Car Connectivity Consortium

CCC Digital Key®

Protection Profile of the Digital Key Applet

Version 1.0
(CCC-CP-023)

CARCONNECTIVITY consortium®

VERSION HISTORY

Version	Date	Comments
1.0	2023-10-16	Approved by CCC Board.

LEGAL NOTICE

The copyright in this Specification is owned by the Car Connectivity Consortium LLC ("CCC LLC"). Use of this Specification and any related intellectual property (collectively, the "Specification"), is governed by these license terms and the CCC LLC Limited Liability Company Agreement (the "Agreement").

Use of the Specification by anyone who is not a member of CCC LLC (each such person or party, a "Member") is prohibited. The legal rights and obligations of each Member are governed by the Agreement and their applicable Membership Agreement, including without limitation those contained in Article 10 of the LLC Agreement.

CCC LLC hereby grants each Member a right to use and to make verbatim copies of the Specification for the purposes of implementing the technologies specified in the Specification to their products ("Implementing Products") under the terms of the Agreement (the "Purpose"). Members are not permitted to make avail-able or distribute this Specification or any copies thereof to non-Members other than to their Affiliates (as defined in the Agreement) and subcontractors but only to the extent that such Affiliates and subcontractors have a need to know for carrying out the Purpose and provided that such Affiliates and subcontractors accept confidentiality obligations similar to those contained in the Agreement. Each Member shall be responsible for the observance and proper performance by such of its Affiliates and subcontractors of the terms and conditions of this Legal Notice and the Agreement. No other license, express or implied, by estoppel or otherwise, to any intellectual property rights are granted herein.

Any use of the Specification not in compliance with the terms of this Legal Notice, the Agreement and Membership Agreement is prohibited and any such prohibited use may result in termination of the applicable Membership Agreement and other liability permitted by the applicable Agreement or by applicable law to CCC LLC or any of its members for patent, copyright and/or trademark infringement.

THE SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION ANY WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS, AND COMPLIANCE WITH APPLICABLE LAWS.

Each Member hereby acknowledges that its Implementing Products may be subject to various regulatory controls under the laws and regulations of various jurisdictions worldwide. Such laws and regulatory controls may govern, among other things, the combination, operation, use, implementation and distribution of Implementing Products. Examples of such laws and regulatory controls include, but are not limited to, road safety regulations, telecommunications regulations, technology transfer controls and health and safety regulations. Each Member is solely responsible for the compliance by their Implementing Products with any such laws and regulations and for obtaining any and all required authorizations, permits, or licenses for their Implementing Products related to such regulations within the applicable jurisdictions.

Each Member acknowledges that nothing in the Specification provides any information or assistance in connection with securing such compliance, authorizations or licenses.

NOTHING IN THE SPECIFICATION CREATES ANY WARRANTIES, EITHER EXPRESS OR IMPLIED, REGARDING SUCH LAWS OR REGULATIONS. ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY INTELLECTUAL PROPERTYRIGHTS OR FOR NONCOMPLIANCE WITH LAWS, RELATING TO USE OF THE SPECIFICATION IS EXPRESSLY DISCLAIMED. BY USE OF THE SPECIFICATION, EACH MEMBER EXPRESSLY WAIVES ANY CLAIM AGAINST CCC LLC AND ITS MEMBERS RELATED TO USE OF THE SPECIFICATION.

CCC LLC reserve the right to adopt any changes or alterations to the Specification as it deems necessary or appropriate.

EXECUTIVE SUMMARY

The Car Connectivity Consortium (CCC) represents a large portion of the global automotive and smartphone industries, with more than one hundred member companies.

The CCC is a cross-industry standards organization with a mission to create sustainable and flexible ecosystems that standardize interface technologies to provide consistently great user experiences across all vehicles and mobile devices.

The Car Connectivity Consortium Digital Key® is a standardized ecosystem that enables mobile devices to store, authenticate, and share Digital Keys for vehicles in a secure, privacy-preserving way that works everywhere, even when the smartphone's battery is low.

Digital Key allows consumers to easily and confidently use their mobile devices to access vehicles. Along with robust capability and convenience, it offers enhanced security and privacy protections. Digital Key aims to complement traditional methods, while being robust enough to fully replace them.

The CCC Digital Key Release 3 defines the standardized interface between the vehicle and the mobile device as a NFC-based wireless interface designed for direct communication between the vehicle and mobile device.

The Digital Key architecture uses standards-based public key infrastructure to establish end-to-end trust. Mobile devices create and store Digital Keys in Secure Elements – embedded technology that provides a tamper-resistant secure implementation – to provide the highest-level of protection from the plethora of known hardware- and software-based attacks, including tampering, storage intrusion, cloning, and unauthorized access as well as side channel, fault injection, and many other forms of attack.

To ensure this is the case, proper standards are in place and implementations will be tested and provide assurance on their security level. Common Criteria is chosen to express the security requirements for security evaluations of these implementations. To enable certification of such implementations CCC requires a Protection Profile or similar document to be created. This is what is presented in this document, a Protection Profile (PP) that may be brought through Common Criteria certification body approval at later stage. This PP has been prepared following the rules and formats of Common Criteria version 3.1 revision 5.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

TERMS AND ABBREVIATIONS

AID	Application Identifier
API	Application Programming Interface
BT	Bluetooth
CA	Certificate Authority
CC	Common Criteria
DK	Digital Key
ECU	Electronic Control Unit
GP	GlobalPlatform
KTS	Key Tracking Server
MITM	Man-in-the-middle attack
NFC	Near Field Communication
NVM	Non-Volatile Memory
OEM	Original Equipment Manufacturer
PP	Protection Profile
RE	Runtime Environment
SCP	Secure Channel Protocol
SE	Secure Element
TOE	Target of Evaluation
TSFI	TOE Security Functionality Interface
UI	User Interface
VM	Virtual Machine

(CC terminology, defined in [CC1] CC part 1 is not listed here.)

TERMINOLOGY AND DEFINITIONS

In this document keywords are capitalized when used to unambiguously specify an interpretation. When these words are not capitalized, they are meant in their natural-language sense.

The key words “SHALL”, “SHALL NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

- SHALL - This word, or the terms “REQUIRED” or “SHALL”, mean that the definition is an absolute requirement of the specification
- SHALL NOT - This phrase, or the phrase “SHALL NOT”, mean that the definition is an absolute prohibition of the specification
- SHOULD - This word, or the adjective “RECOMMENDED”, mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications SHALL be understood and carefully weighed before choosing a different course
- SHOULD NOT - This phrase, or the phrase “NOT RECOMMENDED” mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
- MAY - This word, or the adjective “OPTIONAL”, mean that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation which does not include a particular option SHALL be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality. In the same vein an implementation which does include a particular option SHALL be prepared to interoperate with another implementation which does not include the option (except, of course, for the feature the option provides.)

1 INTRODUCTION

1.1 PP IDENTIFICATION

Title:	**Protection Profile of the Digital Key Applet**
Editor:	Red Alert Labs, SAS
Sponsor:	Car Connectivity Consortium (CCC), LLC
Supported/Certified by:	Federal Office for Information Security (BSI) Germany
CC Version:	version 3.1 revision 5.
Assurance Level:	EAL 4 augmented with ALC_DVS.2 and AVA_VAN.5.
Version:	1.0 as of October 16, 2023
Registration:	BSI-CC-PP-0119
Keywords:	Digital Key, Secure Element,

1.2 PP Presentation

The CCC Digital Key, Release 3 [CCC-DK-TS] the third in a series of releases, allows individual owners to use their mobile devices as keys to their vehicles. The specification enables:

- Security and privacy equivalent to physical keys.
- Interoperability and user experience consistency across mobile devices and vehicles.
- Vehicle access, start, mobilization, and other use cases.
- Owner pairing and key sharing with friends, with standard or custom entitlement profiles.
- Support for mobile devices with low batteries.

The Digital Key architecture uses standards-based public key infrastructure to establish end-to-end trust. Mobile devices create and store Digital Keys in Secure Elements – embedded technology that provides a tamper-resistant secure implementation – to provide the highest-level of protection from the plethora of known hardware- and software-based attacks, including tampering, storage intrusion, cloning, and unauthorized access as well as side channel, fault injection, and many other forms of attack.

Mobile devices may act as either owner or friend devices, but the vehicle-to-device interface is the same in either role. Interoperability between mobile devices and vehicles is supported by standardizing the vehicle-to-device interface – the communication channel (NFC), protocols, and Digital Key structures.

This Protection Profile written in Common Criteria language explains the security requirements of the DK Applet executing on a Secure Element (SE) implementing Java Card specifications and GlobalPlatform Card Specifications.

1.3 TOE Overview

This chapter defines the Target of Evaluation (TOE) type and describes the main security features of the TOE, the components of the TOE environment, the TOE life cycle and TOE intended usage.

The DK Applet evaluation is performed as a composite evaluation in the sense where the DK Applet is implemented on top of:

- a SE Java Card and GlobalPlatform platform certified conformant with [PP0099] or any other protection profile including all claims defined in [PP0099]
- a certified IC conformant with [PP0084] (see Figure 1 TOE Architecture).

1.3.1 *TOE Definition*

This section presents the architecture and common usages of the Target of Evaluation (TOE). The Target of Evaluation (TOE) is the DK Applet embedded in a SE Java Card and GlobalPlatform platform intended to be embedded in a mobile device in order to enable an end user to easily and confidently use their mobile devices as a key to their vehicle granting access to the owner, sharing with a friend, starting the engine, etc.

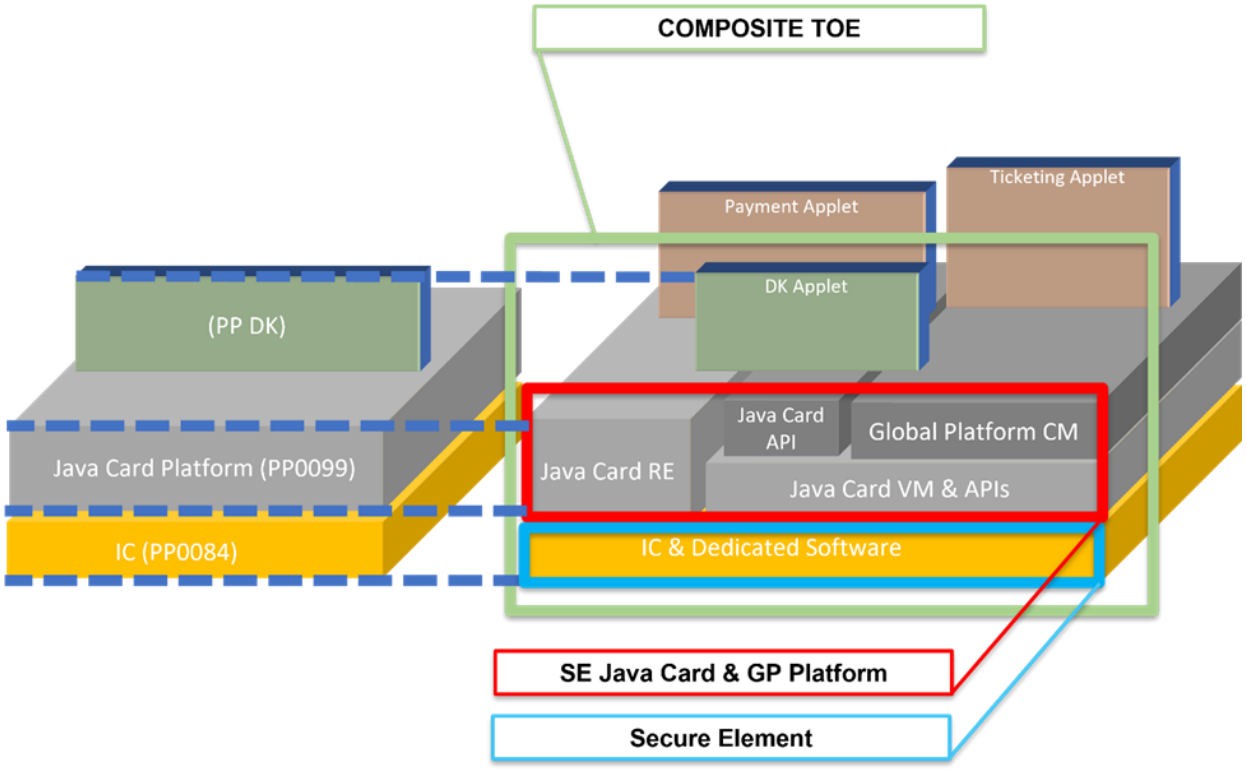
The TOE SHALL comprise at least:

- the Secure Element – a chip (IC layer)
- the Java Card Platform and GlobalPlatform (OS layer)
- the DK applet (Application layer)
- the associated guidance documentation

The TOE SHALL implement CCC Digital Key specifications [CCC-DK-TS].

The generic architecture of the TOE is described hereafter (see Figure 1 TOE Architecture) and detailed in the following paragraphs.

Figure 1 TOE Architecture



The Digital Key architecture uses standards-based public key infrastructure to establish end-to-end trust. Mobile devices create and store Digital Keys in Secure Elements – embedded technology that provides a tamper-resistant secure implementation – to provide the highest-level of protection from the plethora of known hardware- and software-based attacks, including tampering, storage intrusion, cloning, and unauthorized access as well as side channel, fault injection, and many other forms of attack.

The Digital Key applet, which resides within the Secure Element, performs all security-critical processing – authentication, encryption protocols, and key generation used for owner pairing, sharing, and vehicle access and engine start transactions – while also providing secure, tamper-proof storage for Digital Keys and their metadata.

The NFC interface is routed directly to the Digital Key applet, providing a communications path that is protected from, and that operates independently of, the rest of the mobile device.

The Digital Key applet provides the following services:

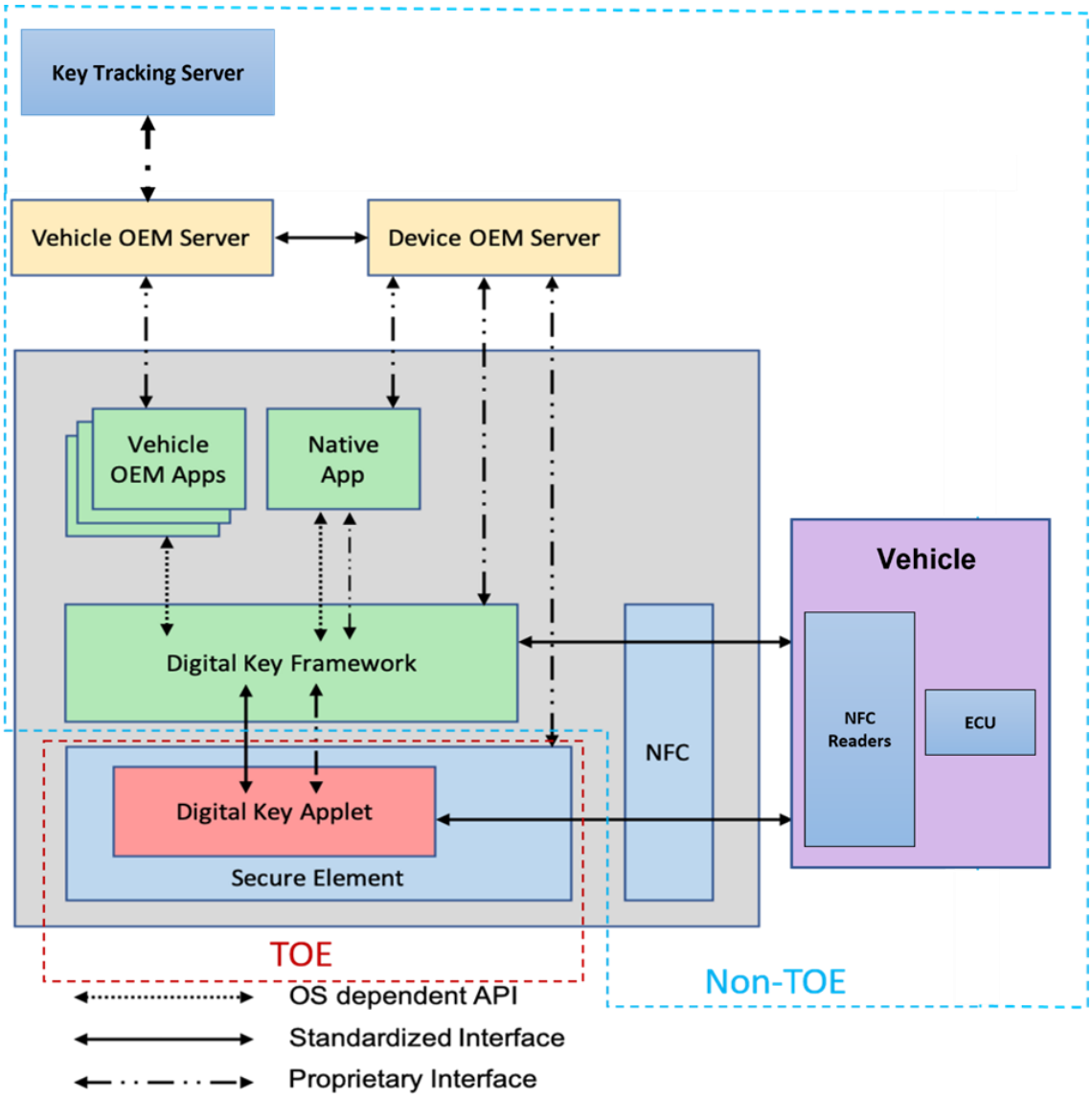
- Hosts Digital Keys (one applet instance hosts Digital Keys of all Vehicle OEMs)
- Implements relevant (fast and standard) transactions
- Implements Instance CA to support offline use cases and privacy protection
- Stores immobilizer tokens when required by the vehicle, offline attestations, access profiles, and other data associated with a Digital Key
- Verifies authenticity of the vehicle
- Verifies certificate chain of friend public key

If the Digital Key applet is the SE-centric applet model, the applet also provides the following service:

- Verifies the Vehicle Public Key Certificate

1.4 Non-TOE HW/SW/FW Available to the TOE

Figure 2 Non-TOE



The above diagram (Figure 2 Non-TOE) 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.

The TOE relies on each one of the non-TOE components described below in order to fulfil a functionality.

Note that Vehicle OEMs and Device OEMs provide their own PK infrastructure and the root CA each in their context. The Vehicle OEM signs Device OEM root CA certificate.

Note: The proprietary interface depicted in (Figure 2 Non-TOE) between the Device OEM Server and Secure Element is supplied by the Device OEMs. This proprietary interface is outside of the responsibility and authority of CCC.

Note: The proprietary interface depicted in (Figure 2 Non-TOE) between the Device OEM Server and Secure Element needs to be assessed during an evaluation of a TOE claiming conformance to this PP.

This section explains in detail about the Non-TOE region consisting of the Digital Key Framework, Vehicle OEM Apps, Native Apps, Device OS, etc.

1.4.1 *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.

1.4.2 *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

1.4.3 *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.

1.4.4 *Device OS*

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

1.4.5 *DK Applet EE*

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

1.4.6 *Vehicle – ECU*

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

1.4.7 *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

1.4.8 *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.

1.4.9 *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

1.4.10 *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

1.4.11 *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)

Note that there are two different usages (for two different roles) on similar devices including different features. For instance, the friend device hosts Digital Key shared by an owner but it cannot share it with any other device. The Friend device may have restricted access rights to the vehicle compared to the owner.

1.4.11.1 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

1.4.11.2 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

1.4.12 *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

1.5 TOE Lifecycle

The TOE life cycle follows the description of the [PP0099] and is part of the product life cycle, i.e. the DK Applet, which goes from product development to its usage by the final user.

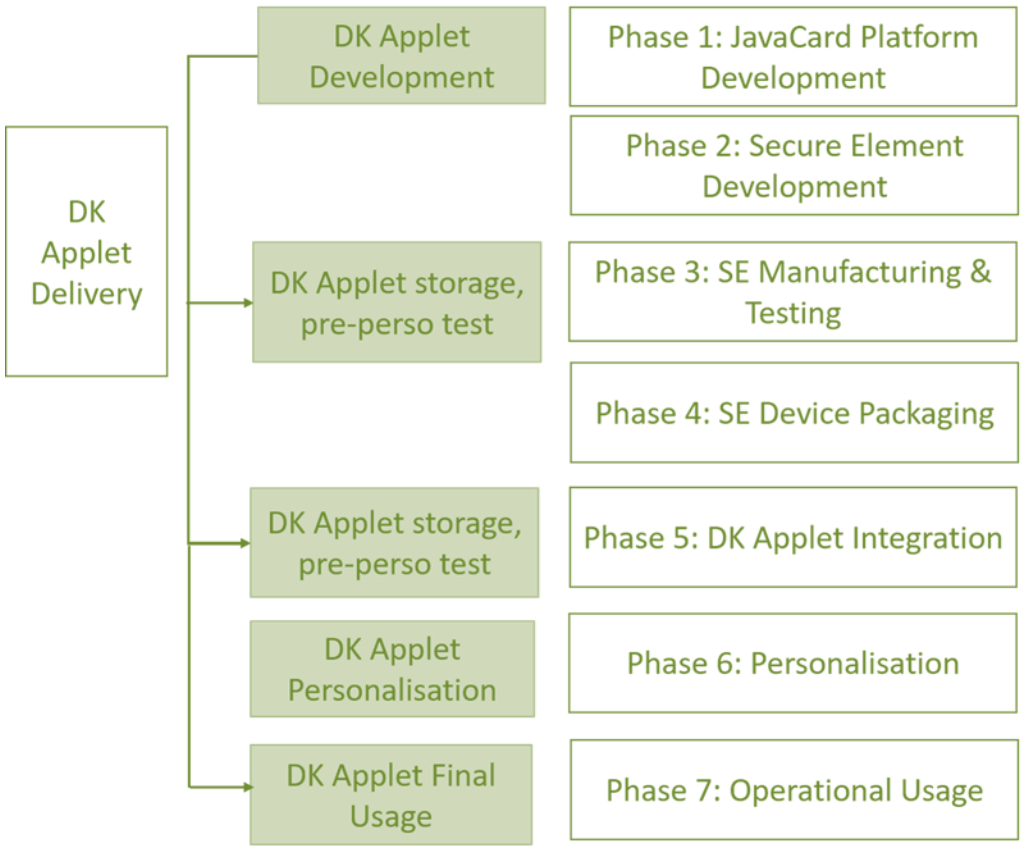
The product life cycle phases are those detailed in Figure 3 TOE Lifecycle. We refer to [PP0084] for a thorough description of Phases 1 to 7:

- Phases 1 and 2 compose the product development: Embedded Software (SE Dedicated Software, OS, Java Card System, DK applet, other platform components such as Card Manager, Applets) and IC development.
- Phase 3 and 4 correspond to SE manufacturing and packaging, respectively. Some SE pre-personalisation steps may occur in Phase 3.
- Phase 5 concerns the embedding of software components within the SE.
- Phase 6 is dedicated to the product personalisation (DK Applet integration & personalisation) prior final use.
- Phase 7 is the product operational phase.

The DK Applet life cycle is composed of four stages:

- Development,
- Storage, pre-personalisation, and test,
- Personalization and test,
- Final usage.

Figure 3 TOE Lifecycle



The above diagram (Figure 3 TOE Lifecycle) explains the TOE Lifecycle. Please note that to have a complete overview of how the DK Applet life-cycle is aligned with the operational environment (mobile device) lifecycle, the ST writer could include the phases of a mobile device development.

Phase 1 consists of the DK Applet Development. This phase could happen in parallel to the SE Java Card and GlobalPlatform platform development and covers the DK applet conception phase, design, implementation, testing and its documentation. The development SHALL be carried out in such a way that it is conformant with the SE Java Card and GlobalPlatform platform guidelines. The development SHALL take place in a controlled environment. This is important to guarantee the integrity of the developing elements and to ensure non-disclosure of sensitive/confidential data. The evaluation of a product against this PP SHALL include the DK Applet development environment.

Exceptionally, the delivery of the DK Applet could take place in complete form or in parts during either of the two phases: Phase 3(SE Manufacturing & Testing) or Phase 5(DK Applet Integration). Otherwise, the typical delivery of the DK Applet happens during Phase 7 in a post-issuance loading form. Delivery and acceptance procedures SHALL guarantee the authenticity, the confidentiality and integrity of the exchanged pieces. The evaluation of a product against this PP SHALL include the delivery process.

Phase 3 consists of the SE Manufacturing & testing part. If the SE has not been certified (e.g. against [PP0084]), the evaluation of a product against this PP SHALL include SE Manufacturing Environment.

During phase 5, the Integrators performs the storage, pre-personalisation of the DK Applet and may also conduct tests. The product Integration environment SHALL also protect the confidentiality and integrity of the DK Applet and of any related material being used including testing material. The evaluation of a product against this PP SHALL take into account the Integration environment.

The personalisation of the DK Applet takes place during phase 6. Thus, the personalisation environment SHALL ensure the confidentiality and integrity of any associated data or material being used.

During phase 7, the operational usage of the DK Applet takes place. At this phase, the DK Applet is loaded into the SE. The DK Applet final usage environment is that of the SE where the DK Applet is embedded in. It covers a wide spectrum of situations that cannot be covered by evaluations. The DK Applet and the product SHALL provide the full set of security functionalities to avoid abuse of the product by untrusted entities.

1.6 TOE Security Features

This section explains in summary the security features of the TOE. The security features of the Java Card Platform compliant with the [PP0099] are not listed in this PP. The security features explained below have been derived from the Technical Specification documentation [CCC-DK-TS].

1.6.1 *Secure Owner Pairing*

The owner pairing flow is operated by the Digital Key framework and DK Applet running on the device. The Digital Key framework uses the APDU commands to manage the configuration of the Digital Key, protected by the SE. The SE provides the root of trust, which is the starting point in the trust chain.

The vehicle is able to select the Digital Key applet over NFC using its AID and to select the Digital Key framework using the corresponding AID. The NFC controller may be reconfigured for changing the routing of the communication from the SE to the Digital Key framework and vice versa, based on the selected AID. A new owner device pairing flow, or owner device change, does not imply an implicit unpairing, i.e., a new device owner pairing flow only changes the owner’s key. Existing shared/friend keys that are already paired, and vehicle public keys, are not necessarily impacted. The Digital Key applet instance SHALL be available on the SE before the time of owner pairing execution. Note that the Device OEM CA root key is never stored in the SE - Either variant 1 (SE root of trust based on CASD) or variant 2 (SE root of trust based on DK applet associated security domain) must be implemented according to the specifications.

1.6.2 *Secure Standard Transaction*

A secure channel between vehicle and device is initiated by generating ephemeral key pairs on the vehicle and device sides. Using a key agreement method, a shared secret can be derived on both sides and used for generation of a shared symmetric key, using Diffie-Hellman and a key derivation function.

The ephemeral public key generated on the vehicle side is signed with the vehicle's private key, vehicle_SK. This results in an authentication of the vehicle by the device. From the device's perspective, this guarantees that no privacy-sensitive data 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.

Finally, the device uses the established secure channel to encrypt its public key identifier along with the signature computed on a vehicle's data-derived challenge and some additional application-specific data. This verification of the device's signature by the vehicle allows the vehicle to authenticate the device.

1.6.3 *Secure Fast Transaction*

The device generates a cryptogram based on a secret previously shared during a standard transaction, and this allows the vehicle to authenticate the device. Optionally, a secure channel between vehicle and device is established by deriving session keys from a secret previously shared during a standard transaction and from the ephemeral keys. The ability of the vehicle to establish the secure channel authenticates the vehicle to the device.

The fast transaction protocol is intended to provide the following properties:

- Device authentication or Mutual authentication
- Integrity and confidentiality
- Tracking resilience

1.6.4 *Secure Check Presence Transaction*

The check presence transaction protocol is intended to provide the following properties:

- Vehicle authentication
- Device identification
- Integrity and confidentiality
- Tracking resilience

The mechanism is similar to the standard transaction mechanism described in Section 1.6.2, except that the device signature is not sent to the vehicle, and user authentication is disabled. The goal is to allow verification of device presence near the vehicle without requiring user authentication, while preventing tracking.

1.6.5 *Secure DK Sharing*

The DK sharing flow is operated by the Digital Key framework and DK Applet running on the owner and the friend device. During the sharing flow the owner device creates a sharing invitation that is sent to the friend device. Based on this invitation, the friend device creates a DK in the DK Applet and generates a key signing request, which is signed by the friend private key. The owner device then creates an attestation package over the friend public key from the key signing request and optionally exports and includes an immobilizer token from the owner DK confidential mailbox. At the end of this flow, the friend's private mailbox stores the attestation package, in which the friend's public key is signed by the owner private key along with a set of entitlements. The optional immobilizer token is stored in the friend`s confidential mailbox.

The attestation package is verified by the vehicle OEM KTS server during key tracking and also at the vehicle`s first transaction with the friend device.

1.6.6 *Key Termination & Suspension*

Unlike physical keys and key fobs, Digital Keys may be easily terminated or suspended by friend devices, owner devices, vehicles, and/or OEM Servers. 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.

1.6.7 *Secure Applet Management*

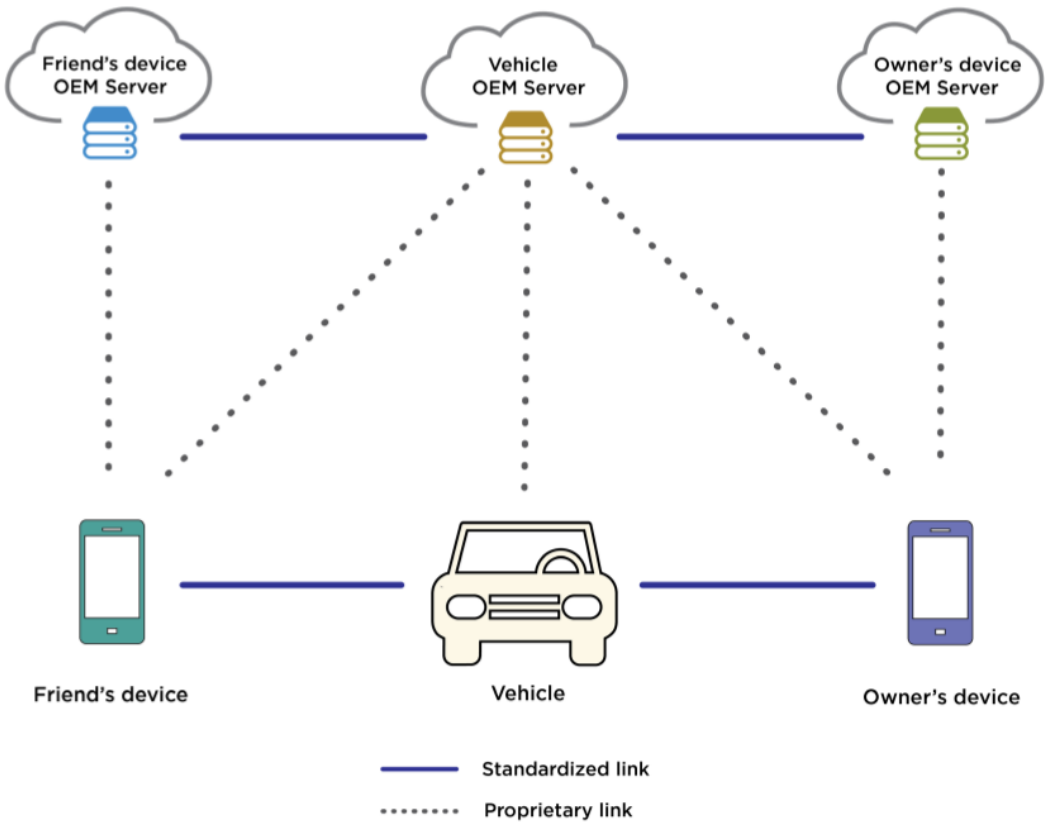
The TOE offers additional security services for applets management, relying on the GlobalPlatform framework:

- The SE issuer is by definition the main authorized entity to manage applications (loading, instantiation, deletion) through a secure communication channel with the SE.
- DK Applet Provider personalize its application and the associated Security Domain (SD) in a confidential manner. The DK Applet provider is usually the SE Issuer. The Security Domain keysets are used to establish a Secure Channel between the TOE and external entities (e.g. Device OEM server). In case the SE Issuer is not the DK applet provider, these Security Domains keysets are not known by the SE Issuer.
- The services provided by the Controlling Authority Security Domain (CASD) allows the implementation of the SE Root.

1.7 TOE Usage

Digital Key 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. It is an important addition to the automotive industry, enabling a significantly improved vehicle access experience that builds consumer confidence through its ease of use, convenience, security and privacy protections, and extensive capability. The CCC, representing the majority of the global automotive and mobile device industries, enables the Digital Key’s broad cross-industry support and facilitates the coordination of mobile Device OEMs and Vehicle OEMs to provide a consistent user experience by increasing interoperability and reducing market fragmentation.

Figure 4 DK System Architecture



The mobile device‘s native app allows consumers to use and manage Digital Keys without any extra apps, and its Digital Key framework enables developers to build custom apps that provide enhanced services and vehicle-specific features. Mobile devices and vehicles interact with their respective OEM servers to share and manage Digital Keys across mobile device and vehicle platforms. The system ensures that you’re able to access your vehicle even when neither your mobile device nor vehicle have Internet connectivity, while still allowing OEMs to add features that require Internet connectivity for certain operations.

The different use cases of DK Applet include:

- **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.

1.8 About Composition

Composition is applied using this PP whenever a DK applet vendor is intending to build upon a previously certified underlying platform (IC + OS). In such case, the underlying certified platform brings with it the results of its certification, together with specific evidence and information allowing an efficient composite evaluation.

More specifically, the TOE addressed in this Protection Profile is designed to allow composite certification. Indeed, the DK applet SHALL be certified in composition on top of a SE Java Card and GlobalPlatform platform (IC + OS + GP). The objective is to reuse as much as possible these elements that come with the certified Java Card Platform in the certification of the overall Secure Element including DK applet.

There are nevertheless a few common rules about composition that SHALL be applied in this composite certification:

- The DK applet SHALL be evaluated at least at the assurance level EAL 4+ as claimed by this PP and the level of assurance of the underlying platform SHALL be at least EAL 4+. This means, higher assurance level could be claimed by the Security Target when applicable.

- The composite TOE integrating the certified underlying platform SHALL satisfy the assumptions on the integrating environment which are enforced by the underlying platform user guidance (AGD_OPE) as defined in [PP0099].

2 CONFORMANCE CLAIM

2.1 CC CONFORMANCE CLAIM

This Protection Profile claims to be conformant to the Common Criteria version 3.1 revision 5 [CC]

Furthermore, it claims conformance to CC Part 2 [CC2] extended with the security functional components FCS_RNG.1 Random numbers generation, CC Part 3 [CC3] and the Common Criteria Evaluation Methodology [CEM].

2.2 CONFORMANCE CLAIM TO A PACKAGE

The minimum assurance level for this Protection Profile is EAL4 augmented with AVA_VAN.5 “Advanced methodical vulnerability analysis” and ALC_DVS.2 “Sufficiency of security measures”.

2.3 PROTECTION PROFILE CONFORMANCE CLAIM

This Protection Profile does not claim conformance to any other Protection Profile.

2.4 CONFORMANCE CLAIMS TO THIS PROTECTION PROFILE

This Protection Profile requires strict conformance for the Security Target or Protection Profile claiming conformance to this PP.

3 SECURITY PROBLEM DEFINITION

3.1 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).

Table 1 User Data Assets, Description and Sensitivity

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 TSF Data Assets, Description and Sensitivity

Assets (TSF Data)	Description	Sensitivity (C, I, A, P)
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

Assets (TSF Data)	Description	Sensitivity (C, I, A, P)
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,	C, I, A

Assets (TSF Data)	Description	Sensitivity (C, I, A, P)
	sufficient entropy, and forward secrecy are to be considered for this asset.	

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

Table 3 Storage of cryptographic keys

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

3.2 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.

Table 4 Threats, Description and Covered Assets

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 package file	D.SE_MNGT_DATA D.DK_APPLET_CODE

Threats	Description	Covered Assets
	• 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 re-personalizes 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 the system of the component. For instance, a malicious DK Sharing request is produced to the attacker's device.	D.DK_APPLET_CODE D.APPLET_ROOT_KEY
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	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

Threats	Description	Covered Assets
	possible for an attacker to use the vulnerable component to perform its attacks.	
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 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.	D.SECRET_SHARED_KEY
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 the Vehicle.	D.SESSION_KEYS

Threats	Description	Covered Assets
T.UPDATE_KEY_OPTIONS	Attackers modify the keys options to give all access to lock/unlock and Engine Start.	D.KEY_OPTIONS
T.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_KEY 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_KEY D.SEC_ATTRIBUTES D.OWNER_DK_SECRET D.OWNER_DK_DATA D.OWNER_DATA D.DK_API_DATA D.MESSAGES_EXCHANGES
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_KEY
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_KEY D.SE_MNGT_DATA D.SESSION_KEYS
T.SIGN_COMPROMISE_VERIFY_COMPROMISE	Attacker manipulates creation and validation of electronic signatures	D.IMMOTOKEN D.DK_APPLET_CODE D.APPLET_ROOT_KEY D.SESSION_KEYS D.OWNER_DK_SECRET

Threats	Description	Covered Assets
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_KEY D.DK_APPLET_CODE D.LONG_TERM_KEY D.APPLET_ROOT_KEY
T.RADIO_RELAY_TRANSACTION	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_SHARING	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_KEY D.LONG_TERM_KEY D.APPLET_ROOT_KEY D.OWNER_DK_SECRET
T.INSTANCE_CA_DISCLOSURE	An attacker is extracting the Instance CA from the secure storage of a Device or signs its own Instance CA with a 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.	D.OWNER_DK_DATA D.KEY_OPTIONS

3.3 Organizational 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.

Table 5 Organizational Security Policies description

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.
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.

Organizational Security Policies	Description
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.

3.4 Assumptions

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

Table 6 Assumptions description

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 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.

Assumptions	Description
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 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

4 SECURITY OBJECTIVES

4.1 SECURITY OBJECTIVES FOR THE TOE

This section describes the security objectives for the TOE.

Table 7 Description of ToE Security Objectives

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 key.
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.

Security Objectives	Description
O.RANDOMNESS	Only random number generators (RNG) generating sufficient entropy[2] 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[PP0084]) 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 [PP0084])
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 [PP0084])
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 [PP0084])
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.

[2] Please refer to standards such as the NIST SP800-90B or the AIS20/31 for an accurate definition of "Sufficient Entropy"

Security Objectives	Description
O.NON-TRACEABILITY	The TOE SHALL be able to ensure the non-traceability of data and keys being shared through an NFC channel.

4.2 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.

Table 8 Description of Operational Environment Security Objectives

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 above-mentioned processes.

Security Objectives	Description
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_SHARING	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

Security Objectives	Description
	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).

4.3 SECURITY OBJECTIVES RATIONALE

4.3.1 *SPD and Security Objectives*

4.3.1.1 Threats

Table 9 Threats and Security Objectives - Coverage

Threats from this PP	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.

Threats from this PP	Security Objectives	Rationale
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 • 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

Threats from this PP	Security Objectives	Rationale
		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 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.

Threats from this PP	Security Objectives	Rationale
T.RADIO_SNIFF	O.FAST_TRANSACTION_AUTH O.STD_TRANSACTION_AUTH	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_INTEG, O.LONG_TERM_KEY_INTEG, O.SESSION_KEYS_INTEG, O.FAST_TRANSACTION_AUTH O.STD_TRANSACTION_AUTH 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 ensures that the users are non-traceable across different vehicles through the same app.

Table 10 Threats and OE.Security Objectives - Coverage

Threats from this PP	OE Security Objectives	Rationale
T.NON_REVOKED	OE.CERTIFICATE_REVOCATION_SET	This threat is covered by OE. CERTIFICATE_REVOCATION_SET which

Threats from this PP	OE Security Objectives	Rationale
		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_SHARING 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_DOWNGRADE	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_TRANSACTION	OE.RADIO_RELAY	This threat is covered by OE.RADIO_RELAY which ensures protection against relay a transaction with radio equipment attack.
T.INTERNET_CONNECTIVITY_DOS	OE.DOS_DETECT	This threat is covered by 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

Threats from this PP	OE Security Objectives	Rationale
T.UNAUTHORIZED_KEY_SHARING	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_DISCLOSURE	OE.INSTANCE_CA_CONFIDENTIALITY	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

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

Table 11 Security Objectives and Threats - Coverage

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

Security Objectives	Threats
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

4.3.1.2 Organizational Security Policies

Table 12 OSPs and Security Objectives - Coverage

Organizational Security Policies	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_RETRIEVE	OE.SECURE_KEY_RETRIEVE	This OSP is enforced by the security objective for the operational environment of the TOE OE.SECURE_KEY_RETRIEVE

Organizational Security Policies	Security Objectives	Rationale
OSP.KEY_SHARE	OE.KEY_SHARE	This OSP is enforced by the security objective for the operational environment of the TOE 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 13 Security Objectives and OSPs - 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

Security Objectives	OSP
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
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	

4.3.1.3 Assumptions

Table 14 Assumptions and Security Objectives for the Operational Environment - coverage

Assumptions	Security Objectives for the Operational Environment	Assumptions
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

Assumptions	Security Objectives for the Operational Environment	Assumptions
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
A.VEHICLE_ROOT_KEY	OE.VEHICLE_ROOT_KEY_CONFID	This assumption is directly upheld by OE.VEHICLE_ROOT_KEY_CONFID

Table 15 Security Objectives for the Operational Environment and Assumptions – Coverage

Security Objectives	Assumptions
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

Security Objectives	Assumptions
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

5 SECURITY REQUIREMENTS

This Protection Profile uses the following text formatting to make the Security Functional Requirements (SFRs) operations more visible to the reader and to simplify the work for the Security Target writer. It highlights how the specific instantiations in the SFRs are derived from the functional components in Part 2 of the CC.

- The **refinement** operation is used to add detail to a requirement and thus further restricts a requirement. Refinements of security requirements are denoted in such a way that added words are in bold text and removed words are crossed out. If a refinement is added as a separate paragraph to an SFR instead of modifying its wording, this paragraph starts with the word "Refinement:" in **bold text**.
- The **selection** operation is used to select one or more options provided by the CC in stating a requirement. Selections having been made by the PP author are denoted as underlined text and in addition a footnote will show the original text from CC, Part 2. Selections to be filled in by the ST author appear in square brackets with an indication that a selection is to be made [selection:] and are *italicized*.
- The **assignment** operation is used to assign a specific value to an unspecified parameter such as the length of a password. Assignments having been made by the PP author are denoted as underlined text and in addition a footnote will show the original text from CC, Part 2. Assignments to be filled in by the ST author appear in square brackets with an indication that an assignment is to be made [assignment:] and are *italicized*. In some cases, the assignment made by the PP authors defines a selection to be performed by the ST author, indicated by [selection:] and text, which is underlined and *italicized* like this.
- The **iteration** operation is used when a component is repeated with varying operations. The fact, that an iteration operation was used is obvious from the fact, that a component is contained (at least) twice in the PP. In order to distinguish the individual instances of a component, the component title is amended by showing a slash "/" and an individual name after the component identifier.
    - **Note:** For the sake of a better readability this notion may also be applied to some single components (being not repeated) in order to indicate that these SFRs belong to the same functional cluster.

In this PP, not all SFRs operations are completed. These are delegated to the author of a complying ST. However, our goal is to provide sufficient information to authors of complying STs, that in the end the operations completed in the ST reflect at least the amount of information provided by the security objectives of the PP. In order to achieve this, the following options for each operation in an SFR are used:

- If there are no restriction for possible completions by the ST author, this PP leaves the operation completely open;
- When the reader notice a given operation is partly completed, it means that the ST author is left only with a restricted choice.
- The ST author can complete the operation already defined in the PP.

Finally, **Application Notes** are also used to give other types of information to authors of complying STs or PPs. For example, guidance on how an ST author can apply the SFR in the

specific context of the TOE or simply translates the SFR into a complete natural language description relating it to the relevant Security Objective of the TOE.

5.1 Extended Components definition

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.

5.1.1 *FCS_RNG Random Number Generation*

This family defines quality requirements for the generation of random numbers which are intended to be use 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*].**

5.2 SECURITY FUNCTIONAL REQUIREMENTS

This section describes the requirements imposed on the TOE in order to achieve the security objectives laid down in the previous chapter. All the requirements identified in this section are instances of those stated in [CC2].

The SFRs listed below state requirements specific to the DK Applet part of the composite TOE.

5.2.1 *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,

[3] Refer to [AIS20] and [AIS31]

- cryptographic key distribution and
- cryptographic key destruction.

FCS_CKM.1 Cryptographic key generation | EC Point Generation

Hierarchical to:	**No other components.**
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction
FCS_CKM.1.1/ECC	The TSF SHALL generate cryptographic keys in accordance with a specified cryptographic key generation algorithm ECC with P-256 (SECP256r1)[4] and specified cryptographic key sizes 256-bit[5] that meet the following standards: [Selection: *[BSI TR-03111], ANSI X9.62 [X9.62a], ANSI X9.63 [X9.63], [FIPS PUB 186-4]*]

Application Note 8

- The keys can be generated and diversified in accordance with Java Card specification in classes KeyPair (at least Session key generation) and RandomData
- This component SHALL be instantiated according to the version of the Java Card API applying to the security target and the implemented algorithms.
- 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). The choice of iteration was made to simplify the mapping to relevant TSFIs and testing procedures that should be provided by the vendor as required coverage evidence.

FCS_CKM.1 Cryptographic key generation | Secure Channel

Hierarchical to:	**No other components.**
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction
FCS_CKM.1.1/Session_keys	The TSF SHALL generate cryptographic keys in accordance with a specified cryptographic key generation algorithm HKDF-SHA-256[6] and specified cryptographic key sizes 256-bit[7] that meet the following standards: IETF [RFC 5869][8]

[4] **[assignment: *cryptographic key generation algorithm*]**
[5] **[assignment: *cryptographic key sizes*]**
[6] **[assignment: *cryptographic key generation algorithm*]**
[7] **[assignment: *cryptographic key sizes*]**
[8] **[assignment: *list of standards*].**

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[9] and specified cryptographic key sizes 256-bit[10] that meet the following standards: IETF [RFC 5869][11]
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[12] and specified cryptographic key sizes 256-bit[13] that meet the following standards: IETF [RFC 5869][14]

FCS_RNG. 1 Random number generation

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*] [15] 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[16] that meet [assignment: *a defined quality metric*].

FCS_CKM.2 Cryptographic key distribution | Key Establishment

Hierarchical to:	**No other components.**
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_CKM.2.1/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[17] that meets the following: NIST Special Publication 800-56A Revision 3 with approved groups from Appendix D[18].

Application Note 9	This is a refinement of the SFR FCS_CKM.2 to deal with key establishment/agreement rather than key distribution. The ST author

[9] **[assignment: *cryptographic key generation algorithm*]**
[10] **[assignment: *cryptographic key sizes*]**
[11] **[assignment: *list of standards*].**
[12] **[assignment: *cryptographic key generation algorithm*]**
[13] **[assignment: *cryptographic key sizes*]**
[14] **[assignment: *list of standards*].**
[15] Refer to [AIS20] or [AIS31]

[17] **[assignment: *cryptographic key distribution method*]**
[18] **[assignment: *list of standards*].**

selects all key establishment schemes used for the selected cryptographic protocols.

FCS_CKM.4 Cryptographic key destruction

Hierarchical to:	**No other components.**
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4.1	The TSF SHALL destroy cryptographic keys in accordance with a specified cryptographic key destruction method [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]* that meets the following: None[19].

5.2.2 *Cryptographic Operation*

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 Cryptographic operation

Hierarchical to:	**No other components.**
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
F*CS_COP*.1.1 / Hash	The TSF SHALL perform Cryptographic Hashing[20] in accordance with a specified cryptographic algorithm SHA-256[21] and cryptographic key sizes None[22] that meet the following**:** [selection*: ISO/IEC 10118-3:2018, FIPS 180-4*]**.**

[19] **[assignment: *list of standards*]**
[20] **[assignment: *list of cryptographic operations*]**
[21] **[assignment: *cryptographic algorithm*]**
[22] **[assignment: *cryptographic key sizes*]**

FCS_COP.1.1 / HMAC	The TSF SHALL perform Keyed Hash Message Authentication[23] in accordance with a specified cryptographic algorithm HMAC-SHA-256[24], and cryptographic key sizes 256-bit[25] that meet the following: ISO/IEC 9797-2:2011[26].
FCS_COP.1.1 / E*ncryption/decryption*	The TSF SHALL perform data encryption or decryption [27] in accordance with a specified cryptographic algorithm AES with CBC mode of operation[28] and cryptographic key sizes 128-bit[29] that meet the following: FIPS PUB 197, NIST SP 800-38A[30].
FCS_COP.1.1 / *CMAC*	The TSF SHALL perform Message Authentication Code [31] in accordance with a specified cryptographic algorithm AES CMAC[32] and cryptographic key sizes 128-bit[33] that meet the following: NIST SP 800-38B[34].
FCS_COP.1.1 / E*CDSA*	The TSF SHALL perform digital signing[35] accordance with a specified cryptographic algorithm ECDSA with NIST P-256 curve [36] and cryptographic key sizes 256-bit [37] that meet the following: ANSI X9.62[38].

5.2.3 *Access Control Policy | Security Domain*

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:

Table 16 Access control SFP - SD_SFP

Type	Short name	Definition
Subjects[39]	S.INSTALLER,	The installer is the on-card entity which acts on behalf of the card issuer. This subject is

[23] **[assignment: *list of cryptographic operations*]**
[24] **[assignment: *cryptographic algorithm*]**
[25] **[assignment: *cryptographic key sizes*]**
[26] **[assignment: *list of standards*]**
[27] **[assignment: *list of cryptographic operations*]**
[28] **[assignment: *cryptographic algorithm*]**
[29] **[assignment: *cryptographic key sizes*]**
[30] **[assignment: *list of standards*]**
[31] **[assignment: *list of cryptographic operations*]**
[32] **[assignment: *cryptographic algorithm*]**
[33] **[assignment: *cryptographic key sizes*]**
[34] **[assignment: *list of standards*]**
[35] **[assignment: *list of cryptographic operations*]**
[36] **[assignment: *cryptographic algorithm*]**
[37] **[assignment: *cryptographic key sizes*]**
[38] **[assignment: *list of standards*]**
[39] 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

Type	Short name	Definition
	(from [PP0099])	involved in the loading of packages and installation of applets
	S.CAD (from [PP0099])	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 |GP] for: • loading (Section 9.3.5 of [GP]); • installation (Section 9.3.6 of [GP]); • extradition (Section 9.4.1 of [GP]); • registry update (Section 9.4.2 of [GP]); • content removal (Section 9.5 of [GP]).

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.

FDP_ACC.2 Complete access control

Hierarchical to:	FDP_ACC.1 Subset access control
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.2.1	The TSF SHALL enforce the SD_SFP[40] on *all subjects, objects defined by the SD_FSP*[41] 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

5.2.4 *Access Control Functions | Security Domain*

FDP_ACF.1 Security attribute base access control

Hierarchical to:	**No other components.**
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1	The TSF SHALL enforce the SFP_AC[42] to objects based on the following: All subjects and objects together with their respective security attributes as defined in SD_SFP[43].
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[44].
FDP_ACF.1.3	The TSF SHALL explicitly authorise access of subjects to objects based on the following additional rules: [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*].
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[45]

Application Note 10	The dependency FMT_MSA.3 will not be fulfilled here, since there is no initialisation of attributes necessary.

[40] [assignment: *access control SFP*]
[41] [assignment: *list of subjects and* ***objects***]
[42] **[assignment: *access control SFP*]**
[43] **[assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]**
[44] **[assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]**
[45] **[assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]**

5.2.5 *Information Flow Control Policy | Secure Channel Protocol*

Table 17 Information Flow Control SFP - SC_SFP

Type	Short name	Definition
Subjects	S.INSTALLER, (from [PP0099])	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 [PP0099])	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.
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 |GP] for: • loading (Section 9.3.5 of [GP]); • installation (Section 9.3.6 of [GP]); • extradition (Section 9.4.1 of [GP]); • registry update (Section 9.4.2 of [GP]); • content removal (Section 9.5 of [GP]).

FDP_IFC.2 Complete Information Flow Control

Hierarchical to:	FDP_IFC.1 Subset information flow control
Dependencies:	FDP_IFF.1 Simple security attributes
FDP_IFC.2.1/SCP	The TSF SHALL enforce the SCP_SFP[46] on subjects, information and operations[47] 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.

[46] [assignment: *information flow control SFP*]
[47] [assignment: *list of subjects and* ***information***]

5.2.6 *Information Flow Control Functions | Secure Channel Protocol*

FDP_IFF.1 Simple security attributes

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation
FDP_IFF.1.1/SCP	The TSF SHALL enforce the SCP_SFP[48] 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 [GP] **and [assignment: *list of additional security attributes*]**[49].
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 following rules hold: Rules R_GPF as defined by the SCP_SFP[50].
FDP_IFF.1.3/SCP	The TSF SHALL enforce the [assignment: *additional information flow control SFP rules*].
FDP_IFF.1.4/SCP	The TSF SHALL explicitly authorise an information flow based on the following rules: [assignment: *rules, based on security attributes, that explicitly authorise information flows].*
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[51].

Application Note 11	For authors of complying STs: In order to allow a more detailed specification, further security attributes may be added as suitable.

Application Note 12	The on-card and the off-card subjects have security attributes such as MAC, Cryptogram, Challenge, Key Set, Static Keys, etc.

Application Note 13	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

5.2.7 *Residual information protection (FDP_RIP)*

FDP_RIP.1 Subset residual information protection

Hierarchical to:	No other components.
Dependencies:	No dependencies.

[48] **[assignment: *information flow control SFP*]**
[49] **[assignment: *list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP*].**
[50] **[assignment: *for each operation, the security attribute-based relationship that must hold between subject and information security attributes*]**
[51] **[assignment: *rules, based on security attributes, that explicitly deny information flows*].**

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[52] the following objects: Cryptographic buffers[53].

Application Note 14	Cryptographic Buffers can be Cryptographic data used in runtime cryptographic computations, like a seed used to generate a key

5.2.8 *Stored data integrity (FDP_SDI)*

FDP_SDI.2 Stored data integrity monitoring and action

Hierarchical to:	FDP_SDI.1 Stored data integrity monitoring
Dependencies:	No dependencies.
FDP_SDI.2.1	The TSF SHALL monitor user data stored in containers controlled by the TSF for [assignment**:** *integrity errors***]** on all objects, based on the following attributes: [assignment: *user data attributes*].
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[54].

5.2.9 *Inter-TSF user data integrity transfer protection*

FDP_UIT.1 Data exchange Integrity | Card Content Management

Hierarchical to:	**No other components**
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path]
FDP_UIT.1.1/CCM	The TSF SHALL enforce the Secure channel protocol Information flow control policy[55] to [selection: *transmit, receive*] user data in a manner protected from [selection: *modification, deletion, insertion, replay***]** errors.
FDP_UIT.1.2/CCM	The TSF SHALL be able to determine on receipt of user data, whether modification, deletion, insertion, replay[56] has occurred.

5.2.10 *Identification and Authentication*

FIA_UAU.3 Unforgeable authentication

Hierarchical to:	**No other components**
Dependencies:	No dependencies.
FIA_UAU.3.1	The TSF SHALL [selection: *detect, prevent*] use of authentication data that has been forged by any user of the TSF.

[52] **[selection: *allocation of the resource to, deallocation of the resource from*]**
[53] **[assignment: *list of objects*].**
[54] **[assignment: *action to be taken*].**
[55] **[assignment: *access control SFP(s) and/or information flow control SFP(s)*]**
[56] **[selection: *modification, deletion, insertion, replay*]**

FIA_UAU.3.2	The TSF SHALL [selection: *detect, prevent*] use of authentication data that has been copied from any other user of the TSF.

5.2.11 *Security Management | TSF data*

FMT_MTD.1 Management of TSF data

Hierarchical to:	**No other components**
Dependencies:	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MTD.1.1/deletion of keys	**The TSF SHALL restrict the ability to**, delete[57], the keys[58] to Digital Key framework, [**ass**ignment: *other authorised identified roles*] [59].

FMT_MTD.3 Secure TSF data

Hierarchical to:	**No other components**
Dependencies:	FMT_MTD.1 Management of TSF data
FMT_MTD.3.1	The TSF SHALL ensure that only secure values are accepted for the applet's AID[60].

Application Note 15	The value of the Applet's AID is defined in [CCC-DK-TS], Section 15.3.2.1.

5.2.12 *Specifications of Management Functions | TSF data*

FMT_SMF.1 Specification of Management Functions

Hierarchical to:	**No other components**
Dependencies:	No dependencies
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[61].

FMT_SMR.1 Security Roles

Hierarchical to:	**No other components**
Dependencies:	FIA_UID.1 Timing of identification
FMT_SMR.1.1	The TSF SHALL maintain the roles Digital Key framework, Vehicle, [**ass**ignment: *other authorised identified roles*] [62].
FMT_SMR.1.2	The TSF SHALL be able to associate users with roles.

Application Note 16	

[57] **[selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]**
[58] **[assignment: *list of TSF data*]**
[59] **[assignment: the authorized identified roles]**
[60] **[assignment: *list of TSF data*].**
[61] **[assignment: *list of management functions to be provided by the TSF*].**
[62] **[assignment: the authorized identified roles]**

The dependency to FIA_UID.1 is not applicable to this TOE. This PP does not require the identification of the roles to be assigned which is handled by the operational environment.

5.2.13 *Unlinkability*

FPR_UNL.1 Unlinkability

Hierarchical to:	**No other components**
Dependencies:	No dependencies
FPR_UNL.1.1/NFC	**FPR_UNL.1.1 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.**

5.2.14 *Protection of the TSF*

FPT_ITC.1 Inter-TSF confidentiality during transmission

Hierarchical to:	**No other components**
Dependencies:	No dependencies
FPT_ITC.1.1	The TSF SHALL protect all TSF data transmitted from the TSF to another trusted IT product from unauthorised disclosure during transmission.

FPT_ITI.1 Inter-TSF detection of modification

Hierarchical to:	**No other components**
Dependencies:	No dependencies
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[63].
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[64] if modifications are detected.

[63] **[assignment: *a defined modification metric*]**
[64] **[assignment: *action to be taken*]**

5.2.15 *Internal TOE TSF data transfer (FPT_ITT)*

FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical to:	**No other components**
Dependencies:	No dependencies
FPT_ITT.1.1/ IMMO_TOKEN	The TSF SHALL protect TSF data from disclosure[65] when it is transmitted between separate parts of the TOE.
FPT_ITT.1.1/ DIGITAL_KEY	The TSF SHALL protect TSF data from disclosure[66] when it is transmitted between separate parts of the TOE.

FPT_ITT.3 TSF data integrity monitoring

Hierarchical to:	**No other components**
Dependencies:	No dependencies
FPT_ITT.3.1/ DIGITAL_KEY	The TSF SHALL be able to detect [selection: *modification of data, substitution of data, re-ordering of data, deletion of data*, [assignment: *other integrity errors*]] for TSF data transmitted between separate parts of the TOE.
FPT_ITT.3.1/IMMO_TOKEN	The TSF SHALL be able to detect [selection: *modification of data, substitution of data, re-ordering of data, deletion of data*, [assignment: *other integrity errors*]] 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: **[**assignment: *specify the action to be taken*].
FPT_ITT.3.2/IMMO_TOKEN	Upon detection of a data integrity error, the TSF SHALL take the following actions: **[**assignment: *specify the action to be taken*].

5.2.16 *Replay Detection*

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.

[65] **[selection: *disclosure, modification*]**
[66] **[selection: *disclosure, modification*]**

FPT_RPL.1 Replay detection

Hierarchical to:	**No other components**
Dependencies:	No dependencies
FPT_RPL.1.1	The TSF SHALL detect replay for the following entities: S.Transaction data[67].
FPT_RPL.1.2	The TSF SHALL perform terminate the transaction[68] when replay is detected.

5.2.17 *Trusted Recovery*

FPT_RCV.2 Automated recovery

Hierarchical to:	FPT_RCV.1 Manual recovery
Dependencies:	AGD_OPE.1 Operational user guidance
FPT_RCV.2.1	When automated recovery from power failure[69] 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[70], the TSF SHALL ensure the return of the TOE to a secure state using automated procedures.

5.2.18 *TSF Self-Tests*

Application Note 17	Startup of the TOE (TSF-testing) can be covered by FPT_TST.1. This SFR component is not mandatory in [PP0099], but appears in most of security requirements documents for masked applications. Testing could also occur randomly. Self-tests may become mandatory in order to comply with FIPS certification [FIPS140-3]

[67] **[assignment: *list of identified entities*]**
[68] **[assignment: *list of specific actions*]**
[69] [assignment: *list of failures/service discontinuities*]
[70] **[assignment: *list of failures/service discontinuities*]**

5.2.19 *Inter-TSF Trusted Channel*

FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to:	**No other components**
Dependencies:	No dependencies
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[71] 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[72].

5.2.20 *Physical Resistance*

FPT_PHP.3 Resistance to physical attack

Hierarchical to:	**No other components**
Dependencies:	No dependencies
FPT_PHP.3.1	The TSF SHALL resist physical manipulation and physical probing[73] to the TSF[74] by responding automatically such that the SFRs are always enforced.

Application Note 18	This SFR is being included to the PP to highlight the possibility that security features implemented on the application level could be required to support the detection of physical tampering provided by the FPT_PHP.3.1 of the IC [PP0084]. In that case the Security Target writer SHALL refine this SFR and map it to the related security functionality in the TSS.

5.3 SECURITY ASSURANCE REQUIREMENTS

The security assurance requirement level is EAL4 augmented with AVA_VAN.5 and ALC_DVS.2 covering all the Security Assurance components highlighted in the table below.

[71] **[selection:** ***the TSF, another trusted IT product*****]**
[72] **[assignment:** ***list of functions for which a trusted channel is required*****]**
[73] **[assignment:** ***physical tampering scenarios*****]**
[74] **[assignment:** ***list of TSF devices/elements*****]**

Figure 5 EAL 4 augmented with ALC_DVS.2 and AVA_VAN.5

Assurance class	Assurance Family	Assurance Components by Evaluation Assurance Level						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
Development	ADV_ARC		1	1	1	1	1	1
	ADV_FSP	1	2	3	4	5	5	6
	ADV_IMP				1	1	2	2
	ADV_INT					2	3	3
	ADV_SPM						1	1
	ADV_TDS		1	2	3	4	5	6
Guidance documents	AGD_OPE	1	1	1	1	1	1	1
	AGD_PRE	1	1	1	1	1	1	1
Life-cycle support	ALC_CMC	1	2	3	4	4	5	5
	ALC_CMS	1	2	3	4	5	5	5
	ALC_DEL		1	1	1	1	1	1
	ALC_DVS			1	1	1	2	2
	ALC_FLR							
	ALC_LCD			1	1	1	1	2
	ALC_TAT				1	2	3	3
Security Target evaluation	ASE_CCL	1	1	1	1	1	1	1
	ASE_ECD	1	1	1	1	1	1	1
	ASE_INT	1	1	1	1	1	1	1
	ASE_OBJ	1	2	2	2	2	2	2
	ASE_REQ	1	2	2	2	2	2	2
	ASE_SPD		1	1	1	1	1	1
	ASE_TSS	1	1	1	1	1	1	1
Tests	ATE_COV		1	2	2	2	3	3
	ATE_DPT			1	1	3	3	4
	ATE_FUN		1	1	1	1	2	2
	ATE_IND	1	2	2	2	2	2	3
Vulnerability assessment	AVA_VAN	1	2	2	3	4	5	5

5.4 SECURITY REQUIREMENTS RATIONALE

5.4.1 *Rationale for the Security Functional Requirements*

Table 18 Security Objectives and SFRs - Coverage

Security Objectives	SFR	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,

Security Objectives	SFR	Rationale
		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: • FPT_ITT.3 which enforces that the immobilizer token transmitted between separate parts of the TOE

Security Objectives	SFR	Rationale
		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:

Security Objectives	SFR	Rationale
		• 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 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.

Security Objectives	SFR	Rationale
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 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

Security Objectives	SFR	Rationale
		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.

Security Objectives	SFR	Rationale
		• 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 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

Security Objectives	SFR	Rationale
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 same user.

5.4.2 *Rationale for the Exclusion of Dependencies*

The dependency to FIA_UID.1 is not applicable to this TOE. This PP does not require the identification of the roles to be assigned which is handled by the operational environment.

5.4.3 *Rationale for the Security Assurance Requirements*

EAL4 allows a developer to attain a reasonably high assurance level without the need for highly specialized processes and practices. It corresponds to a white box analysis and it can be considered as a reasonable level that can be applied to an existing product line without undue expense and complexity.

The TOE is intended to operate in open environments, where attackers can easily exploit vulnerabilities. According to the claimed intended usage of the TOE, it is very likely that it may represent a significant value and then constitute an attractive target for attacks. In some malicious usages of the TOE the statistical or probabilistic mechanisms in the TOE, for instance, may be subjected to analysis and attack in the normal course of operation.

For the matter of fact, we present you below some concerns from the Vehicle industry clarifying the potential of attackers.

5.4.3.1 General relevance of Motor Vehicle Crime

Interpol issued a report in 2014 that provides an overview titled "Motor vehicle crime in global perspective" which provides an overview e.g. of the type of crimes, the estimated annual damage (multi-billion Dollar per Year) and the actors involved.

Beyond the commercial damage, Interpol raises also the relevance of Motor vehicle crime for global crime incl. terrorism: "In fighting transnational organized crime, however, stolen motor vehicles should, in many cases, literally be seen as the “vehicle” of the crime. Stolen vehicles are found to be the way of transport for bank robbers; illegal drugs are paid for with stolen vehicles; victims of trafficking in human beings are being discovered in stolen vehicles and car bombs are traditionally hidden in a stolen vehicle."

5.4.3.2 Attacker profile

Experiences from international law enforcement organizations and vehicle OEMs prove that attacks on the information security assets are also conducted by criminal organizations that have established all necessary means to market stolen cars or parts internationally. These criminal organizations may have extensive financial resources and organisational skills.

5.4.3.3 Attacker motivation

Above mentioned criminal organizations which have established ways to market stolen cars or parts internationally can expect immense income from their criminal activities. Quote from Interpol: Beyond that Factors that influence the cost/benefit calculation of the attacker (quote Interpol):

- A relatively small investment requirement for the necessary tools to commit the crime (this is what we should change!);
- In comparison to other crimes, there is a generally mild punishment if convicted;
- The ample supply and opportunity in origin areas in combination with plenty of prospective customers in destination areas.

5.4.3.4 Capabilities of the assumed attacker

Based on the before mentioned attacker profile, we have to assume organized crime as attacker. Organized crime is generally considered **capable to conduct attacks of level "high"**. The following list of potential capabilities (which reflects a similar methodology which is used in evaluation) may explain this**:**

- The attacker may have the financial resources and organizational skills to employ teams of experts and to conduct "Multiple Expert" attacks.
- The attacker may have the financial resources and organizational skills to get access to highest level equipment and to conduct "Multiple bespoke tool" attacks.
- The attacker may have the financial resources and organizational skills to get access to "critical" knowledge of the components (TOE). This may be conducted by bribing or putting employees of DK stakeholders or their suppliers under pressure so that they provide confidential specifications, cryptographic secrets etc.
- The attacker may have the financial resources and organizational skills to get access to a large number of TOE (e.g. devices or even the relevant parts of vehicles) and to perform attacks on this large number of TOE in parallel.

- The attacker may have the financial resources and organizational skills to get access to "Open Samples" of the components (TOE). This may be conducted by bribing or putting employees of DK stakeholders or their suppliers under pressure so that they provide these open components (e.g. engineering samples of SE).
- The attacker may have the financial resources and organizational skills to develop and insert malware into components. This may be conducted by bribing or putting employees of DKS stakeholders or their suppliers under pressure.
- The attacker may have the organizational and technical skills to conduct concatenated attack scenarios e.g. by obtaining confidential information first, weakening certain functions of the system (e.g. key generation or implementing malware) and finally to perform attacks on the weakened device.

5.4.3.5 Likelihood of attacks

Based on the motivation of potential attackers described before it SHALL be assumed that attacks will be conducted immediately after market introduction of components and that attacks will be performed permanently. It is probable that attackers will focus on stakeholders (i.e. device or vehicle OEMs) who enter the system. The overall likelihood of attacks SHALL be assumed as "high".

Based on all the assumptions presented above an EAL 4 augmented with ALC_DVS.2 and AVA_VAN.5 seems to be the reasonable minimum level for a sensitive application like the DK applet.

5.4.3.5.1 AVA VAN.5 Advanced methodical vulnerability analysis

This component added to EAL 4 package in order to provide sufficient robustness to counter an attacker with high attack potential without the support of a protecting environment. Moreover, the DK applet is a sensitive one handling valuable assets such as an expensive vehicle. Potential attackers for such kind of applications include international organizations, or even a state, disposing of important means and resources. Finally, the evaluator will base their evaluation methodology addressing vulnerability assessment on JIL Application of Attack Potential to Smart Cards and Similar Devices [JIL-attacks].

5.4.3.5.2 ALC_DVS.2 Sufficiency of security measures

This component was added in order to provide a higher assurance on the security of the DK applet development and SE manufacturing processes, especially for the secure handling of the embedded software and data. Those requirements appear as the most adequate ones for a manufacturing process in which several actors (Platform Developer, Operator, Application Developers, IC Manufacturer, etc.) exchange and store highly sensitive information (confidential code, cryptographic keys, personalisation data, etc.).

REFERENCES

Short Name	Description
[AIS20]	Functionality classes and evaluation methodology for deterministic random number generators, reference: AIS 20, BSI (latest version)
[AIS31]	Functionality classes and evaluation methodology for physical random number generators, reference: AIS31, BSI (latest version)
[BSI TR-03111]	Technical Guideline BSI TR-03111 - Elliptic Curve Cryptography - Version 2.10- Date: 2018-06-01
[CC]	Common Criteria for Information Technology Security Evaluation documents version 3.1, revision 5 - Parts 1, 2 and 3 - https://www.commoncriteriaportal.org/cc/
[CC1]	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model. Version 3.1. Revision 5. April 2017. CCMB-2017-04-001.
[CC2]	Common Criteria for Information Technology Security Evaluation, Part 2: Security functional requirements. Version 3.1. Revision 5. April 2017. CCMB-2017-04-002.
[CC3]	Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance requirements. Version 3.1. Revision 5. April 2017. CCMB-2017-04-003.
[CCC-DK-TS]	Car Connectivity Consortium Digital Key Release 3 - Technical Specification Version 1.1.0 (CCC-TS-101)
[CCC-FS-DK]	CCC-WP-xxx-Digital-Key-Functional-Certification-Applet-R2_0.23
[CEM]	Common Methodology for Information Technology Security Evaluation, Evaluation Methodology. Version 3.1. Revision 5. April 2017. CCMB-2017-04-004.
[FIPS140-3]	Security Requirements for Cryptographic Modules - FIPS PUB 140-3 Federal Information Processing Standards Publication (Supersedes FIPS PUB 140-2)
[FIPS PUB 186-4]	The FIPS 186-4 Elliptic Curve Digital Signature Algorithm Validation System (ECDSA2VS) - March 18, 2014
[FIPS PUB 197]	Advanced Encryption Standard (AES) – FIPS 197 – November 26, 2001
[GP]	GlobalPlatform Specifications version 2.3.1 https://globalplatform.org/wp-content/uploads/2018/05/GPC_CardSpecification_v2.3.1_PublicRelease_CC.pdf
[JIL-attacks]	JIL-Application-of-Attack-Potential-to-Smartcards-v3-1
[PP0084]	Security IC Platform Protection Profile with Augmentation Packages - Registered and Certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-CC-PP-0084-2014
[PP0099]	Java Card System – Open Configuration Protection Profile – April 2020 – Version 3.1
[RFC2119]	Key words for use in RFCs to Indicate Requirement Levels, Updated by: 8174 : http://www.ietf.org/rfcs/rfc2119.txt

Short Name	Description
[RFC 5869]	IETF - HMAC-based Extract-and-Expand Key Derivation Function (HKDF) May 2010
[SP-800-38A]	Recommendation for Block Cipher Modes of Operation: Methods and Techniques – NIST Special Publication 800-38A – December 2001
[SP-800-38B]	Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication – NIST Special Publication 800-38B – May 2005
[SP-800-56A]	Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography - SP 800-56A Rev. 3 - April 2018
[X9.62a]	The Elliptic Curve Digital Signature Algorithm (ECDSA), American National Standard for Financial Services, November 16, 2005
[X9.63]	Public Key Cryptography for the Financial Services Industry - Key Agreement and Key Transport Using Elliptic Curve Cryptography