that is deployed has been certified following the CCC Certification Program.
A.OS_CLOCK	The software uses a reliable clock for the proper functioning of the clock.
A.CAR_LOCATION	A locked device never provides information on vehicle location, which it can access.
A.DEVICE_OEM_INSIDER	It is assumed that an insider with access to the device OEM server will not make security changes to the Digital Key content or configuration (for example: may attempt to steal an owner's key or issue new keys)
A.SHARING_MASQUERADE	It is assumed that an attacker will not masquerade as an owner's friend using social engineering or other means and causes the owner to unwittingly share a key with the attacker. Attackers may not also masquerade as owners to get friends to reveal their identity to an attacker.
A.DEVICE_SAFETY	The device is assumed to be protected by the owner from getting stolen as this could lead to unauthorised access to the keys or vehicle by an attacker.
A.REPLAY	It is assumed that the device is protected against replay attacks within the communication protocol such as on NFC signals.
A.RADIO_RELAY	It is assumed that the device ensures protection against relay a transaction with radio equipment attack.
A.OEM_SERVER_SECURITY	It is assumed that the Device OEM Server and

ASSUMPTIONS	DESCRIPTION
	the Vehicle OEM Server are hosted in secure data centers.
A.VEHICLE_ROOT_KEY	The vehicle root key is protected by security measures in the operational environment that ensure its confidentiality

Table 7 Assumptions description

4.5. COMPOSITION TASKS – SECURITY PROBLEM DEFINITION PART

4.5.1. STATEMENT OF COMPATIBILITY – THREATS PART

The following table lists the relevant threats of the security target [ST_PF] and provides the link to the threats on the composite-product, showing that there is no contradiction between the two.

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
From core part		
T.UNAUTHORISED-CARD-MGMT	Threat agent: Attacker Adverse action: 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.PIN, and D.GP_REGISTRY (any other asset may be jeopardised should this attack succeed, depending on the virulence of the installed application).	T.UNAUTHORIZED_SE_MNG
T.LIFE-CYCLE	Threat agent: Attacker Adverse action: An attacker accesses 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.	T.LIFE_CYCLE
T.COM-EXPLOIT	Threat agent: Attacker Adverse action: An attacker remotely exploits the communication channels	T.RADIO_SNIFF T.DATA_BREACH

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	established between a third party and the TOE in order to modify or disclose confidential data. Directly threatened asset(s): All assets are threatened.	
T.BRUTE-FORCE-SCP	Threat agent: Attacker Adverse action: APDU commands/API methods can be repeatedly transmitted/invoked to search the entire space of secret values such as cryptographic keys and attempt their brute force extraction. Directly threatened asset(s): All assets are threatened.	T.DK_PHYSICAL
From package 'Ciphered Load File Data Block (CLFDB)'		
T.CLFDB-DISC	Threat agent: Attacker Adverse action: The attacker discloses a Ciphered Load File Data Block when it is transmitted to the SE for decryption prior to installation. Directly threatened asset(s): All assets are threatened. Note: This threat refines T.COM-EXPLOIT to address the CLFDB.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
From package 'Cardholder Verification Method (CVM)'		
T.CVM-IMPERSONATE	Threat agent: Attacker Adverse action: An attacker could try to impersonate the Cardholder for disclosing or guessing the PIN stored in the CVM, in order to access the services the SE offers. Directly threatened asset(s): D.CVM_PIN	T.DK_PHYSICAL
T.CVM-UPDATE	Threat agent: Attacker Adverse action: An attacker could try executing an application that tries to modify (reset/update) the CVM management data (Retry Limit, retry Counter, CVM value and state). Directly threatened asset(s): D.CVM_MGMT_STATE	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.BRUTE-FORCE-CVM	Threat agent: Attacker Adverse action: APDU commands/API methods could be repeatedly transmitted/invoked to attempt the brute force extraction of secrets such as PINs. Directly threatened asset(s):	T.DK_PHYSICAL

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	D.CVM_PIN, D.CVM_MGMT_STATE	
T.RECEIPT	Threat agent: Attacker Adverse action: The attacker may generate fake receipts in order to hide or falsify completion proofs of card management operations. Directly threatened asset(s): D.RECEIPT-GENERATION-KEY, D.CONFIRMATION-DATA	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.TOKEN	Threat agent: Attacker Adverse action: The attacker may try to impersonate the Card Manager in order to gain access to the card and perform illegitimate card management operations. Directly threatened asset(s): D.TOKEN-VERIFICATION-KEY	T.DK_PHYSICAL
From PP-Module 'Amendment C: Contactless Services (CTL)'		
T.CTL-REGISTRY-OVERWRITE	Threat agent: Attacker Adverse action: The attacker attempts to modify the contents of the Contactless Registry in order to: - Set an application in an unauthorised state (e.g. ACTIVATE a NON_ACTIVATABLE application) - Reset the counter Directly threatened asset(s): D.CTL_REGISTRY, D.CTL_PRO	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.COUNTERS-FREEZE	Threat agent: Attacker Adverse action: The attacker attempts to prevent the counter increment in order to have an operation performed twice as the off-card entity believes no transition has taken place. Directly threatened asset(s): D.CTL_REGISTRY, D.CTL_PRO	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.CTL-AUTH-FORGE	Threat agent: Attacker Adverse action: The attacker attempts to use the STORE DATA command in order to modify the blacklist of tokens and reuse a blacklisted CCM token. The attacker may also use this command to make CRS visible on the CTL interface whereas CRS personalisation is not complete, in	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	order to perform unauthorised transactions. Directly threatened asset(s): D.CTL_REGISTRY, D.CTL_PRO	
T.CRS-BYPASS	Threat agent: Attacker Adverse action: The attacker grants the CRS privileges to an unauthorized application in order to perform unauthorised state transitions (e.g. set a NON-ACTIVATABLE application to ACTIVATED or DEACTIVATED, or make it visible). Directly threatened asset(s): D.CTL_REGISTRY, D.CTL_PRO	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
From PP-Module 'Amendment H: Executable Load File Upgrade (ELFU)'		
T.ELF-UNAUTHORISED	Threat agent: Attacker Adverse action: An attacker tries to load an ELF without authorisation. Directly threatened asset(s): T D.OLD_ELF, D.NEW_ELF, D.ELF_AID	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.ELF-VERSION	Threat agent: Attacker Adverse action: An attacker tries to modify the application version in order to prevent the loading of a new ELF. Directly threatened asset(s): T D.OLD_ELF, D.NEW_ELF, D.ELF_AID	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.ELF-DATA-ACCESS	Threat agent: Attacker Adverse action: An attacker tries to access confidential application instance data. Directly threatened asset(s): D.ELF_APP_INS	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.ELF-DATA-INTEGRITY	Threat agent: Attacker Adverse action: An attacker tries to change application instance data. Directly threatened asset(s): D.ELF_APP_INS	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.ELF-SESSION	Threat agent: Attacker Adverse action: An attacker tries to perturb the Session Status to recognize an incomplete upgrade as being complete. Directly threatened asset(s): D.ELF_SESSION_ST	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.ELF-ILL-COMMAND	Threat agent: Attacker	Analysis of the composite-

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	Adverse action: An attacker tries to execute forbidden commands during the ELF upgrade session. Directly threatened asset(s): All ELFU PP-Module assets are threatened.	product threats does not reveal any contradiction with this PF threat.
T.ELF-RES-DATA	Threat agent: Attacker Adverse action: An attacker tries to reallocate TOE resources from a user or process to another for gaining unauthorised access to ELF data. Directly threatened asset(s): All ELFU PP-Module assets are threatened.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
From PP-Module 'OS Update'		
T.UNAUTHORISED-TOE-CODE-UPDATE	Threat agent: Attacker Adverse action: An attacker loads malicious additional code in order to compromise the security features of the TOE. Directly threatened asset(s): D.OS-UPDATE_ADDITIONALCODE, D.JCS_CODE, D.JCS_DATA.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.FAKE-SGNVER-KEY	Threat agent: Attacker Adverse action: An attacker modifies the signature verification key used by the TOE to verify the signature of the additional code. Hence, the attacker is able to sign and successfully load malicious additional code inside the TOE. Directly threatened asset(s): D.OS-UPDATE_SGNVER-KEY, D.OS UPDATE_ADDITIONALCODE.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.WRONG-UPDATE-STATE	Threat agent: Attacker Adverse action: An attacker prevents the OS Update operation to be performed atomically, resulting in an inconsistency between the resulting TOE code and the identification data: ▪ The additional code is not loaded within the TOE, but the identification data is updated to mention that the additional code is present. ▪ The additional code is loaded within the TOE, but the identification data	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	is not updated to indicate the change. Directly threatened asset(s): D.OS-UPDATE-CODE-ID.	
T.INTEG-OS-UPDATE-LOAD	Threat agent: Attacker Adverse action: The attacker modifies (part of) the additional code when it is transmitted to the TOE for installation. Directly threatened asset(s): D.OS-UPDATE_ADDITIONALCODE, D.JCS_CODE, D.JCS_DATA.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.CONFID-OS-UPDATE-LOAD	Threat agent: Attacker Adverse action: The attacker discloses (part of) the additional code when it is transmitted to the TOE for installation. Directly threatened asset(s): D.OS-UPDATE_ADDITIONALCODE, D.JCS_CODE, D.JCS_DATA.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
From [PP-JCS]		
#.CONFID-APPLI-DATA	Application *data must be protected against unauthorized disclosure. This concerns logical attacks at runtime in order to gain read access to other application's data.*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
#.CONFID-JCS-CODE	*Java Card System code must be protected against unauthorized disclosure. Knowledge of the Java Card System code may allow bypassing the* TSF. *This concerns logical attacks at runtime in order to gain a read access to executable code, typically by executing an application that tries to read the memory area where a piece of Java Card System code is stored.*	T.FLAW_SW
#.CONFID-JCS-DATA	*Java Card System data must be protected* against *unauthorized disclosure. This concerns logical attacks at runtime in order to gain a read access to Java Card System data. Java Card System data includes the data managed by the Java Card RE, the Java Card VM and the internal data of Java Card platform API classes as well.*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
#.INTEG-APPLI-	*Application code must be protected*	Analysis of the composite-

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
CODE	*against unauthorized modification. This* concerns *logical attacks at runtime in order to gain write access to the memory zone where executable code is stored. In post-issuance application loading, this threat also concerns the modification of application code in transit to the card.*	product threats does not reveal any contradiction with this PF threat.
#.INTEG-APPLI-DATA	*Application data must be protected against unauthorized modification. This concerns logical attacks at runtime in order to gain unauthorized write access to application data. In post-issuance application loading, this threat also concerns the modification of application data contained in a CAP file in transit to the card. For instance, a CAP file contains the values to be used for initializing the static fields of the CAP file.*	T .DK_SK_MODIFICATIONS
#.INTEG-JCS-CODE	*Java Card System code must be protected against unauthorized* modification*. This concerns logical attacks at runtime in order to gain write access to executable code.*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
#.INTEG-JCS-DATA	*Java Card System data must be protected against unauthorized modification. This concerns logical attacks at runtime in order to gain write access to Java Card System data. Java Card* System *data includes the data managed by the Java Card RE, the Java Card VM and the internal data of Java Card API classes as well.*	T.INSTANCE_CA_DISCLOSURE T.DK_DISCLOSURE T.IT_DISCLOSURE
#.EXE-APPLI-CODE	*Application (byte)code must be protected against unauthorized execution.* This *concerns (1) invoking a method outside the scope of the accessibility rules provided by the access modifiers of the Java programming language ([JAVASPEC], §6.6); (2) jumping inside a method fragment or interpreting the contents of a data memory area as if it was executable code; (3) unauthorized execution of a remote method from the CAD (if the TOE provides JCRMI*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	functionality).	
#.EXE-JCS-CODE	*Java Card System bytecode must be* protected *against unauthorized execution. Java Card System bytecode includes any code of the Java Card RE or API. This concerns (1) invoking a method outside the scope of the accessibility rules provided by the access modifiers of the Java programming language ([JAVASPEC], §6.6); (2) jumping inside a method fragment or interpreting the contents of a data memory area as if it was executable code. Note that execute access to native code of the Java Card System and applications is the concern of #.NATIVE.*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
#.FIREWALL	*The Firewall shall ensure controlled sharing of class instances, and isolation of their data and code between CAP files (that is, controlled execution contexts) as well as between CAP files and the JCRE context. An applet shall not read, write, compare a piece of data belonging to an applet that is not in the same context, or execute one of the methods of an applet in another context without its authorization.*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
#.NATIVE	*Because the execution of native code is outside of the JCS TSF scope, it must be secured so as to not provide ways to bypass the TSFs of the JCS. Loading of native code, which is as well outside those TSFs, is submitted to the same requirements. Should native software be privileged in this respect, exceptions to the policies must include a rationale for the new security framework they introduce.*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
#.VERIFICATION	*Bytecode must be verified prior to being executed.* Bytecode *verification includes (1) how well-formed CAP file is and the verification of the typing constraints on the bytecode, (2) binary compatibility with installed CAP files and the assurance that the export*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	files used to check the CAP file correspond to those that will be present on the card when loading occurs.	
#.INSTALL	*(1) The TOE must be able to return to a safe and consistent state when the installation of a CAP file or an applet fails or be cancelled (whatever the reasons). (2) Installing an applet must have no effect on the code and data of already installed applets. The installation procedure should not be used to bypass the TSFs. In short, it is an atomic operation, free of harmful effects on the state of the other applets. (3) The procedure of loading and installing a CAP file shall ensure its integrity and authenticity. In case of Extended CAP files, installation of a CAP shall ensure installation of all the packages in the CAP file.*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
#.SID	*(1) Users and subjects of the TOE must be identified. (2) The identity of sensitive users and subjects associated with* administrative *and privileged roles must be particularly protected; this concerns the Java Card RE, the applets registered on the card, and especially the default applet and the currently selected applet (and all other active applets in Java Card System 2.2.x). A change of identity, especially standing for an administrative role (like an applet impersonating the Java Card RE), is a severe violation of the Security Functional Requirements (SFR). Selection controls the access to any data exchange between the TOE and the CAD and therefore, must be protected as well. The loading of a CAP file or any exchange of data through the APDU buffer (which can be accessed by any applet) can lead to disclosure of keys, application code or data, and so on.*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
#.OBJ-DELETION	*(1) Deallocation of objects should not introduce* security *holes in the form of*	Analysis of the composite-product threats does not reveal

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	references pointing to memory zones that are not longer in use, or have been reused for other purposes. Deletion of collection of objects should not be maliciously used to circumvent the TSFs. (2) Erasure, if deemed successful, shall ensure that the deleted class instance is no longer accessible.	any contradiction with this PF threat.
#.DELETION	*(1) Deletion of installed applets (or CAP files) should not introduce security holes in the form of broken references to garbage collected code or data, nor should they alter integrity or confidentiality of remaining applets. The deletion procedure should not be maliciously used to bypass the TSFs. (2) Erasure, if deemed successful, shall ensure that any data owned by the deleted applet is no longer accessible (shared objects shall either prevent deletion or be made inaccessible). A deleted applet cannot be selected or receive APDU commands. CAP file deletion shall make the code of the CAP file is no longer available for execution. In case of Extended CAP files, deletion of a CAP shall ensure that code and data for all the packages in the CAP file is no longer available for execution. (3) Power failure or other failures during the process shall be taken into account in the implementation so as to preserve the SFRs. This does not mandate, however, the process to be atomic. For instance, an interrupted deletion may result in the loss of user data, as long as it does not violate the SFRs.*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
#.RESOURCES	*The TOE controls the availability of resources for the applications in order to prevent unauthorized denial of service or malfunction of the TSFs. This concerns both execution (dynamic memory allocation) and installation (static memory allocation) of applications and CAP files.*	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
#.INTEG-APPLI-DATA-PHYS	*Integrity-sensitive application data must be* protected *against unauthorized modification by physical attacks.*	T.DK_PHYSICAL
From derived by the here-above security aspects		
T.CONFID-APPLI-DATA	The attacker executes an application to disclose data belonging to another application. See #.CONFID-APPLI-DATA for details. Directly threatened asset(s): D.APP_C_DATA, D.PIN and D.APP_KEYs.	T.FLAW_SW
T.CONFID-JCS-CODE	The attacker executes an application to disclose the Java Card System code. See #.CONFID-JCS-CODE for details. Directly threatened asset(s): D.JCS_CODE.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.CONFID-JCS-DATA	The attacker executes an application to disclose data belonging to the Java Card System. See #.CONFID-JCS-DATA for details. Directly threatened asset(s): D.API_DATA, D.SEC_DATA, D.JCS_DATA and D.CRYPTO.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.INTEG-APPLI-CODE	The attacker executes an application to alter (part of) its own code or another application's code. See #.INTEG-APPLI-CODE for details. Directly threatened asset(s): D.APP_CODE.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.INTEG-APPLI-CODE.LOAD	The attacker modifies (part of) its own or another application code when an application CAP file is transmitted to the card for installation. See #.INTEG-APPLI-CODE for details. Directly threatened asset(s): D.APP_CODE.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.INTEG-APPLI-DATA	The attacker executes an application to alter (part of) another application's data. See #.INTEG-APPLI-DATA for details. Directly threatened asset(s): D.APP_I_DATA, D.PIN, and D.APP_KEYs.	T.FLAW_SW
T.INTEG-APPLI-DATA.LOAD	The attacker modifies (part of) the initialization data contained in an application CAP file when the CAP file	Analysis of the composite-product threats does not reveal any contradiction with this PF

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	is transmitted to the card for installation. See #.INTEG-APPLI-DATA for details. Directly threatened asset(s): D.APP_I_DATA and D_APP_KEY.	threat.
T.INTEG-JCS-CODE	The attacker executes an application to alter (part of) the Java Card System code. See #.INTEG-JCS-CODE for details. Directly threatened asset(s): D.JCS_CODE.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.INTEG-JCS-DATA	The attacker executes an application to alter (part of) Java Card System or API data. See #.INTEG-JCS-DATA for details. Directly threatened asset(s): D.API_DATA, D.SEC_DATA, D.JCS_DATA and D.CRYPTO.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.SID.1	An applet impersonates another application, or even the Java Card RE, in order to gain illegal access to some resources of the card or with respect to the end user or the terminal. See #.SID for details. Directly threatened asset(s): D.SEC_DATA (other assets may be jeopardized should this attack succeed, for instance, if the identity of the JCRE is usurped), D.PIN and D.APP_KEYs.	T.FLAW_SW
T.SID.2	The attacker modifies the TOE's attribution of a privileged role (e.g. default applet and currently selected applet), which allows illegal impersonation of this role. See #.SID for further details. Directly threatened asset(s): D.SEC_DATA (any other asset may be jeopardized should this attack succeed, depending on whose identity was forged).	T.FLAW_SW
T.EXE-CODE.1	An applet performs an unauthorized execution of a method. See #.EXE-JCS-CODE and #.EXE-APPLI-CODE for details. Directly threatened asset(s): D.APP_CODE.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.EXE-CODE.2	An applet performs an execution of a method fragment or arbitrary data.	Analysis of the composite-

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	See #.EXE-JCS-CODE and #.EXE-APPLI-CODE for details. Directly threatened asset(s): D.APP_CODE.	product threats does not reveal any contradiction with this PF threat.
T.NATIVE	An applet executes a native method to bypass a TOE Security Function such as the firewall. See #.NATIVE for details. Directly threatened asset(s): D.JCS_DATA.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.RESOURCES	An attacker prevents correct operation of the Java Card System through consumption of some resources of the card: RAM or NVRAM. See #.RESOURCES for details. Directly threatened asset(s): D.JCS_DATA.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.
T.DELETION	The attacker deletes an applet or a CAP file already in use on the card, or uses the deletion functions to pave the way for further attacks (putting the TOE in an insecure state). See #.DELETION for details). Directly threatened asset(s): D.SEC_DATA and D.APP_CODE. Note: T.DELETION is a sub-threat of the T.UNAUTHORISED-CARD-MGMT threat mentioned in [PP-GP] and listed in section 6.3.1.	T.UNAUTHORISED-CARD-MGMT
T.INSTALL	The attacker fraudulently installs post-issuance of an applet on the card. This concerns either the installation of an unverified applet or an attempt to induce a malfunction in the TOE through the installation process. See #.INSTALL for details. Directly threatened asset(s): D.SEC_DATA (any other asset may be jeopardized should this attack succeed, depending on the virulence of the installed application). Note: T.INSTALL is a sub-threat of the T.UNAUTHORISED-CARD-MGMT threat mentioned in [PP-GP] and listed in section 6.3.1.	T.UNAUTHORISED-CARD-MGMT
T.OBJ-DELETION	The attacker keeps a reference to a garbage collected object in order to force the TOE to execute an unavailable method, to make it to crash, or to gain access to a memory containing data that is now being	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.

Threats From Platform	Platform Relevant Threats Content	Link to the Composite-Product Threats
	used by another application. See #.OBJ-DELETION for further details. Directly threatened asset(s): D.APP_C_DATA, D.APP_I_DATA and D.APP_KEYs.	
T.PHYSICAL	The attacker discloses or modifies the design of the TOE, its sensitive data or application code by physical (opposed to logical) tampering means. This threat includes IC failure analysis, electrical probing, unexpected tearing, and DPA. That also includes the modification of the runtime execution of Java Card System or SCP software through alteration of the intended execution order of (set of) instructions through physical tampering techniques. This threatens all the identified assets. Application note: as sensitive array and sensitive result are supported by the TOE, this threat also 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. See #.INTEG-APPLI-DATA-PHYS for details. Directly threatened asset(s): D.APP_I_DATA, D.PIN, and D.APP_KEYs.	Analysis of the composite-product threats does not reveal any contradiction with this PF threat.

Table 8 Threats and Security Objectives – Compatibility

4.5.2.STATEMENT OF COMPATIBILITY – OSPS PART

The following table lists the relevant OSPs of the security target [ST_PF] and provides the link to the OSPs related to the composite-product, showing that there is no contradiction between the two.

Platform Relevant OSP Label	Platform Relevant OSP Content	Link To The Composite-Product Threats
From core part		
OSP.AID-MANAGEMENT	When loading an application that uses shareable object interface, to make its services available to	No contradiction with the present evaluation

Platform Relevant OSP Label	Platform Relevant OSP Content	Link To The Composite-Product Threats
	other applications, the VA shall verify that the AID of the application being loaded does not impersonate the AID known by another application on the card for the use of shareable services.	
OSP.LOADING	Application code, validated or certified depending on the application, is loaded onto the SE Platform using any kind of CCM servers (e.g. OTA or other kinds of servers used to perform card content management) and protocols with contactless or contact (e.g. USB) connectivity. If needed, the Issuer can pre authorize content loading operation through delegated management privilege to an individual on-card representative of APs. In that case the application code is loaded in the APSD. Once loaded, the application is personalized using the appropriate SD keys.	OSP.APPS_VALIDATION
OSP.SERVERS	A security policy shall be employed by the Issuer to ensure the security of the applications stored on its CCM servers (e.g. OTA or other kinds of servers used to perform card content management).	OSP.OEM_SERVERS OSP.KTS_SERVER
OSP.APSD-KEYS	The APSD keys personalization can rely either on the key escrow if the APSD has been created before the usage phase of the SE card, or on the CA if the APSD has been created during the usage phase. In the first case, the APSD keys are generated and stored in a secure way by the personalizer. Then, these keys are transmitted to the AP, via the key escrow. In the second case, one of the following must occur: ▪ The APSD keys are generated	No contradiction with the present evaluation

Platform Relevant OSP Label	Platform Relevant OSP Content	Link To The Composite-Product Threats
	and stored in a secure way by the APSD, then securely transmitted to the AP using the CASD. Or the APSD keys are created by the AP and securely transferred to the APSD using the CASD.	
OSP.ISD-KEYS	The security of the ISD keys shall be ensured by a well-defined security policy that covers generation, storage, distribution, destruction, and recovery. This policy is enforced by the Issuer in collaboration with the personaliser.	OSP.SECURE_KEY_RETRIEVE OSP.KEY_SHARE
OSP.KEY-GENERATION	The personaliser shall enforce a policy ensuring that generated keys cannot be accessed in plaintext.	No contradiction with the present evaluation
OSP.CASD-KEYS	The CASD keys shall be securely generated and stored in the SE card during the personalization process. These keys are not modifiable after card issuance.	No contradiction with the present evaluation
OSP.KEY-CHANGE	The AP shall change its initial keys before any operation on its APSD.	No contradiction with the present evaluation
OSP.SECURITY-DOMAINS	SDs can be dynamically created, deleted, and blocked during usage phase, i.e. post issuance.	No contradiction with the present evaluation
OSP.APPLICATIONS	The applications intending to be used with the TOE shall follow the TOE's security guidance and recommendations.	No contradiction with the present evaluation
From package 'Ciphered Load File Data Block (CLFDB)'		
OSP.CLFDB-ENC-PR	The Load File Data Block must be encrypted securely by a trusted SD provider. Application Note: See [GPCS] section C.6.	No contradiction with the present evaluation
From package 'Delegated Management (DM)'		
OSP.TOKEN-GEN	The Token must be generated securely by a trusted entity according to the signature algorithms defined in	No contradiction with the present evaluation

Platform Relevant OSP Label	Platform Relevant OSP Content	Link To The Composite-Product Threats
	GlobalPlatform specifications. Application Note: See [GPCS] sections B.1, B.2, B.3, B.4, and C.4.	
OSP.RECEIPT-VER	The Receipt must be verified securely by a trusted entity according to the methods defined in GlobalPlatform specifications. Application Note: See [GPCS] sections B.1, B.2, B.3, B.4, and C.5.	No contradiction with the present evaluation
From packages 'DAP Verification' and 'Mandated DAP Verification'		
OSP.DAP_BLOCK_GEN	The DAP Block must be generated securely by a trusted entity that verifies the content of the Load File Data Block linked to the hash.	No contradiction with the present evaluation
From PP-Module 'Amendment A: Confidential Card Content Management (CCCM)'		
OSP.CCCM	APs not required to share the Secure Channel keys with the Issuer should use one of the CCCM Models.	No contradiction with the present evaluation
From PP-Module 'Amendment H: Executable Load File Upgrade (ELFU)'		
OSP.ELF_DELE_OP	The TOE shall provide the possibility to perform the deletion operation of the Application instances and ELF(s) in one transaction, so that either a full operation or no operation can occur (atomic and irreversible operation).	No contradiction with the present evaluation
From PP-Module 'OS Update'		
OSP.ATOMIC_ACTIVATION	Additional code has to be loaded and installed on the Initial TOE through an atomic activation to create the Updated TOE. Each additional code shall be identified with unique Identification Data. During such atomic activation, identification Data of the Initial TOE have to be updated to clearly identify the Updated TOE. In case of interruption or incident during activation, the TOE shall remain in its initial state or fail secure.	No contradiction with the present evaluation

Platform Relevant OSP Label	Platform Relevant OSP Content	Link To The Composite-Product Threats
OSP.TOE_IDENTIFICATION	Identification Data of the resulting Updated TOE shall identify the Initial TOE and the activated additional code. Identification Data shall be protected in integrity.	No contradiction with the present evaluation
OSP.ADDITIONAL_CODE_SIGNING	The additional code has to be signed with a cryptographic key according to relevant standards, and the generated signature is associated with the additional code. The additional code signature must be verified during loading to assure its authenticity and integrity and to assure that loading is authorized on the TOE. The cryptographic key used to sign the additional code shall be of sufficient quality and its generation shall be appropriately secured to ensure the authenticity, integrity, and confidentiality of the key.	No contradiction with the present evaluation
OSP.ADDITIONAL_CODE_ENCRYPTION	The additional code has to be encrypted according to the relevant standard in order to ensure its confidentiality when it is transmitted to the TOE for loading and installation. The encryption key shall be of sufficient quality and its generation shall be appropriately secured to ensure the confidentiality, authenticity, and integrity of the key.	No contradiction with the present evaluation
From [PP-JCS] Protection Profile		
OSP.VERIFICATION	This policy shall ensure the consistency between the export files used in the verification and those used for installing the verified file. The policy must also ensure that no modification of the file is performed in between its verification and the signing by the verification authority. See #.VERIFICATION for details.	OSP.OEM_SERVERS

Platform Relevant OSP Label	Platform Relevant OSP Content	Link To The Composite-Product Threats
	If the application development guidance provided by the platform developer contains recommendations related to the isolation property of the platform, this policy shall also ensure that the verification authority checks that these recommendations are applied in the application code.	

Table 9 OSP and Security Objectives - Compatibility

4.5.3.STATEMENT OF COMPATIBILITY – ASSUMPTIONS PART

Assumptions From Platform	Platform Relevant Assumptions Content	Link To The Composite-Product Threats
From core part		
A.ISSUER	This is the entity that owns the SE and is ultimately responsible for the behavior of the SE.	A.OEM_ADMIN A.DEVICE_OEM
A.ADMIN	These administrators of the CCM servers (e.g. OTA or other kinds of servers) used to perform card content management are trusted actors. They are trained to use and administrate those servers securely. They have the means and the equipment to perform their tasks. They are aware of the sensitivity of the assets they manage and the responsibilities associated with the administration of CCM servers. Administrators obey the security policies and constitute, by this assumption, no source of an inside attack.	A.OEM_ADMIN A.DEVICE_OEM_INSIDER
A.APPS-PROVIDER	The AP is a trusted actor that provides applications. APs are responsible for their APSD keys.	No contradiction with the present evaluation
A.VERIFICATION-AUTHORITY	The VA is a trusted actor with the capability to check and validate the digital signature of an application.	No contradiction with the present evaluation
A.KEY-ESCROW	The key escrow is a trusted actor in charge of the secure storage of the initial APSD keys generated by the TOE personaliser during the initial personalisation.	No contradiction with the present evaluation
A.PERSONALISER	The personaliser is in charge of the TOE personalisation process, which ensures the security of the keys loaded in the SE: ▪ Issuer Security Domain keys	No contradiction with the present evaluation

Assumptions From Platform	Platform Relevant Assumptions Content	Link To The Composite-Product Threats
	(ISD keys) ▪ Application Provider Security Domains keys (APSD keys) ▪ Controlling Authority Security Domain keys (CASD keys)	
A.CONTROLLING-AUTHORITY	The CA is a trusted actor different from the issuer responsible for the CASD keys and associated services.	No contradiction with the present evaluation
A.PRODUCTION	Security procedures are used after TOE Delivery up to delivery to the end consumer to maintain the confidentiality and integrity of the TOE and its data (to prevent any possible copy, modification, retention, theft, or unauthorised use).	A.PRODUCTION_ENV
A.SCP-SUPP	The operational environment supports and uses the SCPs offered by the TOE.	No contradiction with the present evaluation
A.KEYS-PROT	The keys stored outside the TOE and applied for secure communication and authentication between the SE and the external entities are confidentiality and integrity protected in their storage environment. This covers D.APSD_KEYS and D.ISD_KEYS.	A.VEHICLE_ROOT_KEY
From PP-Module 'OS Update'		
A.OS-UPDATE-EVIDENCE	For additional code loaded pre-issuance, it is assumed that evaluated technical and/or audited organisational measures have been implemented to ensure that the additional code: 1. has been issued by the genuine OS Developer 2. has not been altered since it was issued by the genuine OS Developer. For additional code loaded post-issuance, it is assumed that the OS Developer provides digital evidence to the TOE in order to prove the following: 1. he is the genuine developer of the additional code and 2. the additional code has not been modified since it was issued by the genuine OS Developer.	No contradiction with the present evaluation
A.SECURE_ACODE_ MANAGEMENT	It is assumed that: ▪ The Key management process related to the OS	No contradiction with the present evaluation

Assumptions From Platform	Platform Relevant Assumptions Content	Link To The Composite-Product Threats
	▪ Update capability takes place in a secure and audited environment. ▪ The cryptographic keys used by the cryptographic operations are of strong quality and appropriately secured to ensure confidentiality, authenticity, and integrity of those keys.	
From [PP-JCS] Protection Profile		
A.CAP_FILE	CAP Files loaded post-issuance do not contain native methods. The Java Card specification explicitly "does not include support for native methods" ([JCVM310], §3.3) outside the API.	No contradiction with the present evaluation
A.VERIFICATION	All the bytecodes are verified at least once, before the loading, before the installation or before the execution, depending on the card capabilities, in order to ensure that each bytecode is valid at execution time.	No contradiction with the present evaluation

Table 10 Asumptions and Security Objectives - Compatibility

5. SECURITY OBJECTIVES

5.1. SECURITY OBJECTIVES FOR THE TOE

This section describes the security objectives for the TOE.

SECURITY OBJECTIVES	DESCRIPTION
O.SE_MANAGEMENT	The TOE SHALL provide secure element management functionalities (loading, installation, extradition, deletion of applications) in charge of the life cycle of the whole DK Applet and installed applications (applets).
O.IMMO_TOK_CONFID	The TOE SHALL ensure the confidentiality of the Immobilizer Token during storage (data at rest), deletion, processing. The Execution Environment SHALL prevent other applets from accessing the DK Applet secret data.
O.IMMO_TOK_INTEG	The TOE SHALL ensure the integrity of the Immobilizer Token during storage (data at rest), deletion, processing. The Execution Environment SHALL prevent other applets from modifying the DK Applet secret data.
O.DK_CONFID	The TOE SHALL ensure the confidentiality of the assets (to be protected in Confidentiality - including cryptographic keys) of the Digital Key during generation, usage (strong cryptography operations), storage, deletion, such that those that are never known outside the DK Applet within its DK Applet EE.
O.DK_INTEG	The TOE SHALL ensure the integrity of assets (to be protected in Integrity - including cryptographic keys) of the Digital Key when it is generated, used, deleted and stored.
O.LONG_TERM_KEY_CONFID	The TOE SHALL ensure the confidentiality of the Long Term key.
O.LONG_TERM_KEY_INTEG	The TOE SHALL ensure the integrity of the Long Term key.
O.SEC_SHARED_KEY_CONFID	The TOE SHALL ensure the confidentiality of the Secret Shared key.
O.SEC_SHARED_KEY_INTEG	The TOE SHALL ensure the integrity of the Secret Shared ke
O.KCMAC_KEY_CONFID	The TOE SHALL ensure the confidentiality of the Kcmac key.
O.KCMAC_KEY_INTEG	The TOE SHALL ensure the integrity of the Kcmac key.
O.SESSION_KEYS_CONFID	The TOE SHALL ensure the confidentiality of the session keys.
O.SESSION_KEYS_INTEG	The TOE SHALL ensure the integrity of the session keys.
O.ATTESTATION_ON_DELETION	The TOE SHALL ensure that it creates a deletion attestation for the requested key, and that it is securely deleted before the attestation is transferred to the requesting party.
O.RANDOMNESS	Only random number generators (RNG) generating sufficient entropy2 SHALL be used in the TOE.
O.IC_SUPPORT	The TOE SHALL provide protection against manipulation of the TOE (including its software and TSF data), the Security IC Embedded Software and the user data of the Composite TOE. This includes protection against: • reverse-engineering (understanding the design and its properties and functions), • manipulation of the hardware and any data, as well as • undetected manipulation of memory contents. (see O.Phys-Manipulation [PP-0084])

	The TOE SHALL provide protection against disclosure/reconstruction of user data while stored in protected memory areas and processed or against the disclosure of other critical information about the operation of the TOE (see O.Phys_Probing [PP-0084])
O.RECOVERY	The TOE SHALL ensure its correct operation. The TOE SHALL indicate or prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent malfunctions. Examples of environmental conditions are voltage, clock frequency, temperature, or external energy fields. The TOE SHALL be able to recover to a stable secure state. (see O.MALFUNCTION [PP-0084])
O.OS_SUPPORT	The TOE SHALL provide protection against disclosure of confidential data stored and/or processed in the Security IC - by measurement and analysis of the shape and amplitude of signals (for example on the power, clock, or I/O lines) and - by measurement and analysis of the time between events found by measuring signals (for instance on the power, clock, or I/O lines). (see O.Leak-Inherent [PP-0084])
O.FAST_TRANSACTION_AUTH	The TOE SHALL guarantee at least a secure Device authentication to the Vehicle (Fast Transaction).
O.STD_TRANSACTION_AUTH	The TOE SHALL guaranty mutual authentication with the Vehicle (Standard Transaction) and from the device's perspective, this guarantees that no private assets can be leaked by a MITM attack. This principle also allows the device to transmit data to the vehicle without any possibility of leakage by a passive or active eavesdropper.
O.KEY_EXCHANGE_AUTH	The TOE SHALL be able to guarantee the authenticity of the key exchange operation.
O.NON-TRACEABILITY	The TOE SHALL be able to ensure the non-traceability of data and keys being shared through an NFC channel.

Table 11 Description of ToE Security Objectives

5.2. SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT

This section introduces the security objectives to be achieved by the environment associated to the TOE. The significant security objectives for the environment of the TOE are the ones linked to relevant assumptions and OSPs.

SECURITY OBJECTIVES	DESCRIPTION
OE. DEVICE_PERSISTENCE	The device will perform self-tests to ensure the integrity of critical functionality, software/firmware and data is maintained in order to ensure the integrity of the Mobile Device is maintained conformant.
OE.APPLET_EE_HW_MALFUNCTION _PROTECTION	The DK Applet EE SHALL ensure its correct operation and is expected to indicate or prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent malfunctions. Examples of environmental conditions are voltage, clock frequency, temperature, or external energy fields.
OE. CERTIFICATE_REVOCATION_SET	The ecosystem SHALL inform all the relevant parties of the

	ecosystem (i.e. those who could be presented this certificate at any point in time), upon request of revocation of a certificate part of a digital key certificate chain (device side or vehicle side). This includes the CA that issued the certificate.
OE. APPLET_ABUSE_PROTECTION	The DK EE, Device OEM Server SHALL prevent those functions (which may not be used after Delivery) from being abused in order to disclose, manipulate critical assets such as Owner DK Secret, or manipulate, bypass, deactivate, change or explore security features or security services of the DK EE, Applet or Device OEM Server.
OE. INSTANCE_CA_CONFIDENTIALITY	The Certificate Chain (especially the INSTANCE CA, INSTANCE CA ATTESTATION and DEVICE OEM CA) SHALL be protected such that an attacker is not able to create a correctly attestable INSTANCE CA outside of the certified DK Applet / DK Applet EE.
OE. SECURE_DEVELOPMENT_AND_PRODUCTION	This objective SHALL ensure that any attack by internal attackers (employees, visitors) in development, production and provisioning, to directly or indirectly compromise the certificate chain or the Digital Key secret itself are prevented. This SHALL enforce that only trusted personnel are appointed for the abovementioned processes.
OE.DEVICE_OEM	A Device OEM SHALL verify the validity of the DK Applet certificate provided by the CCC (according to the CCC Certification Program) before deploying it.
OE.OEM_SERVERS	Administrators of the OEM server are trusted people. They have been well trained to use and administrate the server securely. The Device OEM Server and the Vehicle OEM Server SHALL be hosted in secure data centers. PKI signature keys and secure operation of the PKI instances must be managed securely.
OE.KTS_SERVER	The device SHALL ensure that unnecessary PII of the device's user is not sent to the Key Tracking Server, thus preventing to track the Vehicle owner. The KTS server SHALL preserve the confidentiality of the user data being received and transmitted.
OE.APPS_VALIDATION	There SHALL be a mechanism to verify/validate the application before being loaded/installed.
OE.PRODUCTION_ENV	The production environment SHALL be equipped with trusted personnel and the development SHALL take place based on the security guidelines that has been put in place. Also, production and provisioning should take place based on the guidelines, to prevent direct or indirect compromise of the certificate chain or the Digital Key secret itself
OE.OS_DOWNGRADE	Adequate issuance measures SHALL be in place to prevent loading older versions of the device's software and firmware that has publicly known security flaws (downgrade).
OE.ANTI_DOWNGRADE	An attacker SHALL not be able to downgrade the protocol to an older version that has publicly known security flaws. DK material SHALL be cleared if downgrade is performed.
OE. DK_PROTOCOL_SECURITY	The device SHALL implement strong communication protocol

	to prevent anti-relay, anti-replay, Man in the middle. This includes implementation or robust integrity mechanisms, use of strong cryptography and random number generators.
OE.SECURE_KEY_RETRIEVE	The Device SHALL implement a secure key retrieval mechanism such that it prevents unauthorized key retrieval by attackers for gaining access to the communication channel.
OE.KEY_SHARE	The device SHALL protect the integrity & confidentiality of the keys being shared.
OE.KEY_OPTIONS	The Device OS SHALL be able to protect the integrity & confidentiality of the KEY options.
OE.CLOCK	The Device OS should provide a reliable source for the clock.
OE.CAR_LOCATION	The Device OS SHALL prevent any display of information related to the location of paired vehicles when locked.
OE.USER_AUTHENTICATION_DK_SH ARING	The device SHALL prevent the sharing if the origin of the authorization (Owner Authentication) is not ensured. Additionally, an unambiguous signal of friend identity SHALL be established before initiating friend sharing. The device SHALL prevent the sharing if the recipient identity cannot be traced back to the friend.
OE. USER_PRIVACY_CONSENT	OEM native app or device framework SHALL make user aware and asks for consent for any private information that is shared by the device with the servers and vehicle.
OE.CERT_VALIDATION	The processing of a certificate provided by an external entity to the device SHALL be verified before being processed. The verification SHALL cover the certificate format and validity.
OE.RADIO_RELAY	There should be mechanism to detect the relay of a transaction using radio equipment.
OE. UNAUTHORIZED_KEY_SHARING	Without the consent/knowledge of the owner, the collaborating entities SHALL not share the keys/access credentials.
OE.DOS_DETECT	A DOS detection mechanism SHALL be implemented to ensure the availability by detecting whether the device is disabled by some means, if the internet connectivity is available, etc..
OE.REPLAY	A detection mechanism SHALL be implemented to ensure that the NFC signal is not being observed and replayed.
OE.COMMUNICATION	A mechanism SHALL be implemented in order to ensure the integrity, confidentiality and authentication of the data transferred through the communication channel.
OE.VEHICLE_ROOT_KEY_CONFID	The vehicle – ECU shall ensure the confidentiality of the vehicle root key (long term key).

Table 12 Description of Operational Environment Security Objectives

5.3. SECURITY OBJECTIVES RATIONALE

5.3.1. THREATS COVERAGE

Threats From [PP-CCC-CP-023]	Security Objectives	Rationale

Threats From [PP-CCC-CP-023]	Security Objectives	Rationale
T.DK_PHYSICAL	O.OS_SUPPORT, O.IC_SUPPORT, O.RANDOMNESS	This threat is countered by physical protections which rely on the underlying platform and the secure element physical protection capabilities
T.UNAUTHORIZED_SE_MNG	O.SE_MANAGEMENT, OE.COMMUNICATION	This threat is covered by O.SE_MANAGEMENT which controls the access to card management functions such as the loading, installation, extradition or deletion of applets and OE.COMMUNICATION which ensures the integrity, confidentiality and authentication of the data transferred through the communication channel.
T.LIFE_CYCLE	O.SE_MANAGEMENT	This threat is covered by O.SE_MANAGEMENT which controls the access to card management functions such as the loading, installation, extradition or deletion of applets and prevent attacks intended to modify or exploit the current life cycle of applications
T.IT_DISCLOSURE	O.IMMO_TOK_CONFID O.IC_SUPPORT	This threat is covered by O.IMMO_TOK_CONFID which ensures the confidentiality of the Immobilizer Token during storage (data at rest), deletion, processing. It is also supported by O.IC_SUPPORT which protects IMMOTOKEN from disclosure due to physical attacks.
T.DK_DISCLOSURE	O.DK_CONFID O.RANDOMNESS	This threat is covered by O.DK_CONFID which ensures the confidentiality of the secret elements of the Digital Key during generation, usage (strong cryptography operations), storage, deletion, such that those are never known outside the DK Applet within its DK Applet EE. It is also covered by O.RANDOMNESS which ensure covering a lack of randomness that could allow an attacker to predict communication.
T.FLAW_SW	O.SE_MANAGEMENT, OE.DEVICE_PERSISTENCE, OE.APPLET_ABUSE_PROTECTION,	This threat is covered by: • O.SE_MANAGEMENT which controls the access to card management functions such as the loading, installation, extradition or deletion of applets

Threats From [PP-CCC-CP-023]	Security Objectives	Rationale
		• OE.DEVICE_PERSISTENCE ensures that the device will perform self-tests to ensure the integrity of critical functionality, software/firmware and data is maintained • OE.APPLET_ABUSE_PROTECTION ensures that DK EE, Applet, Device OEM Backend prevents that functions which may not be used after Delivery can be abused in order to disclose, manipulate critical assets
T.RETRIEVE_SECRET-SHARED-KEY_DKA	O.SEC_SHARED_KEY_CONFID	This threat is covered by O.SEC_SHARED_KEY_CONFID ensures that the DK Applet ensures the confidentiality of the Secret Shared key.
T.RETRIEVE_KCMAC-KEY_DKA	O.KCMAC_KEY_CONFID	This threat is covered by O.KCMAC_KEY_CONFID ensures that the DK Applet ensures the confidentiality of the Kcmac key.
T.RETRIEVE_SESSION-KEYS_DKA	O.SESSION_KEYS_CONFID	This threat is covered by O.SESSION_KEYS_CONFID ensures that the DK Applet ensures the confidentiality of the session keys.
T.UNAUTHORIZED_ACCESS_DK_ASSET	O.IMMO_TOK_CONFID	This threat is covered by • O.IMMO_TOK_CONFID which ensures the confidentiality of the Immobilizer Token during storage (data at rest), deletion, processing.
T.CA_KEY_LEAK	O.DK_CONFID, O.LONG_TERM_KEY_CONFID OE.VEHICLE_ROOT_KEY_CONFID	This threat is covered by • O.DK_CONFID which ensures the confidentiality of the secret elements of the Digital Key during generation, usage (strong cryptography operations), storage, deletion, such that those are never known outside the DK Applet within its DK Applet EE. • O.LONG_TERM_KEY_CONFID which ensures the confidentiality of long term key. • OE.VEHICLE_ROOT_KEY_CONFID which ensures the confidentiality of the long term key residing on the vehicle – ECU.
T.DENIAL_OF_LEGITIMATE_DELETIONS	O.ATTESTATION_ON_DELETION	This threat is covered by O.ATTESTATION_ON_DELETION which ensures that the DK applet

Threats From [PP-CCC-CP-023]	Security Objectives	Rationale
		creates a deletion attestation for the requested key, and that it is securely deleted before the attestation is transferred to the requesting party.
T.DK_SK_MODIFICATIONS	O.DK_INTEG O.IMMO_TOK_INTEG	This threat is covered by the following two security objectives O.DK_INTEG which ensures that the DK Applet and its Execution Environment ensure the integrity of the secret elements of the Digital Key when it is generated, used, deleted and stored. And O.IMMO_TOK_INTEG which ensures the integrity of the Immobilizer Token during storage (data at rest), deletion, processing.
T.RADIO_SNIFF	O.FAST_TRANSACTION_A UTH O.STD_TRANSACTION_AU TH	This threat is covered by O.FAST_TRANSACTION_AUTH and O.STD_TRANSACTION_AUTH which ensure a secure channel is used when communicating between device and vehicle.
T.RADIO_MITM	O.KCMAC_KEY_INTEG, O.SEC_SHARED_KEY_INT EG, O.LONG_TERM_KEY_INTE G, O.SESSION_KEYS_INTEG, O.FAST_TRANSACTION_A UTH O.STD_TRANSACTION_AU TH O.KEY_EXCHANGE_AUTH	This threat is covered by • O.KCMAC_KEY_INTEG which ensures the integrity of Kcmac key. • O.SEC_SHARED_KEY_INTEG which ensures the integrity of secret shared keys • O.LONG_TERM_KEY_INTEG which ensures the integrity of long term key • O.SESSION_KEYS_INTEG which ensures the integrity of session keys • O.FAST_TRANSACTION_AUTH which ensures the authentication takes place from device side • O.STD_TRANSACTION_AUTH which ensures that mutual authentication takes place between device and vehicle. • O.KEY_EXCHANGE_AUTH ensures the authenticity of key share operation.
T.DATA_BREACH	O.DK_CONFID O.NON-TRACEABILITY	This threat is covered by the following security objectives on the TOE: • O.DK_CONFID which ensures the confidentiality of data/keys being shared during a transaction through NFC channel. • O.NON-TRACEABILITY which

Threats From [PP-CCC-CP-023]	Security Objectives	Rationale
		ensures that the users are non-traceable across different vehicles through the same app.

Table 13 Threats and Security Objectives - Coverage

Threats From [PP-CCC-CP-023]	OE Security Objectives	Rationale
T.NON_REVOKED	OE.CERTIFICATE_REVOCATION_SET	This threat is covered by OE. CERTIFICATE_REVOCATION_SET which ensures the ecosystem informs all the relevant parties of the ecosystem upon request of revocation of a certificate part of a digital key certificate chain (device side or vehicle side)
T.KTS_DATA_LEAK	OE.KTS_SERVER	This threat is covered by OE.KTS_SERVER which ensures that unnecessary PII of the device's user are not sent to the Key Tracking Server, thus preventing to track the Vehicle owner
T.UPDATE_KEY_OPTIONS	OE.KEY_OPTIONS	This threat is covered by OE.KEY_OPTIONS ensures that the DK Applet ensures the integrity of the key options
T.DEVICE_THEFT	OE.USER_AUTHENTICATION_DK_SHARING	This threat is covered by OE.USER_AUTHENTICATION_DK_SHA RING which ensures that device provides robust User Authentication methods to identify the DK User & an unambiguous signal of friend identity be established before initiating friend sharing.
T.PROTOCOL_DOWNGRAD E	OE.ANTI_DOWNGRADE	This threat is covered by OE.ANTI_DOWNGRADE which ensures that an attacker will not be able to downgrade the protocol to an older version that has publicly known security flaws.
T.SIGN_COMPROMISE_VERIFY_COMPROMISE	OE.CERT_VALIDATION	This threat is covered by OE.CERT_VALIDATION which ensures the protection of creation and validation of electronic signatures
T.RADIO_RELAY_TRANSA CTION	OE.RADIO_RELAY	This threat is covered by OE.RADIO_RELAY which ensures protection against relay a transaction with radio equipment attack.
T.INTERNET_CONNECTIVIT	OE.DOS_DETECT	This threat is covered by

Threats From [PP-CCC-CP-023]	OE Security Objectives	Rationale
Y_DOS		OE.DOS_DETECT which ensures availability by detecting whether the device is disabled by some means, if the internet connectivity is available, etc.
T.TIME_CHANGE	OE.CLOCK	This threat is covered by OE.CLOCK which ensures that the software uses a reliable source for the clock (date/time).
T.REPLAY_TRANSACTION	OE.REPLAY	This threat is covered by OE.REPLAY which protects against attack such as replay an observed NFC transaction to the vehicle
T.UNAUTHORIZED_KEY_SHARIN G	OE.UNAUTHORIZED_KEY _ SHARING	This threat is covered by OE. UNAUTHORIZED_KEY_SHARING which ensures that two or more collaborating adversaries cannot share (copy) access credentials without the vehicle owner's consent or knowledge.
T.INSTANCE_CA_DISCLOS URE	OE.INSTANCE_CA_CONFI DENTIALITY	This threat is covered by OE. INSTANCE_CA_CONFIDENTIALITY which ensures that the Certificate Chain (especially the INSTANCE CA, INSTANCE CA ATTESTATION and DEVICE OEM CA) is protected such that an attacker is not able to create a correctly attestable INSTANCE CA outside of the certified DK Applet / DK Applet EE

Table 14 Threats and OE.Security Objectives - Coverage

The OE's listed in the following table participates in covering the identified threats but cannot be solely implemented to cover these threats sufficiently.

Security Objectives	Threats
O.SE_MANAGEMENT	T.UNAUTHORIZED_SE_MNG, T.LIFE_CYCLE
O.IMMO_TOK_CONFID	T.IT_DISCLOSURE, T.UNAUTHORIZED_ACCESS_DK_ASSET
O.IMMO_TOK_INTEG	T.DK_SK_MODIFICATIONS
O.DK_CONFID	T.DK_DISCLOSURE, T.CA_KEY_LEAK, T.DATA_BREACH
O.DK_INTEG	T.DK_SK_MODIFICATIONS
O.LONG_TERM_KEY_CONFID	T.CA_KEY_LEAK
O.LONG_TERM_KEY_INTEG	T.RADIO_MITM
O.SEC_SHARED_KEY_CONFID	T.RETRIEVE_SECRET-SHARED-KEY_DKA
O.SEC_SHARED_KEY_INTEG	T.RADIO_MITM
O.KCMAC_KEY_CONFID	T.RETRIEVE_KCMAC-KEY_DKA
O.KCMAC_KEY_INTEG	T.RADIO_MITM
O.SESSION_KEYS_CONFID	T.RETRIEVE_SESSION-KEYS_DKA
O.SESSION_KEYS_INTEG	T.RADIO_MITM
O.ATTESTATION_ON_DELETION	T.DENIAL_OF_LEGITIMATE_DELETIONS
O.RANDOMNESS	T.DK_PHYSICAL

O.IC_SUPPORT	T.DK_PHYSICAL
O.OS_SUPPORT	T.DK_PHYSICAL
O.FAST_TRANSACTION_AUTH	T.RADIO_MITM
O.STD_TRANSACTION_AUTH	T.RADIO_MITM
O.KEY_EXCHANGE_AUTH	T.RADIO_MITM
O.NON-TRACEABILITY	T.DATA_BREACH
OE. DEVICE_PERSISTENCE	T.FLAW_SW
OE. APPLET_ABUSE_PROTECTION	T.FLAW_SW
OE.COMMUNICATION	T.UNAUTHORIZED_SE_MNG
OE.CERTIFICATE_REVOCATION_SET	T.NON_REVOKED
OE.KTS_SERVER	T.KTS_DATA_LEAK
OE.KEY_OPTIONS	T.UPDATE_KEY_OPTIONS
OE.USER_AUTHENTICATION_DK_SHARING	T.DEVICE_THEFT, T.RADIO_SNIFF
OE.ANTI_DOWNGRADE	T.PROTOCOL_DOWNGRADE
OE.CERT_VALIDATION	T.SIGN_COMPROMISE_VERIFY_COMPROMISE
OE.RADIO_RELAY	T.RADIO_RELAY_TRANSACTION
OE.DOS_DETECT	T.INTERNET_CONNECTIVITY_DOS
OE.CLOCK	T.TIME_CHANGE
OE.REPLAY	T.REPLAY_TRANSACTION
OE. UNAUTHORIZED_KEY_SHARING	T.UNAUTHORIZED_KEY_SHARING
OE.INSTANCE_CA_CONFIDENTIALITY	T.INSTANCE_CA_DISCLOSURE
OE.VEHICLE_ROOT_KEY_CONFID	T.CA_KEY_LEAK

Table 15 Security Objectives and Threats – Coverage

5.3.2.OSP COVERAGE

OSP From [PP-CCC-CP-023]	Security Objectives	Rationale
OSP.APPS_VALIDATION	OE.APPS_VALIDATION	This OSP is enforced by the security objective for the operational environment of the TOE OE.APPS_VALIDATION
OSP.OEM_SERVERS	OE.OEM_SERVERS	This OSP is enforced by the security objective for the operational environment of the TOE OE.OEM_SERVERS
OSP.KTS_SERVER	OE.KTS_SERVER	This OSP is enforced by the security objective for the operational environment of the TOE OE.KTS_SERVER
OSP.OS_DOWNGRADE	OE.OS_DOWNGRADE	This OSP is enforced by the security objective for the operational environment of the TOE OE.OS_DOWNGRADE
OSP.SECURE_KEY_RETRI EVE	OE.SECURE_KEY_RETRIE VE	This OSP is enforced by the security objective for the operational environment of the TOE OE.SECURE_KEY_RETRIEVE
OSP.KEY_SHARE	OE.KEY_SHARE	This OSP is enforced by the security objective for the operational environment of the TOE

OSP From [PP-CCC-CP-023]	Security Objectives	Rationale
		OE.KEY_SHARE
OSP.KEY_OPTIONS	OE.KEY_OPTIONS	This OSP is enforced by the security objective for the operational environment of the TOE OE.KEY_OPTIONS
OSP.SERVER_COMMUNICATION	OE.OEM_SERVERS, OE.KTS_SERVER	This OSP is enforced by the security objective for the operational environment of the TOE OE.OEM_SERVERS and OE.KTS_SERVER
OSP.PROTOCOL_FAILURE	OE. DK_PROTOCOL_SECURITY	This OSP is enforced by the security objective for the operational environment of the TOE OE. DK_PROTOCOL_SECURITY
OSP.DOS_DETECT	OE.DOS_DETECT	This OSP is enforced by the security objective for the operational environment of the TOE OE.DOS_DETECT
OSP.CERTIFICATE	OE.CERT_VALIDATION	This OSP is enforced by the security objective for the operational environment of the TOE OE.CERT_VALIDATION
OSP.PKI_POLICY	OE.OEM_SERVERS	This OSP is enforced by the security objective for the operational environment of the TOE OE.OEM_SERVERS

Table 16 OSPs and Security Objectives - Coverage

Security Objectives	OSP
OE. DEVICE_PERSISTENCE	
OE.APPLET_EE_HW_MALFUNCTION_PROTECTION	
OE. CERTIFICATE_REVOCATION_SET	
OE. APPLET_ABUSE_PROTECTION	
OE. INSTANCE_CA_CONFIDENTIALITY	
OE. SECURE_DEVELOPMENT_AND_PRODUCTION	
OE.DEVICE_OEM	
OE.OEM_SERVERS	OSP.OEM_SERVERS, OSP.SERVER_COMMUNICATION OSP.PKI_POLICY
OE.KTS_SERVER	OSP.KTS_SERVER, OSP.SERVER_COMMUNICATION
OE.APPS_VALIDATION	OSP.APPS_VALIDATION
OE.PRODUCTION_ENV	
OE.OS_DOWNGRADE	OSP.OS_DOWNGRADE
OE.ANTI_DOWNGRADE	
OE. DK_PROTOCOL_SECURITY	OSP.PROTOCOL_FAILURE
OE.SECURE_KEY_RETRIEVE	OSP.SECURE_KEY_RETRIEVE
OE.KEY_SHARE	OSP.KEY_SHARE
OE.KEY_OPTIONS	OSP.KEY_OPTIONS

Security Objectives	OSP
OE.CLOCK	
OE.CAR_LOCATION	
OE.USER_AUTHENTICATION_DK_SHARING	
OE.USER_PRIVACY_CONSENT	
OE.CERT_VALIDATION	OSP.CERTIFICATE
OE.RADIO_RELAY	
OE. UNAUTHORIZED_KEY_SHARING	
OE.DOS_DETECT	OSP.DOS_DETECT
OE.REPLAY	
OE.COMMUNICATION	
OE.VEHICLE_ROOT_KEY_CONFID	

Table 17 Security Objectives and OSPs - Coverage

5.3.3.ASSUMPTIONS COVERAGE

Assumptions From This PP	Security Objectives For The Operational Environment	Rationale
A.USER_AUTHENTICATION	OE.USER_AUTHENTICATION_DK_SHARING	This assumption is directly upheld by OE.USER_AUTHENTICATION_DK_SHARING
A.USER_PRIVACY_CONSENT	OE.USER_PRIVACY_CONSENT	This assumption is directly upheld by OE.USER_PRIVACY_CONSENT
A.CERTIFICATION_REVOCATION_SET	OE.CERTIFICATE_REVOCATION_SET	This assumption is directly upheld by OE.CERTIFICATE_REVOCATION_SET
A.OEM_ADMIN	OE.OEM_SERVERS	This assumption is directly upheld by OE.OEM_SERVERS
A.PRODUCTION_ENV	OE.PRODUCTION_ENV	This assumption is directly upheld by OE.PRODUCTION_ENV
A.DEVICE_OEM	OE.DEVICE_OEM	This assumption is directly upheld by the OE.DEVICE_OEM
A.OS_CLOCK	OE.CLOCK	This assumption is directly upheld by OE.CLOCK
A.CAR_LOCATION	OE.CAR_LOCATION	This assumption is directly upheld by OE.CAR_LOCATION
A.DEVICE_OEM_INSIDER	OE. SECURE_DEVELOPMENT_AND_PRODUCTION	This assumption is directly upheld by OE. SECURE_DEVELOPMENT_AND_PRODUCTION
A.SHARING_MASQUERADE	OE.KEY_SHARE	This assumption is directly upheld by OE.KEY_SHARE
A.DEVICE_SAFETY	OE.USER_AUTHENTICATION_DK_SHARING	This assumption is directly upheld by OE.USER_AUTHENTICATION_DK_SHARING
A.REPLAY	OE.REPLAY	This assumption is directly upheld by OE.REPLAY
A.RADIO_RELAY	OE.RADIO_RELAY	This assumption is directly upheld by OE.RADIO_RELAY
A.OEM_SERVER_SECURITY	OE.OEM_SERVERS	This assumption is directly upheld by OE.OEM_SERVERS

Assumptions From This PP	Security Objectives For The Operational Environment	Rationale
A.VEHICLE_ROOT_KEY	OE.VEHICLE_ROOT_KEY_CONFID	This assumption is directly upheld by OE.VEHICLE_ROOT_KEY_CONFID

Table 18 Assumptions and Security Objectives for the Operational Environment - coverage

SECURITY OBJECTIVES	ASUMPTIONS
OE. DEVICE_PERSISTENCE	
OE.APPLET_EE_HW_MALFUNCTION_PROTECTION	
OE.CERTIFICATE_REVOCATION_SET	A.CERTIFICATE_REVOCATION_SET
OE.APPLET_ABUSE_PROTECTION	
OE.INSTANCE_CA_CONFIDENTIALITY	
OE.SECURE_DEVELOPMENT_AND_PRODUCTION	
OE.DEVICE_OEM	A.DEVICE_OEM
OE.OEM_SERVERS	A.OEM_ADMIN, A.OEM_SERVER_SECURITY
OE.KTS_SERVER	
OE.APPS_VALIDATION	
OE.PRODUCTION_ENV	A.PRODUCTION_ENV
OE.OS_DOWNGRADE	
OE.ANTI_DOWNGRADE	
OE.DK_PROTOCOL_SECURITY	
OE.SECURE_KEY_RETRIEVE	
OE.KEY_SHARE	A.SHARING_MASQUERADE
OE.KEY_OPTIONS	
OE.CLOCK	A.OS_CLOCK
OE.CAR_LOCATION	A.CAR_LOCATION
OE.USER_AUTHENTICATION_DK_SHARING	A.USER_AUTHENTICATION, A.DEVICE_SAFETY
OE. USER_PRIVACY_CONSENT	A.USER_PRIVACY_CONSENT
OE.CERT_VALIDATION	
OE.RADIO_RELAY	A.RADIO_RELAY
OE. UNAUTHORIZED_KEY_SHARING	
OE.DOS_DETECT	
OE.REPLAY	A.REPLAY
OE.COMMUNICATION	
OE.VEHICLE_ROOT_KEY_CONFID	A.VEHICLE_ROOT_KEY

Table 19 Security Objectives for the Operational Environment and Assumptions - coverage

5.4. COMPOSITION TASKS – OBJECTIVES PART

5.4.1. STATEMENT OF COMPATIBILITY – TOE OBJECTIVES PART

The following table lists the relevant TOE security objectives of the security target [ST_PF], and provides the link to the composite-product TOE security objectives, showing that there is no contradiction between the two sets of objectives.

5.4.1.1. FROM [PP-GP] PROTECTION PROFILE

Platform TOE Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
From core part		
O.CARD-MANAGEMENT	The TOE shall provide the card manager as defined in [GPCS]. The card manager 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. 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 shall prevent card content management operations (loading, installation, deletion) from being carried out, for instance, at invalid states of the card or by unauthorised actors. It shall also enforce security policies established by the Issuer.	**O.SE_MANAGEMENT**
O.DOMAIN-RIGHTS	The Issuer shall not access or change personalised APSD keys, which belong exclusively to the AP. Modification of an SD key set is restricted to the AP owning the SD.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.APPLI-AUTH	The card manager shall enforce the application security policies established by the Issuer. The enforcement shall be implemented by requiring application authentication during application loading on the card.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.SECURITY-DOMAINS	SDs can be dynamically created, deleted, and blocked during the end	

Platform TOE Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
	use phase.	
O.COMM-AUTH	The TOE shall authenticate the origin of the card management requests received by the card, and authenticate itself to the remote actor.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.COMM-INTEGRITY	The TOE shall verify the integrity of the (card management) requests that the card receives.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.COMM-CONFIDENTIALITY	The TOE shall be able to process card management requests containing encrypted data.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.NO-KEY-REUSE	The TOE shall ensure that session keys can be used only once.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.PRIVILEGES-MANAGEMENT	The TOE shall provide Privileges assignment and management functionalities for the on-card entities ISD, SSD, and Applications. The TOE shall control the access to the Privileges assignment and management functions.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.LC-MANAGEMENT	The TOE shall provide a state machine that enforces the TOE's life cycle, keeps track of the TOE's current state, and controls that the operations required by the users are consistent with the current life cycle state of the TOE. The TOE shall provide Life Cycle (LC) management functionalities for the Card, ELFs, SDs, and Applications.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
From package 'Ciphered Load File Data Block (CLFDB)'		
O.CLFDB-DECIPHER	If the SD to be associated with the Executable Load File has the Ciphered Load File Data Block privilege, then the card shall support encryption schemes as defined by GlobalPlatform specifications and the	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.

Platform TOE Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
	SD shall be able to decipher the Ciphered Load File Data Blocks. *Application Note*: See [GPCS] section C.6.	
From package 'Cardholder Verification Method (CVM)'		
O.GLOBAL-CVM	The TOE shall restrict the modification of the security attributes of the CVM only to defined privileged applications appointed by the Card Manager. Any SD allowed to perform CVM can grant the CVM privilege to an Application.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.CVM-BLOCK	If the maximum number of attempts has been reached, further Cardholder authentication attempts are blocked. The blocking can be removed by special action of the Card Manager or a privileged user.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.CVM-MGMT	The TOE shall provide means to securely manage CVM objects. Secure management of CVM objects includes: • Atomic update of PIN code and of the try counter, • No rollback of the number of unsuccessful authentication attempts, • Protection of confidentiality of the PIN value, Protection of the PIN comparison process against observation.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
From package 'Delegated Management (DM)'		
O.RECEIPT	The TOE shall generate non-repudiable receipts of the completion of card management operations. The generation of the receipt shall be performed by an SD with 'Receipt Generation' Privilege.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.TOKEN	The TOE shall verify tokens during the processing of card management operations. The verification of the token shall be performed by an SD with 'Token Verification' Privilege.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
From PP-Module 'Amendment A: Confidential Card Content Management (CCCM)'		

Platform TOE Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
O.CCCM	The TOE shall address the Confidential Card Content Management requirements defined in [Amd A]. These requirements are: ▪ Secure personalisation of APSD by the CA using one of the following scenarios: Pull Model, Push Model, Key Agreement Model, or Key Agreement Model with no Secure Channel ▪ Confidential loading of initial Secure Channel Key Sets Confidential loading of applications by an AP	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
From PP-Module 'Amendment C: Contactless Services (CTL)'		
O.CTL_REGISTRY	The CRS shall ensure that only authorised changes in the Contactless Registry are performed. The SET STATUS command shall only impact CRS-registered applications and shall not perform unauthorised state transitions. The Contactless Registry shall be integrity protected like other data in the OPEN. The CRS shall ensure that the activation state of CRS-registered applications reflects the Contactless Registry content.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.CTL_SC	The CRS shall ensure that the STORE DATA command to modify blacklists of CCM tokens or to change the CRS visibility state on the CTL interface comes through a Secure Channel with at least level "AUTHENTICATED".	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.CRS_PRIVILEGES	The CRS shall securely manage the assignment of the 'Contactless Activation' Privilege and the 'Global Registry' Privilege.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.CRS_COUNTERS	The CRS shall ensure that the Update Counters are protected for integrity and increased by one at	No direct link with the composite product security objectives, but this platform security objective

Platform TOE Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
	each completed operation or sequence of operations.	is used to endure the security of the composite TOE.
From PP-Module 'Amendment H: Executable Load File Upgrade (ELFU)'		
O.ELF_AUTHORISED	Only authorised entities shall be able to load ELFs.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.ELF_INTEGRITY	The ELF integrity shall be preserved during the loading process – (confidentiality maintained if required).	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.ELF_APP_DATA	The application instance data shall be securely stored when saved. The OPEN shall maintain the integrity & consistency of Registry data.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.ELF_SESSION	The session status shall be consistent throughout the upgrade process. Forbidden commands shall be rejected during the upgrade process.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.ELF_DELE_IRR	The TOE must be able to provide an atomic and irreversible deletion operation of the Application instances and ELF(s).	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.ELF_DATA_PRO	The TOE must ensure that any ELF information contained in a protected resource is not inappropriately disclosed when the resource is reallocated.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
From PP-Module 'OS Update'		
O.SECURE_LOAD_ACODE	The TOE shall check an evidence of authenticity and integrity of the additional code to be loaded. The TOE enforces that only an 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 TOE.	Not relevant for this composite product security objectives. The OS agility mechanism is fully managed by the platform.

Platform TOE Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
	During the loading of the additional code, the TOE shall remain secure.	
O.SECURE_AC_ACTIVATION	Activation of the additional code and update of the Identification Data shall be performed at the same time in an atomic way. All the operations needed for the code to be able to operate as in the Updated 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 prevents the forming of the Updated TOE), the TOE shall preserve a secure state.	Not relevant for this composite product security objectives. The OS agility mechanism is fully managed by the platform.
O.TOE_IDENTIFICATION	The TOE provides means to store Identification Data in its non-volatile memory and guarantees the integrity of these data. After atomic activation of the additional code, the Identification Data of the Updated TOE allows identifications of both the Initial TOE and additional code. The user must be able to uniquely identify Initial TOE and additional code(s) which are embedded in the Updated TOE.	Not relevant for this composite product security objectives. The OS agility mechanism is fully managed by the platform.
O.CONFID-OS-UPDATE.LOAD	The TOE shall decrypt the additional code prior installation. *Application Note*: Confidentiality protection must be enforced when the additional code is transmitted to the TOE for loading (See OE.OS-UPDATE-ENCRYPTION later in this table). Confidentiality protection can be achieved either through direct encryption of the additional code, or by means of a trusted path ensuring the confidentiality of the communication to the TOE.	Not relevant for this composite product security objectives. The OS agility mechanism is fully managed by the platform.

Table 20 Statement of compatibility – TOE Objectives part from [PP-GP] Protection Profile

5.4.1.2. FROM [PP-JCS] PROTECTION PROFILE

Platform Toe Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
From core part		
O.SID	The TOE shall uniquely identify every subject (applet, or CAP file) before granting it access to any service.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.FIREWALL	The TOE shall ensure controlled sharing of data containers owned by applets of different CAP files or the JCRE and between applets and the TSFs. See #.FIREWALL for details.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.GLOBAL_ARRAYS_CONFID	The TOE shall ensure that the APDU buffer that is shared by all applications is always cleared upon applet selection. The TOE shall ensure that the global byte array used for the invocation of the install method of the selected applet is always cleared after the return from the install method.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.GLOBAL_ARRAYS_INTEG	The TOE shall ensure that no application can store a reference to the APDU buffer, a global byte array created by the user through makeGlobalArray method and the byte array used for invocation of the install method of the selected applet.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.ARRAY_VIEWS_CONFID	The TOE shall ensure that no application can read elements of an array view not having array view security attribute ATTR_READABLE_VIEW. The TOE shall ensure that an application can only read the elements of the array view within the bounds of the array view.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.ARRAY_VIEWS_INTEG	The TOE shall ensure that no application can write to an array view not having array view security attribute ATTR_WRITABLE_VIEW. The TOE shall ensure that an application can only write within the bounds of the array view.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.NATIVE	The only means that the Java Card VM shall provide for an application to execute native code is the invocation of a method of the Java Card API, or	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.

Platform Toe Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
	any additional API. See #.NATIVE for details.	
O.OPERATE	The TOE must ensure continued correct operation of its security functions. See #.OPERATE for details.	O.IC_SUPPORT O.RECOVERY
O.REALLOCATION	The TOE shall ensure that the re-allocation of a memory block for the runtime areas of the Java Card VM does not disclose any information that was previously stored in that block.	O.OS_SUPPORT
O.RESOURCES	The TOE shall control the availability of resources for the applications. See #.RESOURCES for details.	O.RECOVERY
O.ALARM	The TOE shall provide appropriate feedback information upon detection of a potential security violation. See #.ALARM for details.	O.RECOVERY
O.CIPHER	The TOE shall provide a means to cipher sensitive data for applications in a secure way. In particular, the TOE must support cryptographic algorithms consistent with cryptographic usage policies and standards. See #.CIPHER for details.	O.IMMO_TOK_CONFID O.IMMO_TOK_INTEG O.DK_CONFID O.DK_INTEG O.LONG_TERM_KEY_CONFID O.SEC_SHARED_KEY_CONFID O.SEC_SHARED_KEY_INTEG O.KCMAC_KEY_CONFID O.KCMAC_KEY_INTEG O.SESSION_KEYS_CONFID O.SESSION_KEYS_INTEG
O.RNG	The TOE shall ensure the cryptographic quality of random number generation. For instance random numbers shall not be predictable and shall have sufficient entropy. The TOE shall ensure that no information about the produced random numbers is available to an attacker since they might be used for instance to generate cryptographic keys.	O.RANDOMNESS
O.KEY-MNGT	The TOE shall provide a means to securely manage cryptographic keys. This concerns the correct generation, distribution, access and destruction of cryptographic keys. See #.KEY-MNGT.	O.IMMO_TOK_CONFID O.IMMO_TOK_INTEG O.DK_CONFID O.DK_INTEG O.LONG_TERM_KEY_CONFID O.SEC_SHARED_KEY_CONFID

Platform Toe Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
		O.SEC_SHARED_KEY_INTEG O.KCMAC_KEY_CONFID O.KCMAC_KEY_INTEG O.SESSION_KEYS_CONFID O.SESSION_KEYS_INTEG
O.PIN-MNGT	The TOE shall provide a means to securely manage PIN objects (including the PIN try limit, PIN try counter and states). If the PIN try limit is reached, no further PIN authentication must be allowed. See #.PIN-MNGT for details. Application Note: PIN objects may play key roles in the security architecture of client applications. The way they are stored and managed in the memory of the smart card must be carefully considered, and this applies to the whole object rather than the sole value of the PIN. For instance, the try limit and the try counter's value are as sensitive as that of the PIN and the TOE must restrict their modification only to authorized applications such as the card manager.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.TRANSACTION	The TOE must provide a means to execute a set of operations atomically. See #.TRANSACTION for details.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.OBJ-DELETION	The TOE shall ensure the object deletion shall not break references to objects. See #.OBJ-DELETION for further details.	O.ATTESTATION_ON_DELETION
O.DELETION	The TOE shall ensure that both applet and CAP file deletion perform as expected. See #.DELETION for details.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.LOAD	The TOE shall ensure that the loading of a CAP file into the card is safe. Besides, for code loaded post-issuance, the TOE shall verify the integrity and authenticity evidences generated during the verification of the application CAP file by the verification authority. This verification by the	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.

Platform Toe Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
	TOE shall occur during the loading or later during the install process. Application Note: Usurpation of identity resulting from a malicious installation of an applet on the card may also be the result of perturbing the communication channel linking the CAD and the card. Even if the CAD is placed in a secure environment, the attacker may try to capture, duplicate, permute or modify the CAP files sent to the card. He may also try to send one of its own applications as if it came from the card issuer. Thus, this objective is intended to ensure the integrity and authenticity of loaded CAP files.	
O.INSTALL	The TOE shall ensure that the installation of an applet performs as expected (See #.INSTALL for details). Besides, for code loaded post-issuance, the TOE shall verify the integrity and authenticity evidences generated during the verification of the application CAP file by the verification authority. If not performed during the loading process, this verification by the TOE shall occur during the install process.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
O.SCP.IC	The SCP shall provide all IC security features against physical attacks. This security objective refers to the point (7) of the security aspect #.SCP: It is required that the IC is designed in accordance with a well-defined set of policies and Standards (likely specified in another protection profile), and will be tamper resistant to actually prevent an attacker from extracting or altering security data (like cryptographic keys) by using commonly employed techniques (physical probing and sophisticated analysis of the chip). This especially matters to the	O.OS_SUPPORT

Platform Toe Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
	management (storage and operation) of cryptographic keys.	
O.SCP.RECOVERY	If there is a loss of power, or if the smart card is withdrawn from the CAD while an operation is in progress, the SCP must allow the TOE to eventually complete the interrupted operation successfully, or recover to a consistent and secure state. This security objective refers to the security aspect #.SCP (1): The smart card platform must be secure with respect to the SFRs. Then after a power loss or sudden card removal prior to completion of some communication protocol, the SCP will allow the TOE on the next power up to either complete the interrupted operation or revert to a secure state.	O.RECOVERY
O.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: (2) It does not allow the TSFs to be bypassed or altered and does not allow access to other low-level functions than those made available by packages of the API. That includes the protection of its private data and code (against disclosure or modification) from the Java Card System. (3) It provides secure low-level cryptographic processing to the Java Card System. (4) It supports the needs for any update to a single persistent object or class field to be atomic, and possibly a low-level transaction mechanism. (5) It allows the Java Card System to store data in "persistent technology memory" or in volatile memory, depending on its needs (for instance, transient objects must not be stored in non-volatile memory). The memory model is structured	O.OS_SUPPORT

Platform Toe Security Objective Label	Platform Toe Security Objective Label Content	Linked Composite-Product Toe Security Objectives
	and allows for low-level control accesses (segmentation fault detection).	
From 'Sensitive Array' package		
O.SENSITIVE_ARRAYS_INTEG	The TOE shall ensure that only the currently selected applications may have a write access to the integrity-sensitive array object (javacard.framework.SensitiveArrays) created by that application. Any unauthorized modification through physical attacks to that integrity-sensitive array must be detected by the TOE and notified to the application.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
From 'Sensitive Result' package		
O.SENSITIVE_RESULTS_INTEG	The TOE shall ensure that the sensitive results (javacardx.security.SensitiveResults) of sensitive operations executed by applications through the Java Card API are protected in integrity specifically against physical attacks.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
From 'Monotonic Counters' package		
O.MTC-CTR-MNGT	The TOE shall provide a means to securely manage value of the monotonic counter. This concerns the optional package javacardx.security.util of the Java Card platform.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.
From 'Cryptographic Certificate Management' package		
O.CRT-MNGT	The TOE shall provide a means to securely manage cryptographic certificates. This concerns the optional package javacardx.security.cert of the Java Card platform.	No direct link with the composite product security objectives, but this platform security objective is used to endure the security of the composite TOE.

Table 21 Statement of compatibility – TOE Objectives part from [PP-JCS] Protection Profile

5.4.2.STATEMENT OF COMPATIBILITY – ENV OBJECTIVES PART

The following table lists the relevant ENV security objectives of the security target [ST_PF], and provides the link to the composite-product, showing that they have been taken into account and that no contradiction has been introduced.

– IrOE: The objectives for the environment being not relevant for the Composite-ST.

– CfPOE: 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 group IrOE nor CfOE.

5.4.2.1. FROM [PP-GP] PROTECTION PROFILE

Platform ENV Security Objective Label	Platform ENV Security Objective Label Content	IrOE	CfP OE	SgO E	Linked Composite-Product TOE ENV Security Objectives
From core part					
OE.ISSUER	The Issuer shall be a trusted actor responsible for the behaviour of the SE.		X		OE.PRODUCTION_ENV
OE.ADMIN	The administrators of the CCM servers (e.g. OTA or other kinds of servers) shall be trusted actors. They shall be trained to use and administrate those servers. They have the means and the equipment to perform their tasks. They must be aware of the sensitivity of the assets they manage and the responsibilities associated with the administration of CCM servers. Administrators obey the security policies and constitute, by this OE, no source of an inside attack.		X		OE.PRODUCTION_ENV
OE.APPS-PROVIDER	The AP shall be a trusted actor that provides applications. The AP must be responsible for the APSD keys.		X		OE. SECURE_DEVELOPMENT_AND_PRO DUCTION OE.APPS_VALIDATION
OE.VERIFICATION-AUTHORITY	The VA shall be a trusted actor with the capability to check and validate the digital signature attached to an application.		X		OE.DEVICE_OEM
OE.KEY-ESCROW	The key escrow shall be a trusted actor in charge of the secure storage of the AP initial keys generated by the personaliser.		X		
OE.PERSONALISER	The personaliser shall be a trusted actor in charge of the personalisation process. The personaliser shall ensure the security of the keys managed and loaded into the card: Issuer Security Domain keys (ISD keys) Application Provider Security Domain keys (APSD keys) Controlling Authority Security Domain keys (CASD keys).		X		OE.PRODUCTION_ENV
OE.CONTROLLING-AUTHORITY	The CA shall be a trusted actor responsible for securing the creation and personalisation of APSD keys. The CA must be responsible for the CASD keys.		X		
OE.SCP-SUPP	Secure Communication Protocols shall be supported and used by the operational environment.		X		OE. DK_PROTOCOL_SECURITY OE.COMMUNICATION
OE.KEYS-PROT	During the TOE's use, the terminal in interaction with the TOE shall ensure the protection (integrity and confidentiality) of the applied keys by operational means and/or procedures.		X		OE.VEHICLE_ROOT_KEY_CONFID
OE.PRODUCTION	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 data (to prevent any		X		OE.PRODUCTION_ENV

Platform ENV Security Objective Label	Platform ENV Security Objective Label Content	IrOE	CfPOE	SgOE	Linked Composite-Product TOE ENV Security Objectives
	possible copy, modification, retention, theft, or unauthorised use).				
OE.APPLICATIONS	Developers and Validators shall comply with the security guidance and ensure that the rules are enforced.		X		OE. SECURE_DEVELOPMENT_AND_PRO DUCTION
OE.AID-MANAGEMENT	The VA shall verify that the AID of the application being loaded does not impersonate the AID known by another application on the card for the use of shareable services.		X		OE.APPS_VALIDATION
OE.LOADING	Application code, validated or certified depending on the application, is loaded onto the SE Platform using any kind of CCM servers (e.g. OTA or other kinds of servers used to perform card content management) and protocols with contactless or contact (e.g. USB) connectivity.		X		OE.APPS_VALIDATION
OE.SERVERS	The Issuer must enforce a policy to ensure the security of the applications stored on its CCM servers (e.g. OTA or other kinds of servers used to perform card content management).		X		OE.OEM_SERVERS
OE.AP-KEYS	The SD-key-personaliser, the AP, and the key escrow must enforce a security policy securing the transmissions.			X	OE. SECURE_DEVELOPMENT_AND_PRO DUCTION
OE.ISD-KEYS	The security of the ISD keys must be ensured in the environment of the TOE.		X		OE. SECURE_DEVELOPMENT_AND_PRO DUCTION
OE.KEY-GENERATION	The personaliser must ensure that the generated keys cannot be accessed by unauthorised users.		X		OE.KEY_SHARE
OE.CA-KEYS	The CASD keys must be securely generated prior to storage in the SE card.	X			
OE.KEY-CHANGE	The AP must change the initial keys of APSD before any operation on it.		X		
From package 'Ciphered Load File Data Block (CLFDB)'					
OE.CLFDB-ENC-PR	The Load File Data Block shall be encrypted securely by a trusted SD provider. Application Note: See [GPCS] section C.6.	X			Not relevant for this composite product security objectives. The mechanism is fully managed by the platform.
OE.CLFDB-ENC-PR	The Load File Data Block shall be encrypted securely by a trusted SD provider. Application Note: See [GPCS] section C.6.	X			Not relevant for this composite product security objectives. The mechanism is fully managed by the platform.
From package 'Delegated Management (DM)'					
OE.TOKEN-GEN	The Token shall be generated securely by a trusted entity according to the signature algorithms defined in GlobalPlatform specifications. Application Note: See [GPCS] sections B.1, B.2, B.3, B.4, and C.4.	X			Not relevant for this composite product security objectives. The mechanism is fully managed by the platform.
OE.RECEIPT-VER	The Receipt shall be verified securely by a trusted entity according to the methods defined in GlobalPlatform specifications. Application Note: See [GPCS] sections B.1, B.2, B.3, B.4, and C.5.	X			Not relevant for this composite product security objectives. The mechanism is fully managed by the platform.
From packages 'DAP Verification' and 'Mandated DAP Verification'					
OE.DAP_BLOCK_GEN	The DAP Block shall be generated securely by a trusted entity that	X			Not relevant for this composite product security objectives. The

Platform ENV Security Objective Label	Platform ENV Security Objective Label Content	IrOE	CfPOE	SgOE	Linked Composite-Product TOE ENV Security Objectives
	verifies the content of the Load File Data Block linked to the hash.				DAP mechanism is fully managed by the platform.
From PP-Module 'OS Update'					
OE.OS-UPDATE-EVIDENCE	For additional code loaded pre issuance, evaluated technical measures implemented by the TOE or audited organisational measures must ensure that the additional code (1) has been issued by the genuine OS Developer and (2) has not been altered since it was issued by the genuine OS Developer. For additional code loaded post issuance, the OS Developer shall provide digital evidence to the TOE that (1) he is the genuine developer of the additional code and (2) the additional code has not been modified since it was issued by the genuine OS Developer.	X			Not relevant for this composite product security objectives. The OS agility mechanism is fully managed by the platform.
OE.OS-UPDATE-ENCRYPTION	For additional code loaded post issuance, the OS Developer shall encrypt the additional code so that its confidentiality is ensured when it is transmitted to the TOE for loading and installation.	X			Not relevant for this composite product security objectives. The OS agility mechanism is fully managed by the platform.
OE.SECURE_ACODE_MANAGEMENT	Key management processes related to the OS Update capability shall take place in a secure and audited environment. The key generation processes shall guarantee that cryptographic keys are of sufficient quality and appropriately secured to ensure confidentiality, authenticity, and integrity of the keys.	X			Not relevant for this composite product security objectives. The OS agility mechanism is fully managed by the platform.

Table 22 Statement of compatibility – ENV Objectives part from [PP-GP] Protection Profile

5.4.2.2. FROM [PP-JCS] PROTECTION PROFILE

Platform ENV Security Objective Label	Platform ENV Security Objective Label Content	IrOE	CfPOE	SgOE	Linked Composite-Product TOE ENV Security Objectives
OE.CAP_FILE	No CAP file loaded post-issuance shall contain native methods.	X			The mDKA applet doesn't contain native methods.
OE.VERIFICATION	All the bytecodes shall be verified at least once, before the loading, before the installation or before the execution, depending on the card capabilities, in order to ensure that each bytecode is valid at execution time. See #.VERIFICATION for details. Additionally, the applet shall follow all the recommendations, if any, mandated in the platform guidance for maintaining the isolation property of the platform.		X		OE.PRODUCTION_ENV

	Application Note: constraints to maintain the isolation property of the platform are provided by the platform developer in application development guidance. The constraints apply to all application code loaded in the platform.				
OE.CODE-EVIDENCE	For application code loaded pre-issuance, evaluated technical measures implemented by the TOE or audited organizational measures must ensure that loaded application has not been changed since the code verifications required in OE.VERIFICATION. For application code loaded post-issuance and verified off-card according to the requirements of OE.VERIFICATION, the verification authority shall provide digital evidence to the TOE that the application code has not been modified after the code verification and that he is the actor who performed code verification. For application code loaded post-issuance and partially or entirely verified on-card, technical measures must ensure that the verification required in OE.VERIFICATION are performed. On-card bytecode verifier is out of the scope of this Protection Profile. Application Note: for application code loaded post-issuance and verified off-card, the integrity and authenticity evidence can be achieved by electronic signature of the application code, after code verification, by the actor who performed verification.		X		OE.PRODUCTION_ENV

Table 23 Statement of compatibility – ENV Objectives part from [PP-JCS] Protection Profile

6. EXTENDED COMPONENTS DEFINITION

6.1. EXTENDED COMPONENT FCS_RNG.1

To define the IT security functional requirements of the TOE an additional family (FCS_RNG) of the Class FCS (cryptographic support) is defined here. This family describes the functional requirements for random number generation used for cryptographic purposes.

This family also defines quality requirements for the generation of random numbers which are intended to be used for cryptographic purposes.

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] 3 that implements: [assignment: list of security capabilities].**
FCS_RNG.1.2	**The TSF SHALL provide random numbers that meet [assignment: a defined quality metric].**

7. SECURITY REQUIREMENTS

7.1. TYPOGRAPHICAL CONVENTIONS

The following conventions are used in the definitions of the SFRs:

- Selections, assignments and refinements that have already been made in the [PP-CCC-CP-023] Protection Profiles are **in bold**, and the original text on which the selection, assignment or refinement has been made is not reminded.
- Selections, assignments and refinements made in this ST are **in bold and underlined**, and the PP original text on which the selection or assignment has been made is indicated in a footnote.
- Iteration operations on SFR components are denoted by showing a slash "/" and the iteration indicator after the SFR component identifier. This may lead to differences with [PP-CCC-CP-023] architecture.

7.2. SECURITY FUNCTIONAL REQUIREMENTS

7.2.1. CRYPTOGRAPHIC KEY MANAGEMENT

Cryptographic keys SHALL be managed throughout their life cycle. The following SFRs are intended to support that lifecycle and consequently defines requirements for the following activities:

- cryptographic key generation,
- cryptographic key distribution and
- cryptographic key destruction.

FCS_CKM.1/ECC - Cryptographic key generation | EC Point Generation

FCS_CKM.1.1/ECC	The TSF SHALL generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **ECC with P-256 (SECP256r1)** and specified cryptographic key sizes **256-bit** that meet the following standards: **[FIPS PUB 186-4]**[3]

Application Note 8 :

- **FCS_CKM.1/Keys_Crypto is used in the rest of the document to cover all the FCS_CKM.1 iterations stated in this Security Target (FCS_CKM.1.1/ECC, FCS_CKM.1.1/Session_keys, FCS_CKM.1.1/Long_Term_key, FCS_CKM.1.1/Secret_Shared_key).**

FCS_CKM.1/Session_keys - Cryptographic key generation | Secure Channel

[3] **[Selection: [BSI TR-03111], ANSI X9.62 [X9.62a], ANSI X9.63 [X9.63], [FIPS PUB 186-4]]**

FCS_CKM.1.1/Session_keys	The TSF SHALL generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **HKDF-SHA-256** and specified cryptographic key sizes **256-bit** that meet the following standards: **IETF [RFC 5869]**

FCS_CKM.1/Long_Term_key - Cryptographic key generation | Secure Channel

FCS_CKM.1.1/Long_Term_key	The TSF SHALL generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **HKDF-SHA-256** and specified cryptographic key sizes **256-bit** that meet the following standards: **IETF [RFC 5869]**

FCS_CKM.1/Secret_Shared_key - Cryptographic key generation | Secure Channel

FCS_CKM.1.1/Secret_Shared_key	The TSF SHALL generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **HKDF-SHA-256** and specified cryptographic key sizes **256-bit** that meet the following standards: **IETF [RFC 5869]**

FCS_RNG.1 Random number generation

FCS_RNG.1.1	The TSF SHALL provide a [**hybrid deterministic**][4] random number generator [**DRG.4**][5] that **implements: generation of strong cryptographic random numbers, key generation functions use adequate entropy source from approved random number generator(s).**
FCS_RNG.1.2	The TSF shall provide random numbers that meet [**AIS31**][6].

FCS_CKM.2/ECDHE - Cryptographic key distribution | Key Establishment

[4] **[selection: physical, non-physical true, deterministic, hybrid physical, hybrid deterministic]**
[5] **Refer to [AIS20] or [AIS31]**
[6] **[assignment: a defined quality metric]**

FCS_CKM.2.1/ECDHE	The TSF SHALL distribute cryptographic keys in accordance with a specified cryptographic key distribution method **Elliptic curve-based Diffie-Hellman Ephemeral key agreement and cryptographic key sizes 256-bit** that meets the following: **NIST Special Publication [NIST-SP-800-56A]Revision 3 with approved groups from Appendix D**.

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.4.1	The TSF SHALL destroy cryptographic keys in accordance with a specified cryptographic key destruction method [**pseudo-random pattern**][7] that meets the following: **None**.

7.2.2.CRYPTOGRAPHIC OPERATION (FCS_COP)

In order for a cryptographic operation to function correctly, the operation SHALL be performed in accordance with a specified algorithm and with a cryptographic key of a specified size. The following SFRs specify all this latter information to be enforced by the TSF.

It covers the following:

- data encryption and/or decryption,
- digital signature generation and/or verification,
- cryptographic checksum generation for integrity and/or verification of checksum,
- secure hash (message digest),
- cryptographic key encryption and/or decryption,
- and cryptographic key agreement.

FCS_COP.1 is used in the rest of the document to cover all the FCS_COP.1 iterations stated in this Security Target.

FCS_COP.1/Hash - Cryptographic operation

FCS_COP.1.1/Hash	The TSF SHALL perform **Cryptographic Hashing** in accordance with a specified cryptographic algorithm **SHA-256** and cryptographic key sizes **None** that meet the following: **[FIPS 180-4]**[8].

[7] **[selection: zeroes, ones, pseudo-random pattern, a new value of a key of the same size, a value that does not contain any security attribute]**

[8] **[selection: ISO/IEC 10118-3:2018, FIPS 180-4]**

FCS_COP.1/HMAC Cryptographic operation

FCS_COP.1.1/HMAC	The TSF SHALL perform **Keyed Hash Message Authentication** in accordance with a specified cryptographic algorithm **HMAC-SHA256**, and cryptographic key sizes **256-bit** that meet the following: **ISO/IEC 9797-2:2011**.

FCS_COP.1/Encryption/decryption - Cryptographic operation

FCS_COP.1.1/Encryption/decryption	The TSF SHALL perform **data encryption or decryption** in accordance with a specified cryptographic **algorithm AES with CBC mode of operation** and cryptographic key sizes **128-bit** that meet the following: **[FIPS PUB 197], [NIST SP 800-38A]**.

FCS_COP.1/CMAC - Cryptographic operation

FCS_COP.1.1/CMAC	The TSF SHALL perform **Message Authentication Code** in accordance with a specified cryptographic algorithm **AES CMAC** and cryptographic key sizes **128-bit** that meet the following: **[NIST SP 800-38B]**.

FCS_COP.1/ECDSA - Cryptographic operation

FCS_COP.1.1/ECDSA	The TSF SHALL perform **digital signing** accordance with a specified cryptographic algorithm **ECDSA with NIST P-256 curve** and cryptographic key sizes **256-bit** that meet the following: **[ANSI X9.62]**.

7.2.3.ACCESS CONTROL POLICY | SECURITY DOMAIN (FDP_ACC)

The following SFR defines the Security Functional Policy for access control to the TOE, which will be called SD_SFP. For better readability SD_FSP is defined in the following table and the SFRs will refer to it:

TYPE	SHORT NAME	DEFINITION
Subjects[9]	S.INSTALLER, (from [PP-JCS])	The installer is the on-card entity which acts on behalf of the card issuer. This subject is involved in the loading of packages and installation of applets
	S.CAD (from [PP-JCS])	The CAD represents off-card entity that communicates with the S.INSTALLER
	S.SD	SD stands for Security Domain and here S.SD can be representing an off-card entity on the card such as a validation authority, application provider etc.
Objects	O.Load_File	DK Applet Load file or Executable File, in case of application loading, installation, extradition or registry update, with a set of intended privileges and its targeted associated SD AID.
	O.Delegation_Token	The Delegation Token, in case of Delegated Management operations, with the attributes Present or Not Present;
	O.DAP	The DAP Block, in case of application loading, with the attributes Present or Not Present;
Operations	O.GP_CCM	GlobalPlatform's card content management commands
	O.API	API methods
Rules	R_GPF	Runtime behavior rules defined by GlobalPlatform [GPCS] for: • loading (Section 9.3.5 of [GPCS]); • installation (Section 9.3.6 of [GPCS]); • extradition (Section 9.4.1 of [GPCS]); • registry update (Section 9.4.2 of [GPCS]); • content removal (Section 9.5 of [GPCS]).

Table 24 Access control SFP - SD_SFP

FDP_ACC.2 Complete access control

FDP_ACC.2.1	The TSF SHALL enforce the **SD_SFP** on all subjects, objects defined by the **SD_FSP** and all operations among subjects and objects covered by the SFP.
FDP_ACC.2.2	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

[9] Subjects are active components of the TOE that (essentially) act on the behalf of users. The users of the TOE include people or institutions (like the applet developer, the card issuer, the verification authority), hardware and software components (like the application packages installed on the card). Some of the users may just be aliases for other users. For instance, the verification authority in charge of the bytecode verification of the applications may be just an alias for the card issuer.

7.2.4.ACCESS CONTROL FUNCTIONS | SECURITY DOMAIN (FDP_ACF)

FDP_ACF.1 Security attribute base access control

FDP_ACF.1.1	The TSF SHALL enforce the **SFP_AC** to objects based on the following: **All subjects and objects together with their respective security attributes as defined in SD_SFP**.
FDP_ACF.1.2	The TSF SHALL enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **Rules for all access methods and access rules defined in SD_SFP**.
FDP_ACF.1.3	The TSF SHALL explicitly authorize access of subjects to objects based on the following additional rules: **none**[10].
FDP_ACF.1.4	The TSF SHALL explicitly deny access of subjects to objects based on the following additional rules: **when at least one of the rules R_GPF defined in the SD_SFP does not hold**

Application Note 9 : The dependency FMT_MSA.3 will not be fulfilled here, since there is no initialization of attributes necessary.

7.2.5.INFORMATION FLOW CONTROL POLICY | SECURE CHANNEL PROTOCOL (FDP_IFC)

TYPE	SHORT NAME	DEFINITION
Subjects[11]	S.INSTALLER, (from [PP-JCS])	The installer is the on-card entity which acts on behalf of the card issuer. This subject is involved in the loading of packages and installation of applets
	S.CAD (from [PP-JCS])	The CAD represents off-card entity that communicates with the S.INSTALLER
	S.SD (from [PP-JCS])	SD stands for Security Domain and here S.SD can be representing an off-card entity on the card such as a

[10] [assignment: rules, based on security attributes, that explicitly authorise access of subjects to objects]

[11] Subjects are active components of the TOE that (essentially) act on the behalf of users. The users of the TOE include people or institutions (like the applet developer, the card issuer, the verification authority), hardware and software components (like the application packages installed on the card). Some of the users may just be aliases for other users. For instance, the verification authority in charge of the bytecode verification of the applications may be just an alias for the card issuer.

		validation authority, application provider etc.
Information	I.CCM	The information controlled by this policy is the card content management command, including personalization commands, in the APDUs sent to the card and their associated responses returned to the CAD.
Operations	O.GP_CCM	GlobalPlatform's card content management commands
	O.API	API methods
Rules	R_GPF	Runtime behavior rules defined by GlobalPlatform [GPCS] for: • loading (Section 9.3.5 of [GPCS]); • installation (Section 9.3.6 of [GPCS]); • extradition (Section 9.4.1 of [GPCS]); • registry update (Section 9.4.2 of [GPCS]); • content removal (Section 9.5 of [GPCS]).

Table 25 Information Flow Control SFP - SC_SFP

FDP_IFC.2 Complete Information Flow Control

FDP_IFC.2.1/SCP	The TSF SHALL enforce the **SCP_SFP** on **subjects, information and operations** and all operations that cause that information to flow to and from subjects covered by the SCP_SFP.
FDP_IFC.2.2/SCP	The TSF SHALL ensure that all operations that cause any information in the TOE to flow to and from any subject in the TOE are covered by an information flow control SFP.

7.2.6. INFORMATION FLOW CONTROL FUNCTIONS | SECURE CHANNEL PROTOCOL (FDP_IFF)

FDP_IFF.1 - Simple security attributes

FDP_IFF.1.1/SCP	The TSF SHALL enforce the **SCP_SFP** based on the following types of subject and information security attributes: **Subjects and information as defined by the SCP_SFP, and for each, the security attributes as defined in** [GPCS] **and no other additional security attributes**[12].
FDP_IFF.1.2/SCP	The TSF SHALL permit an information flow between a controlled subject and controlled information via a controlled operation if the

[12] **[assignment: list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP].**

	following rules hold: **Rules R_GPF as defined by the SCP_SFP**.
FDP_IFF.1.3/SCP	The TSF SHALL enforce the **no additional information rules**[13].
FDP_IFF.1.4/SCP	The TSF SHALL explicitly authorise an information flow based on the following rules: **no additional information rules**[14].
FDP_IFF.1.5/SCP	The TSF SHALL explicitly deny an information flow based on the following rules: **When none of the conditions listed in the element FDP_IFF.1.4 of this component hold and at least one of those listed in the element FDP_IFF.1.2 does not hold**.

Application Note 10 : The on-card and the off-card subjects have security attributes such as MAC, Cryptogram, Challenge, Key Set, Static Keys, etc.

Application Note 11 : An SFR FMT_MSA.3 is not used here, since the security attributes used in the SCP_SFP are already contained in the I.CCM when entering the TOE, therefore rules for creation of information and default values of security attributes are not applicable

7.2.7.RESIDUAL INFORMATION PROTECTION (FDP_RIP)

FDP_RIP.1 Subset residual information protection

FDP_RIP.1.1	The TSF SHALL ensure that any previous information content of a resource is made unavailable upon **the deallocation of the resource from** the following objects: **Cryptographic buffers**.

7.2.8.STORED DATA INTEGRITY (FDP_SDI)

FDP_SDI.2 Stored data integrity monitoring and action

FDP_SDI.2.1	The TSF SHALL monitor user data stored in containers controlled by the TSF for **integrity errors** [15]on all objects, based on the following attributes: **Integrity check data** [16].
FDP_SDI.2.2	Upon detection of a data integrity error, the TSF SHALL **prohibit the use of the altered data, send notification of the error where applicable**.

[13] **[assignment: additional information flow control SFP rules]**
[14] **[assignment: rules, based on security attributes, that explicitly authorise information flows]**
[15] **[assignment: *integrity errors*]**
[16] **[assignment: *user data attributes*]**

7.2.9.INTER-TSF USER DATA INTEGRITY TRANSFER PROTECTION (FDP_UIT)

FDP_UIT.1 Data exchange Integrity | Card Content Management

FDP_UIT.1.1/CCM	The TSF SHALL enforce the **Secure channel protocol Information flow control policy** to **transmit, receive**[17] user data in a manner protected from **modification, deletion, insertion, replay**[18] errors.
FDP_UIT.1.2/CCM	The TSF SHALL be able to determine on receipt of user data, whether **modification, deletion, insertion, replay** has occurred.

7.2.10. IDENTIFICATION AND AUTHENTICATION (FIA_UAU)

FIA_UAU.3 Unforgeable authentication

FIA_UAU.3.1	The TSF SHALL **detect** [19]use of authentication data that has been forged by any user of the TSF.
FIA_UAU.3.2	The TSF SHALL **detect** [20]use of authentication data that has been copied from any other user of the TSF.

7.2.11. SECURITY MANAGEMENT | TSF DATA (FMT_MTD)

FMT_MTD.1 Management of TSF data

FMT_MTD.1.1/deletion of keys	**The TSF SHALL restrict the ability to**, **delete**, the **keys** to **Digital Key framework**, **Digital Key Applet** [21] .

FMT_MTD.3 Secure TSF data

FMT_MTD.3.1	The TSF SHALL ensure that only secure values are accepted **for the applet's AID**.

[17] **[selection: *transmit, receive*]**
[18] **[selection: *modification, deletion, insertion, replay*]**
[19] **[selection: *detect, prevent*]**
[20] **[selection: *detect, prevent*]**
[21] **[assignment: the authorized identified roles]**

Application Note 12 : The value of the Applet's AID is defined in [CCC-TS-101], Section 15.3.2.1.

7.2.12. SPECIFICATIONS OF MANAGEMENT FUNCTIONS | TSF DATA (FMT_SMF)

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1	The TSF SHALL be capable of performing the following management functions: **creates a deletion attestation for the requested key (for deletion), and that it is securely deleted before the attestation is transferred to the requesting party**.

7.2.13. SECURITY MANAGEMENT ROLES (FMT_SMR)

FMT_SMR.1 Security Roles

FMT_SMR.1.1	The TSF SHALL maintain the roles **Digital Key framework, Vehicle**, **no other authorized identified roles** [22] .
FMT_SMR.1.2	The TSF SHALL be able to associate users with roles.

7.2.14. UNLINKABILITY (FPR_UNL)

FPR_UNL.1 Unlinkability

FPR_UNL.1.1/NFC	**The TSF shall ensure that any entity (other than the TOE, the DK Framework or the Vehicle) is unable to determine whether data and key exchanged over NFC (between the TOE and the Vehicle) were caused by the same user.**

7.2.15. PROTECTION OF THE TSF (FPT_ITC)

FPT_ITC.1 Inter-TSF confidentiality during transmission

FPT_ITC.1.1	The TSF SHALL protect all TSF data transmitted from the TSF to another trusted IT product from unauthorized disclosure during transmission.

[22] **[assignment: the authorized identified roles]**

FPT_ITI.1/Vehicle_Integrity - Inter-TSF detection of modification

FPT_ITI.1.1/Vehicle_Integrity	The TSF SHALL provide the capability to detect modification of all TSF data during transmission between the TSF and another trusted IT product within the following metric: **metric as defined in FCS_COP.1.1/CMAC**.
FPT_ITI.1.2/Vehicle_Integrity	The TSF SHALL provide the capability to verify the integrity of all TSF data transmitted between the TSF and another trusted IT product and perform **terminate the on-going process** if modifications are detected.

7.2.16. INTERNAL TOE TSF DATA TRANSFER (FPT_ITT)

FPT_ITT.1 is used in the rest of the document to cover all the FPT_ITT.1 iterations stated in this Security Target.

FPT_ITT.3 is used in the rest of the document to cover all the FPT_ITT.3 iterations stated in this Security Target.

FPT_ITT.1/IMMO_TOKEN - Basic internal TSF data transfer protection

FPT_ITT.1.1/IMMO_TOKEN	The TSF SHALL protect TSF data from **disclosure** when it is transmitted between separate parts of the TOE.

FPT_ITT.1/DIGITAL_KEY - Basic internal TSF data transfer protection

FPT_ITT.1.1/DIGITAL_KEY	The TSF SHALL protect TSF data from **disclosure** when it is transmitted between separate parts of the TOE.

FPT_ITT.3/DIGITAL_KEY - TSF data integrity monitoring

FPT_ITT.3.1/DIGITAL_KEY	The TSF SHALL be able to detect **modification of data**[23] for TSF data transmitted between separate parts of the TOE.
FPT_ITT.3.2/DIGITAL KEY	Upon detection of a data integrity error, the TSF SHALL take the following actions: **decrease the global fault detection counter. Once the global fault detection counter reaches 0, the card is put in degraded mode.** [24].

FPT_ITT.3/IMMO_TOKEN - TSF data integrity monitoring

FPT_ITT.3.1/IMMO_TOKEN	The TSF SHALL be able to detect **modification of data**[25] for TSF data transmitted between separate parts of the TOE.
FPT_ITT.3.2/IMMO_TOKEN	Upon detection of a data integrity error, the TSF SHALL take the following actions: **decrease the global fault detection counter. Once the global fault detection counter reaches 0, the card is put in degraded mode.**[26].

7.2.17. REPLAY DETECTION (FPT_RPL)

The following SFR uses the Subject defined hereafter: S.Transaction data: This can be the data/keys being shared between the DK Applet and the vehicle (through NFC) or between the DK Framework and the DK Applet.

FPT_RPL.1 Replay detection

FPT_RPL.1.1	The TSF SHALL detect replay for the following entities: **S.Transaction data**
FPT_RPL.1.2	The TSF SHALL perform **terminate the transaction** when replay is detected.

[23] **[selection: *modification of data, substitution of data, re-ordering of data, deletion of data*, [assignment: *other integrity errors*]]**
[24] **[assignment: *specify the action to be taken*]**
[25] **[selection: *modification of data, substitution of data, re-ordering of data, deletion of data*, [assignment: *other integrity errors*]]**
[26] **[assignment: *specify the action to be taken*]**

7.2.18. TRUSTED RECOVERY (FPT_RCV)

FPT_RCV.2 Automated recovery

FPT_RCV.2.1	When automated recovery from **power failure** is not possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.
FPT_RCV.2.2	For **power loss**, the TSF SHALL ensure the return of the TOE to a secure state using automated procedures.

7.2.19. INTER-TSF TRUSTED CHANNEL (FTP_ITC)

FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1	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	The TSF SHALL permit **another trusted IT product** to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF SHALL initiate communication via the trusted channel for **sharing of secret keys, user data, immobiliser token**.

7.2.20. PHYSICAL RESISTANCE (FPT_PHP)

FPT_PHP.3 Resistance to physical attack

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.

7.3. SECURITY ASSURANCE REQUIREMENTS

This ST is based on the EAL4 assurance package augmented with the components AVA_VAN.5, ALC_DVS.2 and ALC_FLR.1.

7.4. SECURITY REQUIREMENTS RATIONALE

7.4.1. SECURITY FUNCTIONAL REQUIREMENTS RATIONALE

Security Objectives	SFRs	Rationale
O.SE_MANAGEMENT	FDP_UIT.1 FDP_ACC.2 FDP_ACF.1 FMT_SMF.1 FMT_SMR.1 FDP_IFC.2 FDP_IFF.1 FCS_CKM.1/Keys_Crypto FCS_CKM.2/ECDHE, FCS_CKM.4, FDP_RIP.1, FMT_MTD.3	The Security Objective O.SE_MANAGEMENT is met by the following SFR's: ▪ FDP_UIT.1 which enforces the Secure Channel Protocol information flow control policy and the Security Domain access control policy to ensure the integrity of card management operations. ▪ All SFRs related to Security Domains (FDP_ACC. 2, FDP_ACF. 1, FMT_SMF.1, FCS_CKM.1/Keys_Crypto, FCS_CKM.2/ECDHE, FCS_CKM.4, FDP_RIP.1, FMT_MTD.3, FMT_SMR. 1(as an SFR-supporting)) cover this security objective by enforcing a Security Domain access control policy (rules and restrictions) that ensures a secure card content management. ▪ All SFRs related to the secure channel (FDP_IFC.2, FDP_IFF.1) support this security objective by enforcing Secure Channel Protocol information flow control policy that ensures the integrity and the authenticity of card management operations.
O.IMMO_TOK_CONFID	FPT_ITT.1, FTP_ITC.1, FPT_PHP.3	The Security Objective O.IMMO_TOK_CONFID is met by the following SFR's: ▪ FPT_ITT.1 which ensures that the data is protected when transmitted between separate parts of the TOE against disclosure, thus ensuring the confidentiality of immobilizer token. ▪ FTP_ITC.1 which requires that the TSF provide a trusted communication channel between itself and another trusted IT product, which will further ensure the confidentiality of the immobilizer token being transmitted. ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical probing.
O.IMMO_TOK_INTEG	FPT_ITT.3	The Security Objective O.IMMO_TOK_INTEG is met by the following SFR's:

Security Objectives	SFRs	Rationale
		▪ FPT_ITT.3 which enforces that the immobilizer token transmitted between separate parts of the TOE is monitored for identified integrity errors. ▪ FPT_ITT.3 which enforces the actions to be taken in the event of an integrity violation detection.
O.DK_CONFID	FPT_ITT.1, FPT_PHP.3	The security Objective O.DK_CONFID is met by the following SFR's: ▪ FPT_ITT.1 which ensures that the secret elements of the Digital Key are protected when transmitted between separate parts of the TOE against disclosure. ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical probing.
O.DK_INTEG	FPT_ITT.3 FDP_SDI.2	The Security Objective O.DK_INTEG is met by the following SFR's: ▪ FPT_ITT.3 which enforces that the assets of the Digital Key transmitted between separate parts of the TOE are monitored for identified integrity errors. FDP_SDI.2 ensures that the user data imported into the TOE are monitored for integrity violations
O.LONG_TERM_KEY_CONFID	FPT_PHP.3, FPT_ITT.1	The Security Objective O.LONG_TERM_KEY_CONFID is met by the following SFR's: ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical manipulation and physical probing. ▪ FPT_ITT.1 which ensures that the Long Term key is protected when transmitted between separate parts of the TOE against disclosure.
O.LONG_TERM_KEY_INTEG	FDP_SDI.2, FPT_PHP.3	The Security Objective O.LONG_TERM_KEY_INTEG is met by the following SFR's: ▪ FDP_SDI.2 which monitors stored user data for integrity errors. ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical manipulation and physical probing.
O.SEC_SHARED_KEY_CONFID	FPT_PHP.3, FPT_ITT.1	The Security Objective O.SEC_SHARED_KEY_CONFID is met by the following SFR's: ▪ FPT_PHP.3 which enforces the appropriate

Security Objectives	SFRs	Rationale
		mechanisms to continuously counter physical manipulation and physical probing. ▪ FPT_ITT.1 which ensures that the Secret shared key is protected when transmitted between separate parts of the TOE against disclosure.
O.SEC_SHARED_KEY_INTEG	FDP_SDI.2, FPT_PHP.3	The Security Objective O.SEC_SHARED_KEY_INTEG is met by the following SFR's: ▪ FDP_SDI.2 which monitors stored user data for integrity errors. ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical manipulation and physical probing.
O.KCMAC_KEY_CONFID	FPT_PHP.3 FPT_ITT.1	The Security Objective O.KCMAC_KEY_CONFID is met by the following SFR's: ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical manipulation and physical probing. ▪ FPT_ITT.1 which ensures that the Kcmac key is protected when transmitted between separate parts of the TOE against disclosure.
O.KCMAC_KEY_INTEG	FDP_SDI.2, FPT_PHP.3	The Security Objective O.KCMAC_KEY_INTEG is met by the following SFR's: ▪ FDP_SDI.2 which monitors stored user data for integrity errors. ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical manipulation and physical probing.
O.SESSION_KEYS_CONFID	FPT_PHP.3, FPT_ITT.1	The Security Objective O.SESSION_KEYS_CONFID is met by the following SFR's: ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical manipulation and physical probing. ▪ FPT_ITT.1 which ensures that the session keys are protected when transmitted between separate parts of the TOE against disclosure.
O.SESSION_KEYS_INTEG	FDP_SDI.2, FPT_PHP.3	The Security Objective O.SESSION_KEYS_INTEG is met by the following SFR's: ▪ FDP_SDI.2 which monitors stored user data for integrity errors. ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical

Security Objectives	SFRs	Rationale
		manipulation and physical probing.
O.ATTESTATION_ON_DELETION	FMT_SMF.1, FMT_MTD.1	The Security Objective O.ATTESTATION_ON_DELETION is met by the following SFR's: ▪ FMT_SMF.1 which defines the management functions concerning the attestation creation and secure transferring of the same during a deletion operation. ▪ FMT_MTD.1 which defines the management functions to be enforced and defines the concerned roles involved during a deletion operation.
O.RANDOMNESS	FCS_RNG.1	The Security Objective O.RANDOMNESS is met by FCS_RNG.1 which enforces the algorithms to be used for Random number generation and the entropy to be used based on certain standards.
O.IC_SUPPORT	FPT_PHP.3	The Security Objective O.IC_SUPPORT is met by the following SFR's: ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical manipulation and physical probing.
O.RECOVERY	FPT_RCV.2	The Security Objective O.RECOVERY is met by the following SFR's: ▪ FPT_RCV.2.1 which enforces the TOE to enter a maintenance mode where the ability to return to a secure state is provided, when automated recovery from certain failures is not possible. ▪ FPT_RCV.2.2 which enforces the return of the TOE to a secure state using automated procedures during certain failures which can occur as defined.
O.OS_SUPPORT	FPT_PHP.3	The security Objective O.OS_SUPPORT is met by the following SFR: ▪ FPT_PHP.3 which enforces the appropriate mechanisms to continuously counter physical manipulation and physical probing.
O.FAST_TRANSACTION_AUTH	FCS_COP.1/CMAC FPT_ITC.1 FPT_ITI.1/Vehicle_Integrity FPT_RPL.1 FTP_ITC.1	The Security Objective O.FAST_TRANSACTION_AUTH is met by the following SFR: ▪ FCS_COP.1/CMAC provides the MAC used to detect modifications. ▪ FPT_ITC.1 requires protection of TSF data from unauthorised disclosure during transmission

Security Objectives	SFRs	Rationale
		▪ FPT_ITI.1/Vehicle_Integrity requires that modifications to TSF data are detected when transmitted between the applet and vehicle ▪ FTP_ITC.1 which requires that the TSF provide a trusted communication channel between itself and the vehicle guaranteeing a secure Device authentication to the Vehicle. ▪ FPT_RPL.1 which protects against replay attacks over for such transactions
O.STD_TRANSACTION_AUTH	FCS_COP.1/CMAC FPT_ITC.1 FPT_ITI.1/Vehicle_Integrity FTP_ITC.1 FPT_RPL.1	The Security Objective O.STD_TRANSACTION_AUTH is met by the following SFR: ▪ FCS_COP.1/CMAC provides the MAC used to detect modifications. ▪ FPT_ITC.1 requires protection of TSF data from unauthorised disclosure during transmission ▪ FPT_ITI.1/Vehicle_Integrity requires that modifications to TSF data are detected when transmitted between the applet and vehicle ▪ FTP_ITC.1 which requires that the TSF provide a trusted communication channel between itself and the vehicle allowing the device to transmit data to the vehicle without any possibility of leakage by a passive or active eavesdropper and protecting the private assets from an MITM attack. ▪ FPT_RPL.1 which protects against replay attacks over for such transactions
O.KEY_EXCHANGE_AUTH	FIA_UAU.3 FCS_COP.1	The Security Objective O.KEY_EXCHANGE_AUTH is met by the following SFR's: ▪ FIA_UAU.3 which prevents and detects forged data which could be used for key exchange operation, guaranteeing the authenticity of the key exchange operation. ▪ FCS_COP.1 which ensures that the key exchange takes place accordance with a specified cryptographic algorithm & are based on defined standards.
O.NON-TRACEABILITY	FPR_UNL.1	The Security Objective O.NON-TRACEABILITY is met by the following SFR: ▪ FPR_UNL.1 enforces that any entity (other than the TOE, the DK Framework or the Vehicle) is unable to determine whether data and key exchanged over NFC (between the TOE and the Vehicle) were caused by the

Security Objectives	SFRs	Rationale
		same user.

Table 26 Security Objectives and SFRs

7.4.2.SFR DEPENDENCY RATIONALE

Security Functional Requirement	CC Dependencies	Satisfied Dependencies
FCS_CKM.1/Keys_Crypto	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/HMAC FCS_COP.1/ECDSA FCS_CKM.4
FCS_RNG.1	No dependencies	
FCS_CKM.2/ECDHE	FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1/Keys_Crypto FCS_CKM.4
FCS_CKM.4	[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.1/Keys_Crypto
FCS_COP.1	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1/Keys_Crypto FCS_CKM.4
FDP_ACC.2	FDP_ACF.1 Security attribute based access control	FDP_ACF.1
FDP_ACF.1	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.2 **Application Note 13 : The dependency FMT_MSA.3 will not be fulfilled here, since there is no initialization of attributes necessary.**
FDP_IFC.2	FDP_IFF.1 Simple security attributes	FDP_IFF.1
FDP_IFF.1	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation	FDP_IFC.1 See **Application Note 13**
FDP_RIP.1	No dependencies	
FDP_SDI.2	No dependencies	
FDP_UIT.1	[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 FTP_ITC.1
FIA_UAU.3	No dependencies	

Security Functional Requirement	CC Dependencies	Satisfied Dependencies
FMT_MTD.1	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMR.1 FMT_SMF.1
FMT_MTD.3	FMT_MTD.1 Management of TSF data	FMT_MTD.1
FMT_SMF.1	No dependencies	
FMT_SMR.1	FIA_UID.1 Timing of identification	No dependencies **Application Note 14 : The dependency to FIA_UID.1 is not applicable to this TOE. The PP does not require the identification of the roles to be assigned which is handled by the operational environment.**
FPR_UNL.1	No dependencies	
FPT_ITC.1	No dependencies	
FPT_ITI.1/Vehicle_Integrity	No dependencies	
FPT_ITT.1	No dependencies	
FPT_ITT.3	No dependencies	
FPT_RPL.1	No dependencies	
FPT_RCV.2	AGD_OPE.1 Operational user guidance	AGD_OPE.1
FTP_ITC.1	No dependencies	
FPT_PHP.3	No dependencies	

Table 27 SFRs dependency rationale

7.4.3.RATIONALE FOR THE EXCLUSION OF DEPENDENCIES

The dependency to FIA_UID.1 is not applicable to this TOE. The [PP-CCC-CP-023] does not require the identification of the roles to be assigned which is handled by the operational environment.

7.4.4.SECURITY ASSURANCE REQUIREMENT RATIONALE

The EAL4 package and addition of ALC_DVS.2, AVA_VAN.5 and ALC_FLR.1 are required by [PP-CCC-CP-023].

7.4.5.SAR DEPENDENCY RATIONALE

Security Assurance Requirement	CC Dependencies	Satisfied Dependencies
ADV_ARC.1	(ADV_FSP.1) and (ADV_TDS.1)	ADV_FSP.4 ADV_TDS.3
ADV_FSP.4	(ADV_TDS.1)	ADV_TDS.3
ADV_TDS.3	(ADV_FSP.4)	ADV_FSP.4
ADV_IMP.1	(ADV_TDS.3) and (ALC_TAT.1)	ADV_TDS.3 ALC_TAT.1
AGD_OPE.1	(ADV_FSP.1)	ADV_FSP.4

Security Assurance Requirement	CC Dependencies	Satisfied Dependencies
AGD_PRE.1	No dependencies	
ALC_CMC.4	(ALC_CMS.1) and (ALC_DVS.1) and (ALC_LCD.1)	ALC_CMS.4 ALC_DVS.2 ALC_LCD.1
ALC_CMS.4	No dependencies	
ALC_DEL.1	No dependencies	
ALC_DVS.2	No dependencies	
ALC_FLR.1	No dependencies	
ALC_LCD.1	No dependencies	
ALC_TAT.1	(ADV_IMP.1)	ADV_IMP.1
ASE_CCL.1	(ASE_ECD.1) and (ASE_INT.1) and (ASE_REQ.1)	ASE_ECD.1 ASE_INT.1 ASE_REQ.2
ASE_ECD.1	No dependencies	
ASE_INT.1	No dependencies	
ASE_OBJ.2	(ASE_SPD.1)	ASE_SPD.1
ASE_REQ.2	(ASE_ECD.1) and (ASE_OBJ.2)	ASE_ECD.1 ASE_OBJ.2
ASE_SPD.1	No dependencies	
ASE_TSS.1	(ADV_FSP.1) and (ASE_INT.1) and (ASE_REQ.1)	ADV_FSP.4 ASE_INT.1 ASE_REQ.2
ATE_COV.2	(ADV_FSP.2) and (ATE_FUN.1)	ADV_FSP.4 ATE_FUN.1
ATE_DPT.1	(ADV_ARC.1) and (ADV_TDS.2) and (ATE_FUN.1)	ADV_ARC.1 ADV_TDS.3 ATE_FUN.1
ATE_FUN.1	(ATE_COV.1)	ATE_COV.2
ATE_IND.2	(ADV_FSP.2) and (AGD_OPE.1) and (AGD_PRE.1) and (ATE_COV.1) and (ATE_FUN.1)	ADV_FSP.4 AGD_OPE.1 AGD_PRE.1 ATE_COV.2 ATE_FUN.1
AVA_VAN.5	(ADV_ARC.1) and (ADV_FSP.4) and (ADV_IMP.1) and (ADV_TDS.3) and (AGD_OPE.1) and (AGD_PRE.1) and (ATE_DPT.1)	ADV_ARC.1 ADV_FSP.4 ADV_IMP.1 ADV_TDS.3 AGD_OPE.1 AGD_PRE.1 ATE_DPT.1

Table 28 SAR dependency rationale

The table here-above shows that all SAR dependencies are met.

7.5. COMPOSITION TASKS – SFR PART

The following table lists the SFRs that are declared in the security target [ST_PF] and separates them in relevant platform[27]-SFRs (RP_SFR-SERV and RP_SFR-MECH[28]) and irrelevant platform-SFRs (IP_SFR), as requested in [CCDB].

The table also provides the link between the relevant platform-SFRs and the composite product SFRs.

No contradictions have been found between the RP_SFR set and the SFRs related to the composite-product.

Platform SFR	RP_SFR-SERV	RP_SFR-MECH	IP_SFR	Composite product SFRs
FDP_IFC.2/GP-ELF		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_IFF.1/GP-ELF		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_ITC.2/GP-ELF		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_IFC.2/GP-KL		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_IFF.1/GP-KL		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_ITC.2/GP-KL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MTD.1/GP-LC	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MTD.1/GP-PR	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCS_CKM.1/GP-SCP	X			FCS_CKM.1/Keys_Crypto
FCS_COP.1/GP-SCP	X			FCS_COP.1
FTP_TRP.1/GP-TF	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MSA.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MSA.3/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE

[27] Using the composition tasks terminology, the platform is the Connected eSE 5.3.4 v1.1.

[28] RP_SFR-SERV designates relevant platform SFRs used by the composite TOE to implement security services with associated TSFI. RP_SFR-MECH designates relevant platform SFRs used by the composite TOE as mechanisms to provide global protection against attacks.

Platform SFR	RP_SFR-SERV	RP_SFR-MECH	IP_SFR	Composite product SFRs
				SFRs
FMT_SMR.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_SMF.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FPT_RCV.3/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FPT_FLS.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FPT_TDC.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FTP_ITC.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCO_NRO.2/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FIA_UID.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_UIT.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_ROL.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_UCT.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FPR_UNO.1/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FIA_UAU.1/GP		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FIA_UAU.4/GP		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FIA_AFL.1/GP		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FMT_MTD.3/GP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCS_COP.1/GP-CLFDB	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_ACC.1/GP-GS		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_ACF.1/GP-GS		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FMT_MSA.1/GP-GS	X			No direct link to TOE SFRs but No contradiction with composite TOE

Platform SFR	RP_SFR-SERV	RP_SFR-MECH	IP_SFR	Composite product SFRs
				SFRs
FMT_MSA.3/GP-GS	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_SMR.1/GP-GS	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_SMF.1/GP-GS	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FIA_AFL.1/GP-CVM	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FPR_UNO.1/GP-CVM	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCO_NRR.1/GP-RECEIPT	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCO_NRO.2/GP-TOKEN	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCS_COP.1/GP-TOKEN	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCS_COP.1/GP-RECEIPT	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCS_COP.1/GP-DAP_SHA	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCS_COP.1/GP-DAP_VER	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCO_NRO.2/GP-DAP	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCS_CKM.1/GP-CCCM	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCS_COP.1/GP-CCCM	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_IFC.2/GP-CCCM		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_IFF.1/GP-CCCM		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FMT_MSA.1/GP-CCCM	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MSA.3/GP-CCCM	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FTP_ITC.1/GP-CCCM	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_ACC.1/GP-CTL	X			No direct link to TOE SFRs but No contradiction with composite TOE

Platform SFR	RP_SFR-SERV	RP_SFR-MECH	IP_SFR	Composite product SFRs
				SFRs
FDP_ACF.1/GP-CTL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_ROL.1/GP-CTL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MSA.1/GP-CTL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MSA.3/GP-CTL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_SMR.1/GP-CTL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_SMF.1/GP-CTL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FTP_ITC.1/GP-CTL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_ACC.1/GP-ELFU		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_ACF.1/GP-ELFU		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_ROL.1/GP-ELFU	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MSA.1/GP-ELFU	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MSA.3/GP-ELFU	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_SMF.1/GP-ELFU	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FPT_FLS.1/GP-ELFU	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_ACC.1/OS-UPDATE			X	Not used
FDP_ACF.1/OS-UPDATE			X	Not used
FMT_MSA.3/OS-UPDATE			X	Not used
FMT_SMR.1/OS-UPDATE			X	Not used
FMT_SMF.1/OS-UPDATE			X	Not used
FIA_ATD.1/OS-UPDATE			X	Not used
FTP_TRP.1/OS-UPDATE			X	Not used
FCS_COP.1/OS-UPDATE-DEC			X	Not used
FCS_COP.1/OS-UPDATE-VER			X	Not used
FPT_FLS.1/OS-UPDATE			X	Not used
FDP_ACC.2/FIREWALL		X		FDP_ACC.2
FDP_ACF.1/FIREWALL		X		FDP_ACF.1
FDP_IFC.1/JCVM		X		FDP_IFC.1
FDP_IFF.1/JCVM		X		FDP_IFF.1

Platform SFR	RP_SFR-SERV	RP_SFR-MECH	IP_SFR	Composite product SFRs
FDP_RIP.1/OBJECTS		X		FDP_RIP.1
FMT_MSA.1/JCRE		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FMT_MSA.1/JCVM		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FMT_MSA.2/FIREWALL_JCVM		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FMT_MSA.3/FIREWALL		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FMT_MSA.3/JCVM		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FMT_SMF.1	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_SMR.1	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCS_CKM.1/TDES			X	Not used
FCS_CKM.1/AES			X	Not used
FCS_CKM.1/RSA			X	Not used
FCS_CKM.1/ECDSA	X			FCS_CKM.1.1/ECC
FCS_CKM.1/HMAC	X			FCS_CKM.1.1/Session_keys FCS_CKM.1.1/Long_Term_key FCS_CKM.1.1/Secret_Shared_key
FCS_CKM.4	X			FCS_CKM.4
FCS_COP.1/TDES_CIPHER			X	Not used
FCS_COP.1/TDES_MAC			X	Not used
FCS_COP.1/AES_CIPHER	X			FCS_COP.1/Encryption/decryption
FCS_COP.1/AES_MAC			X	Not used
FCS_COP.1/RSA_SIGN			X	Not used
FCS_COP.1/RSA_CIPHER			X	Not used
FCS_COP.1/ECDSA_SIGN	X			FCS_COP.1/ECDSA
FCS_COP.1/ECDH			X	Not used
FCS_COP.1/DH			X	Not used
FCS_COP.1/Hash	X			FCS_COP.1/Hash
FCS_COP.1/HMAC	X			FCS_COP.1/HMAC
FCS_COP.1/CRC	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FCS_RNG.1	X			FCS_RNG.1
FDP_RIP.1/ABORT	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_RIP.1/APDU	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_RIP.1/GlobalArray	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_RIP.1/bArray	X			No direct link to TOE SFRs but No contradiction with composite TOE

Platform SFR	RP_SFR-SERV	RP_SFR-MECH	IP_SFR	Composite product SFRs
				SFRs
FDP_RIP.1/KEYS	X			FDP_RIP.1
FDP_RIP.1/TRANSIENT	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_ROL.1/FIREWALL		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FAU_ARP.1		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_SDI.2/DATA		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FPR_UNO.1		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FPT_FLS.1/JCS		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FPT_TDC.1	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FIA_ATD.1/AID	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FIA_UID.2/AID	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FIA_USB.1/AID	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MTD.1/JCRE	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MTD.3/JCRE	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_ACC.2/ADEL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_ACF.1/ADEL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_RIP.1/ADEL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MSA.1/ADEL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_MSA.3/ADEL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_SMF.1/ADEL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FMT_SMR.1/ADEL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FPT_FLS.1/ADEL	X			No direct link to TOE SFRs but No

Platform SFR	RP_SFR-SERV	RP_SFR-MECH	IP_SFR	Composite product SFRs
				contradiction with composite TOE SFRs
FDP_RIP.1/ODEL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FPT_FLS.1/ODEL	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FPT_RCV.3/OS	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FPT_RCV.4/OS	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs
FDP_SDI.2/ARRAY		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_SDI.2/RESULT		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_SDI.2/MONOTONIC_COUNTER		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FDP_SDI.2/CRT_MNGT		X		No direct link to composite TOE SFRs but provides global protection against attacks.
FCS_COP.1/CRT_MNGT	X			No direct link to composite TOE SFRs but provides global protection against attacks.
FCS_CKM.5/KDF	X			FCS_CKM.1.1/Session_keys FCS_CKM.1.1/Long_Term_key FCS_CKM.1.1/Secret_Shared_key
FPT_STM.1/SYS_TIME	X			No direct link to TOE SFRs but No contradiction with composite TOE SFRs

Table 29 Statement of compatibility – SFR Part

8. TOE SUMMARY SPECIFICATION

8.1. mDKA V2.5-15E0 TOE SECURITY FUNCTIONS

The following TOE Security Functions are provided by the Thales Embedded Software:

TOE SECURITY FUNCTIONS IDENTIFICATION	NAME
SF.CRYPTO	Cryptography management
SF.ACCESS_CONTROL	Access control
SF.INTEGRITY	Integrity monitoring
SF.AUTHENTICATION	Authentication management
SF.MANAGEMENT	Security management
SF.PROTECTION	Protection management
SF.SECURE_MESSAGING	Secure messaging management

8.1.1. SF.CRYPTO

SF.CRYPTO provides the crypto management on the TOE. It encompasses:

- EC Point Generation: the TSF generates EC point based on ECC with P-256 (SECP256r1) protocol. The generation is compliant with [FIPS PUB 186-4] standard as defined in **FCS_CKM.1/ECC**.
- Session keys: the TSF generates sessions keys for secure channel based on HKDF-SHA-256 The generation is compliant with IETF [RFC 5869] as defined in **FCS_CKM.1/Session_keys**, **FCS_CKM.1/Long_Term_key** and **FCS_CKM.1/Secret_Shared_key**.
- Key Establishment: the TSF distribute cryptographic keys based on Elliptic curve-based Diffie-Hellman Ephemeral key agreement. The distribution is compliant with NIST Special Publication [NIST-SP-800-56A] Revision 3 with approved groups from Appendix D as defined in **FCS_CKM.2/ECDHE**.
- Random numbers generation: the TSF generates random numbers that meet the [AIS31] quality as defined in **FCS_RNG.1**.
- Cryptographic Hashing: the TSF perform Cryptographic Hashing based on SHA-256. The operation is compliant with ISO/IEC 10118-3:2018, FIPS 180-4 as defined in **FCS_COP.1/Hash**.
- Keyed Hash Message Authentication: the TSF perform Cryptographic Hashing based on HMAC-SHA-256. The operation is compliant with ISO/IEC 9797-2:2011 as defined in **FCS_COP.1/HMAC**.
- Data encryption or decryption: the TSF perform Cryptographic Hashing based on AES with CBC mode of operation. The operation is compliant with [FIPS PUB 197], [NIST SP 800-38A] as defined in **FCS_COP.1/Encryption/decryption**.
- Message Authentication Code: the TSF perform Cryptographic Hashing based on AES CMAC. The operation is compliant with [NIST SP 800-38B] as defined in **FCS_COP.1/CMAC**.
- Digital signing: the TSF perform Cryptographic Hashing based on ECDSA with NIST P-256 curve. The operation is compliant with [ANSI X9.62] as defined in **FCS_COP.1/ECDSA**.
- Key destruction: the TSF destroy cryptographic keys based on [pseudo-random pattern] method and make sure no residual information is available as defined in **FCS_CKM.4.1** and in **FDP_RIP.1**.

8.1.2. SF.ACCESS_CONTROL

SF.ACCESS_CONTROL provides access control to the TOE. It encompasses:

- Access control policy and functions for Security domain as defined in **FDP_ACC.2** and **FDP_ACF.1**.
- Information flow control policy and functions for secure channel protocol as defined in **FDP_IFC.2** and **FDP_IFF.1**.

8.1.3.SF.INTEGRITY

SF.INTEGRITY provides the integrity monitoring of the TOE. It encompasses:

- The integrity of sensitive stored user data as defined in **FDP_SDI.2**.
- The integrity of exchanged user data as defined in **FDP_UIT.1**.

8.1.4.SF.AUTHENTICATION

SF.AUTHENTICATION provides the authentication management of the TOE. It encompasses:

- Detection of forgeable authentication as defined in **FIA_UAU.3**.

8.1.5.SF.MANAGEMENT

SF.MANAGEMENT provides security management of the TOE. It encompasses:

- Restriction of keys deletion as defined in **FMT_MTD.1**.
- Check of applet's AID accepted values as defined in **FMT_MTD.3.**
- Ability to create a deletion attestation for the requested key (for deletion), and that it is securely deleted before the attestation is transferred to the requesting party as defined in **FMT_SMF.1**.
- Management of roles as defined in **FMT_SMR.1**.

8.1.6.SF.PROTECTION

SF.PROTECTION provides security protection of the TOE. It encompasses:

- Insurance that any entity is unable to determine whether data and key exchanged over NFC (between the TOE and the Vehicle) were caused by the same user as defined in **FPR_UNL.1**.
- Protection of all TSF data transmitted from the TSF to another trusted IT product from unauthorized disclosure during transmission as defined in **FPT_ITC.1**.
- Modification detection and integrity verification all TSF data transmitted between the TSF and another trusted IT product or separate part of the TOE as defined in **FPT_ITI.1/Vehicle_Integrity** and **FPT_ITT.3**.
- Protection from disclosure as defined in **FPT_ITT.1.**
- Replay detection as defined in **FPT_RPL.1**.
- Automated recovery as defined in **FPT_RCV.2**.
- Resistance to physical manipulation and physical probing to the TSF as defined in **FPT_PHP.3**.

8.1.7.SF.SECURE_MESSAGING

SF.SECURE_MESSAGING provides secured communication channel management for the TOE. It encompasses:

- Communication channel between itself and another trusted IT product management as defined in **FTP_ITC.1**.

8.2. TSS RATIONALE

Security Functional Requirement	Coverage by TSS security function(s)
FCS_CKM.1/Keys_Crypto	SF.CRYPTO
FCS_RNG.1	SF.CRYPTO
FCS_CKM.2/ECDHE	SF.CRYPTO
FCS_CKM.4	SF.CRYPTO
FCS_COP.1	SF.CRYPTO
FDP_ACC.2	SF.ACCESS_CONTROL
FDP_ACF.1	SF.ACCESS_CONTROL
FDP_IFC.2	SF.ACCESS_CONTROL
FDP_IFF.1	SF.ACCESS_CONTROL
FDP_RIP.1	SF.CRYPTO
FDP_SDI.2	SF.INTEGRITY
FDP_UIT.1	SF.INTEGRITY
FIA_UAU.3	SF.AUTHENTICATION
FMT_MTD.1	SF.MANAGEMENT
FMT_MTD.3	SF.MANAGEMENT
FMT_SMF.1	SF.MANAGEMENT
FMT_SMR.1	SF.MANAGEMENT
FPR_UNL.1	SF.PROTECTION
FPT_ITC.1	SF.PROTECTION
FPT_ITI.1/Vehicle_Integrity	SF.PROTECTION
FPT_ITT.1	SF.PROTECTION
FPT_ITT.3	SF.PROTECTION
FPT_RPL.1	SF.PROTECTION
FPT_RCV.2	SF.PROTECTION
FTP_ITC.1	SF.SECURE_MESSAGING
FPT_PHP.3	SF.PROTECTION

Table 30 TOE SECURITY FUNCTIONS RATIONALE

END OF DOCUMENT