DigitalKey applet v3 JxU on JCOP 6.2 SN220

Security Target Lite

Rev. 1.0 — 5 April 2024

CCC-DK-23-AC0C00E9

Evaluation document

Document information

Information	Content
Keywords	Common Criteria, Security Target, DigitalKey, CCC, JCOP 6.2, SN220
Abstract	Evaluation of the DigitalKey v3 JxU applet for JCOP 6.2 on SN220 secure element, developed and provided by NXP Semiconductors, Business Line Secure Connected Edge, according to the Common Criteria for Information Technology Evaluation Version 3.1 R5 at EAL4 augmented.

Revision History

Rev.	Date	Description
1.0	2024-04-05	First release Security Target Lite based on Security Target v1.2.

1 Introduction

1.1 ST Reference

Table 1. ST Reference

ST Title	DigitalKey applet v3 JxU on JCOP 6.2 SN220
ST Version	Revision 1.0
ST Date	5 April 2024

1.2 TOE Reference

Table 2. TOE Reference

TOE Name	DigitalKey applet v3 JxU on JCOP 6.2 SN220
Applet Version	3.12.1.1JxU
Applet Identification	4003030C01

The TOE is a composite TOE based on the certified JavaCard OS platform and certified hardware IC referenced in the tables below.

Table 3. Platform Reference

Platform Name	NXP JCOP 6.2 on SN220 Secure Element
Platform Version	R1.01.1, R1.02.1, R2.01.1
Platform Certificate	NSCIB-CC-22-0428888
Platform ST Reference	NXP JCOP 6.2 on SN220 Secure Element, Security Target, Product evaluation document, NXP Semiconductors, Revision 1.8, 24 November 2022 [10].

Table 4. IC Reference

IC Name	SN220 Series - Secure Element with Crypto Library
IC Hardware Version	B0.1 C13, B0.1 C37
IC Certificate	NSCIB-CC-22-0258298
IC ST Reference	SN220 Series - Secure Element with Crypto Library, Security Target, Evaluation document, NXP Semiconductors, Revision 1.5, 29 September 2022 [11].

1.3 TOE Overview

The Target of Evaluation (TOE) is a Java Card applet implementing the Digital Key Technical Specification Release 3 [14] as specified by the Car Connectivity Consortium LLC. The applet is to be installed on the JCOP 6.2 Java Card platform and SN220 Secure Element, as referenced in Section 1.2 of this Security Target.

The TOE allows individual owners of vehicles to use their mobile devices as keys to enter their vehicles. The Digital Key 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 a public key infrastructure to establish end-to-end trust. Mobile devices create and store Digital Keys in Secure Elements to provide the highest level of protection from various 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.

Interoperability between mobile devices and vehicles is supported by standardizing the vehicle-to-device interface, based on a NFC communication channel, protocols, and Digital Key structures.

The TOE 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 different use cases of the TOE include:

- **Owner Pairing:** A mobile device containing the applet can be paired as an owner device with the vehicle. 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. The limited operational range of NFC prevents attackers from tricking the vehicle into thinking that your mobile device is nearby when it’s not.
- **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).
- **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.

The TOE evaluation is based on the Digital Key Protection Profile [6], which requires that the TOE is based on a certified SE Java Card and GlobalPlatform platform, and based on a certified IC hardware Secure Element. The certification details of the platform and hardware IC used by the TOE are given in Section 1.2.

1.3.1 TOE Type

The TOE type is a Java Card applet compliant with the TOE type defined in Section 1.3 of the Protection Profile [6], to which this Security Target claims strict conformance.

1.3.2 Required non-TOE Hardware/Software/Firmware

The applet uses a Secure UWB Service API to provide UWB access to the vehicle. For the applet to function, this API needs to be loaded on the JCOP platform, even if UWB services are not used. Please note that UWB functionality is outside of the scope of this certification.

In addition, Section 1.4 of the Digital Key Protection Profile [6] discusses other required non-TOE hardware, software and firmware. An overview can be seen in Figure 1. Please see the Protection Profile for more details.

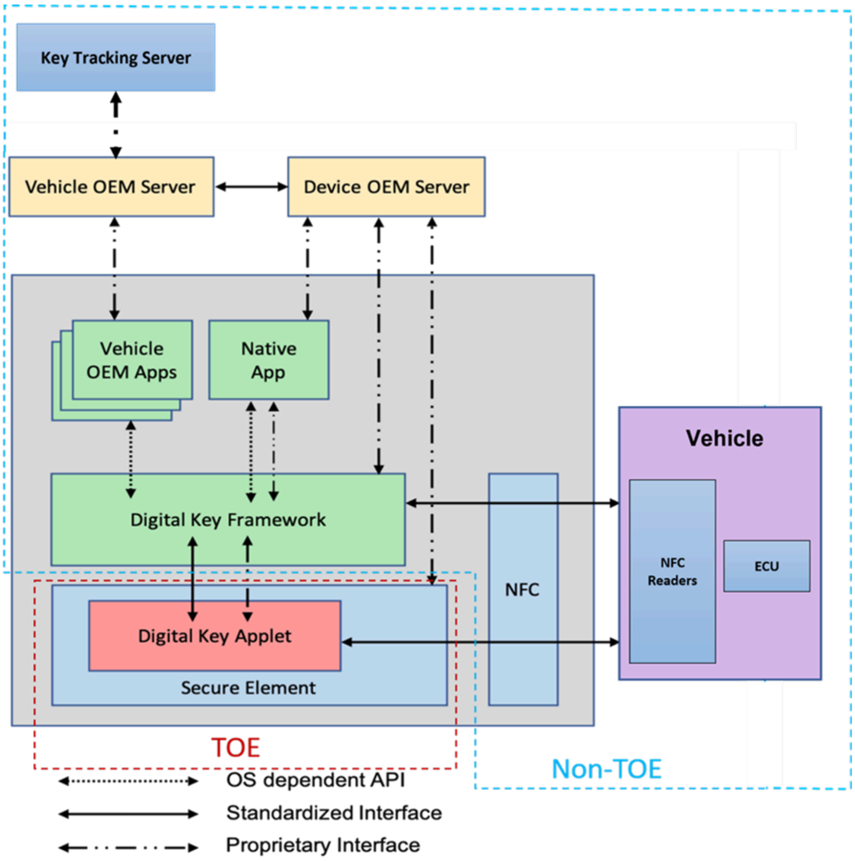

Figure 1. Non-TOE components

1.4 TOE Description

1.4.1 TOE Components

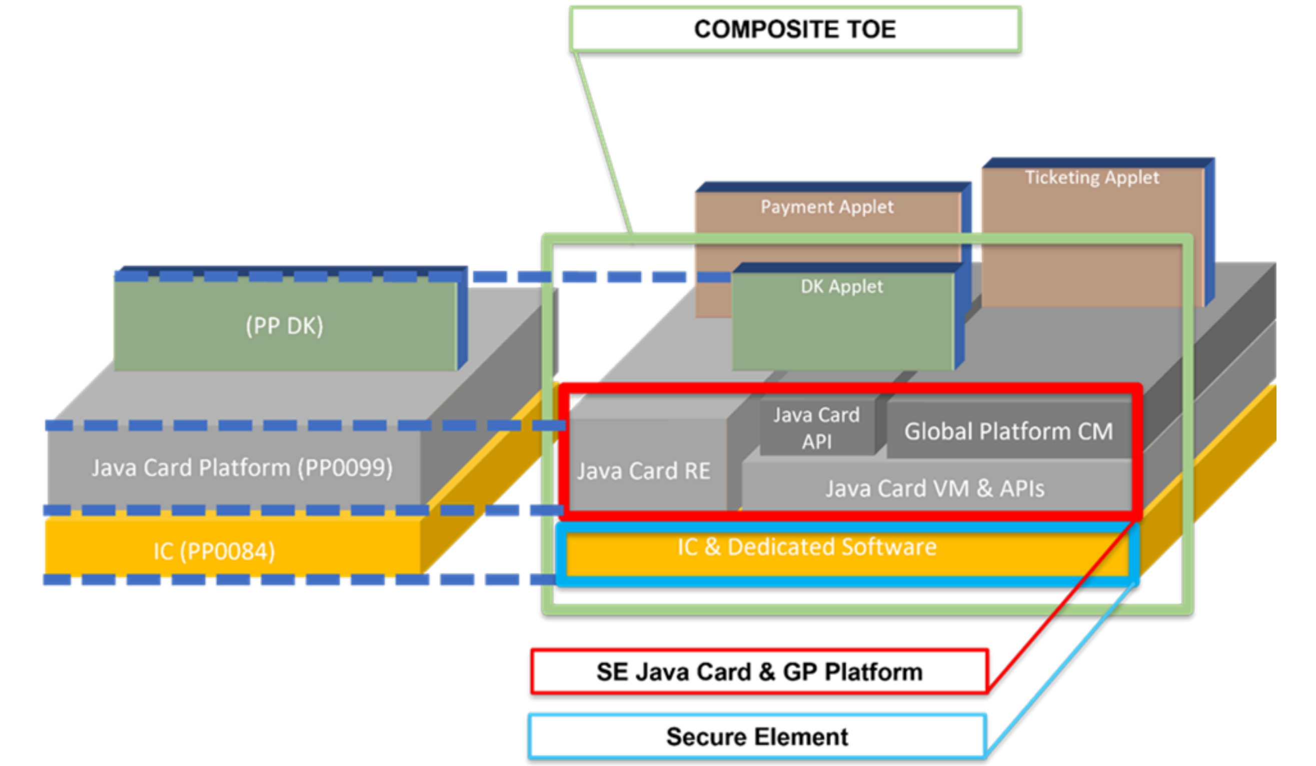

Figure 2. TOE architecture

The TOE consists of the following components:

- The Secure Element (IC) with its IC Dedicated Software (firmware and cryptographic library) providing a secured execution environment, support for cryptographic operations, random numbers, and memory management. The IC certificate (based on PP0084 [8]) is re-used for the evaluation of the TOE (see the IC Security Target [11] for more details).
- The SE Java Card & GP platform, which provides the operating system implementing the JCVM, JCRE, JCAPI and the GlobalPlatform (GP) framework. The platform certificate (Java Card OS + IC) is based on PP0099 [7] and is re-used for the evaluation of the TOE (see the platform Security Target [10] for more details).
- The Digital Key applet which is the subject of the current certification. The security functionality which is in scope of the current certification is specified by the DigitalKey Protection Profile [6].
- The TOE guidance documentation as specified in Section 1.4.2.

1.4.2 TOE Delivery

The TOE delivery comprises the following items:

Table 5. TOE deliverables

Type	Name	Release	Form of delivery
Platform	SN220 Series - Secure Element with Crypto Library NXP JCOP 6.2 on SN220 Secure Element	R1.01.1, R1.02.1, R2.01.1	Hardware modules with on-chip software

Table 5. TOE deliverables*...continued*

Type	Name	Release	Form of delivery
Applet	DigitalKey applet v3 JxU on JCOP 6.2 SN220	3.12.1.1JxU	Standalone CAP file
Document	UM11616 - CCC Digital Key Applet, User Manual [9]	1.6	Electronic document (PDF via NXP DocStore)

The platform guidance documents are referenced in the platform Security Target [10].

1.4.3 Physical Scope of the TOE

The physical scope of the TOE is the DigitalKey applet as a standalone CAP file, and the hardware IC device (Secure Element) loaded with the software components identified in Section 1.4.1 (IC Dedicated Software, Java Card Operating System). The DigitalKey applet is then installed on the device by the customer using the interfaces provided by the OS.

The physical interfaces of the TOE are the IC die surface (attacker interface) and the NFC controller interface, enabling NFC communication with the device and OS/applet.

1.4.4 Logical Scope of the TOE

The TOE provides its Digital Key services by communicating using the NFC interface provided by the platform. Some possible use cases for the TOE are described in Section 1.3. Other potential security feature of the TOE are:

- **Secure Owner Pairing:** The owner pairing flow is operated by the Digital Key framework and TOE applet running on the mobile device. The Digital Key framework uses the APDU commands of the applet to manage the configuration of the Digital Key, protected by the Secure Element. The Secure Element provides the root of trust, which is the starting point in the trust chain.
  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.
- **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. 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.
- **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.
- **Secure Check Presence Transaction:** The check presence transaction protocol is intended to provide the following properties: Vehicle authentication, Device identification, Integrity and confidentiality, and Tracking resilience.
  The mechanism is similar to the Secure Standard Transaction mechanism listed above, except that the device signature is not sent to the vehicle, and user authentication is disabled. The goal is to allow verification of the device presence near the vehicle without requiring user authentication, while preventing tracking.

- **Secure Digital Key Sharing:** All messages exchanged over the communication channels are compliant with ASN.1 DER encoding rule and use implicit tagging rules unless otherwise specified, and the certificates are compliant with X.509 v3 format. Once the communication channel is established, the Digital Key sharing protocol data is exchanged between the owner device and the friend device.
  The key sharing protocol data is encrypted using AES-256-GCM with 96 bits nonce randomly generated and using a tag length of 128 bits.
- **Key Termination and 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.
- **Secure Applet Management:** The TOE offers additional security services for applets management, relying on the GlobalPlatform framework:
  - The Secure Element 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.

1.5 TOE Life Cycle

Section 1.5 of the Protection Profile defines the TOE life cycle. An overview of the TOE life cycle can be seen in Figure 3. For more information see the Protection Profile [6].

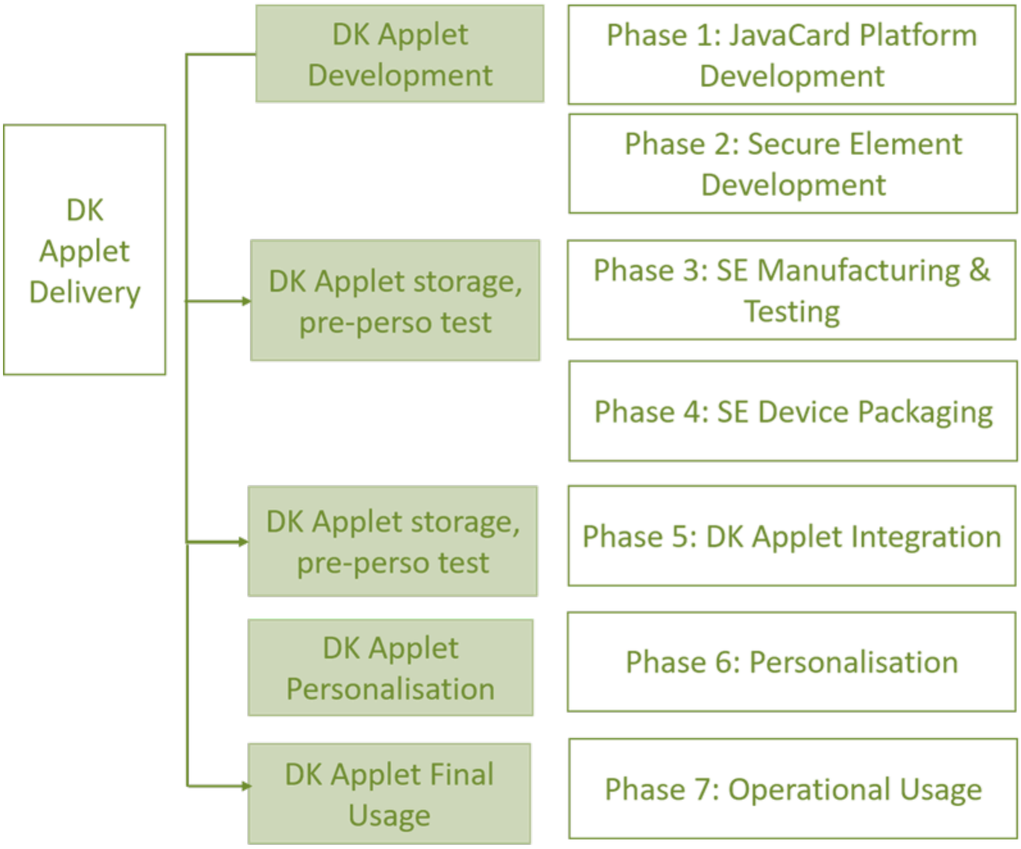
DK Applet Delivery
DK Applet Development
DK Applet storage, pre-perso test
DK Applet storage, pre-perso test
DK Applet Personalisation
DK Applet Final Usage
Phase 1: JavaCard Platform Development
Phase 2: Secure Element Development
Phase 3: SE Manufacturing & Testing
Phase 4: SE Device Packaging
Phase 5: DK Applet Integration
Phase 6: Personalisation
Phase 7: Operational Usage

Figure 3. TOE Life Cycle

2 Conformance Claims

2.1 CC Conformance Claim

This Security Target claims to be conformant to the Common Criteria version 3.1:

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, Version 3.1, Revision 5, CCMB-2017-04-001, April 2017 [2].
- Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components, Version 3.1, Revision 5, CCMB-2017-04-002, April 2017 [3].
- Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components, Version 3.1, Revision 5, CCMB-2017-04-003, April 2017 [4].

For the evaluation the following methodology will be used:

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

2.2 Protection Profile Claim

This Security Target claims strict conformance to the following Protection Profile:

- Car Connectivity Consortium Digital Key, Protection Profile of the Digital Key Applet, Version 1.0, CCC-CP-023, 16 October 2023 [6].

The Protection Profile listed above does not claim conformance to any other Protection Profile.

2.3 Package Claim

This Security Target claims conformance to the assurance package EAL4 augmented with ALC_DVS.2 and AVA_VAN.5, as required by the Protection Profile [6].

2.4 Conformance Claim Rationale

As the Protection Profile [6] requires strict conformance, no conformance claim requirement is needed in this Security Target.

3 Security Problem Definition

3.1 Assets

The assets of the TOE are strictly compliant with the assets described in Section 3.1 of the Protection Profile [6]. Please see the Protection Profile for details.

3.2 Threats

The threats to the assets of the TOE are strictly compliant with the threats described in Section 3.2 of the Protection Profile [6]. Please see the Protection Profile for details.

3.3 Organisational Security Policies

The Organisational Security Policies (OSPs) for the TOE are strictly compliant with the OSPs described in Section 3.3 of the Protection Profile [6]. Please see the Protection Profile for details.

3.4 Assumptions

The assumptions for the TOE are strictly compliant to the assumptions described in Section 3.4 of the Protection Profile [6]. Please see the Protection Profile for details.

4 Security Objectives

4.1 Security Objectives for the TOE

The Security Objectives for the TOE are strictly compliant with the Security Objectives for the TOE described in Section 4.1 of the Protection Profile [6]. Please see the Protection Profile for details.

4.2 Security Objectives for the Operational Environment

The Security Objectives for the Operational Environment are strictly compliant with the Security Objectives for the Operational Environment described in Section 4.2 of the Protection Profile [6]. Please see the Protection Profile for details.

4.3 Security Objectives Rationale

The Security Objectives rationale is strictly compliant with the Security Objectives rationale described in Section 4.3 of the Protection Profile [6]. Please see the Protection Profile for details.

5 Extended Components Definition

This Security Target does not define any additional extended components.

Note that the Protection Profile [6] defines extended security functional requirement FCS_RNG.1, which is specified in this Security Target but its definition is not duplicated in this section.

6 Security Requirements

This chapter defines the security requirements that shall be met by the TOE. These security requirements are composed of the security functional requirements and the security assurance requirements that the TOE must meet in order to achieve its security objectives.

CC allows several operations to be performed on security requirements (on the component level); refinement, selection, assignment, and iteration are defined in section 8.1 of CC Part 1 [2]. These operations are used in this Security Target.

- The refinement operation is used to add details to requirements, and thus, further intensifies a requirement.
- The selection operation is used to select one or more options provided by the Protection Profile or CC in stating a requirement. Selections having been made are denoted as *italic* text.
- The assignment operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments having been made are denoted as *italic* text.
- The iteration operation is used when a component is repeated with varying operations. For the sake of a better readability, the iteration operation may also be applied to some single components (being not repeated) in order to indicate belonging of such SFRs to same functional cluster. In such a case, the iteration operation is applied to only one single component.

Note: *This Security Target does not indicate the operations already performed in the Protection Profile [6] and will only highlight those operations which are left open.*

6.1 Security Functional Requirements

All Security Functional Requirements (SFRs) for this Security Target are required and defined by Section 5.2 of the Protection Profile [6] . These SFRs apply in their entirely to this Security Target.

Some SFRs are completely and correctly defined in the Protection Profile [6] and do not require an operation to be performed in the Security Target. These SFRs are listed in the table below.

Table 6. Security Functional Requirements completely defined in the Protection Profile

Name	Title
FCS_CKM.1/Session_keys	Cryptographic key generation | Secure Channel
FCS_CKM.1/Long_Term_key	Cryptographic key generation | Secure Channel
FCS_CKM.1/Secret_Shared_key	Cryptographic key generation | Secure Channel
FCS_COP.1/HMAC	Cryptographic operation
FCS_COP.1/Encryption/decryption	Cryptographic operation
FCS_COP.1/CMAC	Cryptographic operation
FDP_ACC.2	Complete access control
FDP_IFC.2/SCP	Complete Information Flow Control
FDP_RIP.1	Subset residual information protection
FMT_MTD.3	Secure TSF data
FMT_SMF.1	Specification of Management Functions
FPR_UNL.1/NFC	Unlinkability
FPT_ITC.1	Inter-TSF confidentiality during transmission
FPT_ITI.1/Vehicle_Integrity	Inter-TSF detection of modification
FPT_ITT.1/IMMO_TOKEN	Basic internal TSF data transfer protection

Table 6. Security Functional Requirements completely defined in the Protection Profile*...continued*

Name	Title
FPT_ITT.1/DIGITAL_KEY	Basic internal TSF data transfer protection
FPT_RPL.1	Replay detection
FPT_RCV.2	Automated recovery
FTP_ITC.1	Inter-TSF trusted channel
FPT_PHP.3	Resistance to physical attack

The definitions of the SFRs listed in the table above are not repeated in this Security Target. More information on these can be found in the Protection Profile [6].

Other SFRs defined in the Protection Profile [6] do require an operation to be performed in the Security Target. These SFRs are listed in the table below.

Table 7. Security Functional Requirements requiring operations in this Security Target

Name	Title
FCS_CKM.1/ECC	Cryptographic key generation | EC Point Generation
FCS_RNG.1	Random number generation
FCS_CKM.2/ECDHE[1]	Cryptographic key distribution | Key Establishment
FCS_CKM.4	Cryptographic key destruction
FCS_COP.1/Hash	Cryptographic operation
FCS_COP.1/ECDSA[2]	Cryptographic operation
FDP_ACF.1	Security attribute base access control
FDP_IFF.1/SCP	Simple security attributes
FDP_SDI.2	Stored data integrity monitoring and action
FDP_UIT.1/CCM	Data exchange Integrity | Card Content Management
FIA_UAU.3	Unforgeable authentication
FMT_MTD.1/deletion of keys	Management of TSF data
FMT_SMR.1	Security Roles
FPT_ITT.3/DIGITAL_KEY	TSF data integrity monitoring
FPT_ITT.3/IMMO_TOKEN	TSF data integrity monitoring
FPT_TST.1	TSF Self-Tests

[1] FCS_CKM.2/ECDHE is completely defined in the Protection Profile, but is refined to correct the standard being used.
[2] FCS_COP.1/ECDSA is completely defined in the Protection Profile, but is refined to also include signature verification in its scope.

The following sections will perform the operations on the security functional requirements listed in the table above.

6.1.1 FCS_CKM.1/ECC

The TOE shall meet the requirement "Cryptographic key generation | EC Point Generation" as specified below.

FCS_CKM.1/ECC **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) and specified cryptographic key sizes 256-bit that meets the following: *FIPS PUB 186-4* [17][1].

6.1.2 FCS_RNG.1

The TOE shall meet the requirement "Random number generation" as defined in the Protection Profile [6], and as specified below.

FCS_RNG.1	**Random number generation**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FCS_RNG.1.1	The TSF shall provide a *deterministic*[2] random number generator *DRG.3*[3] that implements: generation of strong cryptographic random numbers, key generation functions use adequate entropy source from approved random number generator(s).
FCS_RNG.1.2	The TSF shall provide random numbers that meet:[4] *(DRG.3.4) The RNG, initialized with a random seed using a PTRNG of class PTG.2 as random source, generates output for which 2^{48} strings of bit length 128 are mutually different with probability of at least 1 - 2^{-24}.* *(DRG.3.5) Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A as defined in [1].*
Application Note:	The defined quality metric is taken from Class DRG.3 as defined in AIS 20/31 [1]. The random numbers are generated by the underlying hardware platform.

6.1.3 FCS_CKM.2/ECDHE

The TOE shall meet the requirement "Cryptographic key distribution (ECDHE)" as specified below.

FCS_CKM.2/ECDHE	**Cryptographic key distribution (ECDHE)**

[1] [assignment: *list of standards*]
[2] [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*]
[3] [selection: *DRG.2, DRG.3, DRG.4, PTG.2, PTG.3, NTG.1*]
[4] [assignment: *a defined quality metric*]

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 that meets the following: NIST Special Publication 800-56A Revision 3 with approved groups from Appendix D.

Refinement: According to the CCC Technical Specification [14], the standard to be followed for Diffie-Hellman key agreement is BSI TR-03111 [13]. The NIST standard assigned by the Protection Profile as part of FCS_CKM.2.1/ECDHE is a mistake in the Protection Profile and is not correct. The TOE implements this functionality according to BSI TR-03111 as specified by [14].

6.1.4 FCS_CKM.4

The TOE shall meet the requirement "Cryptographic key destruction" as specified below.

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 *pseudo-random pattern*[5] that meets the following: None.

6.1.5 FCS_COP.1/Hash

The TOE shall meet the requirement "Cryptographic Operation (Hash)" as specified below.

FCS_COP.1/Hash **Cryptographic Operation (Hash)**

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

[5] [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*]

FCS_COP.1.1/Hash The TSF shall perform Cryptograhic Hashing in accordance with the specified cryptographic algorithm SHA-256 and cryptographic key sizes None that meet the following: *FIPS 180-4* [16][6].

6.1.6 FCS_COP.1/ECDSA

The TOE shall meet the requirement "Cryptographic Operation (ECDSA)" as specified below.

FCS_COP.1/ECDSA **Cryptographic Operation (ECDSA)**

Hierarchical to: No other components.

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

FCS_COP.1.1/ECDSA The TSF shall perform digital signing in accordance with a specified cryptographic algorithm ECDSA with NIST P-256 curve and cryptographic key sizes 256-bit that meet the following: ANSI X9.62 [12].

Refinement: In this Security Target the cryptographic operation defined in the Protection Profile as *digital signing* is refined to include both signature generation and signature verification.

6.1.7 FDP_ACF.1

The TOE shall meet the requirement "Security attribute based access control" as specified below.

FDP_ACF.1 **Security attribute based access control**

Hierarchical to: No other components.

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

FDP_ACF.1.1 The TSF shall enforce the SFP_AC to objects based on the following: All subjects and objects together with their respective security attributes as defined in SD_SFP.

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: Rules for all access methods and access rules defined in SD_SFP.

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: none[7].

[6] [selection: *ISO/IEC 10118-3:2018, FIPS 180-4*]
[7] [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.

Application Note: As stated in Application Note 10 of the PP [6], the dependency FMT_MSA.3 will not be fulfilled, since there is no initialisation of attributes necessary.

6.1.8 FDP_IFF.1/SCP

The TOE shall meet the requirement "Simple security attributes (SCP)" as specified below.

FDP_IFF.1/SCP **Simple security attributes (SCP)**

Hierarchical to: No other components.

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

FDP_IFF.1.1/SCP The TSF shall enforce the SCP_SFP based on the following types of subject and information security attributes: Subjects and information as defined by the SCP_SFP, and for each, the security attributes as defined in GP [15] and *no additional security attributes*[8]

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.

FDP_IFF.1.3/SCP The TSF shall enforce the *no additional rules*[9].

FDP_IFF.1.4/SCP The TSF shall explicitly authorise an information flow based on the following rules: *no additional rules*[10].

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.

6.1.9 FDP_SDI.2

The TOE shall meet the requirement "Stored data integrity monitoring and action" as specified below.

FDP_SDI.2 **Stored data integrity monitoring and action**

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring

[8] [assignment: *list of additional security attributes*]
[9] [assignment: *additional information flow control SFP rules*]
[10] [assignment: *rules, based on security attributes, that explicitly authorise information flows*]

Dependencies:	No dependencies.
FDP_SDI.2.1	The TSF shall monitor user data stored in containers controlled by the TSF for *integrity errors*[11] on all objects, based on the following attributes: *cryptographic keys*[12].
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.

6.1.10 FDP_UIT.1/CCM

The TOE shall meet the requirement "Data exchange Integrity (CCM)" as specified below.

FDP_UIT.1/CCM	**Data exchange Integrity (CCM)**
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 to *receive*[13] user data in a manner protected from *modification, deletion, insertion and replay*[14] errors.
FDP_UIT.1.2/CCM	The TSF shall be able to determine on receipt of user data, whether modification, deletion, insertion, replay has occurred.

6.1.11 FIA_UAU.3

The TOE shall meet the requirement "Unforgeable authentication" as specified below.

FIA_UAU.3	**Unforgeable authentication**
Hierarchical to:	No other components
Dependencies:	No dependencies
FIA_UAU.3.1	The TSF shall *detect and prevent*[15] use of authentication data that has been forged by any user of the TSF.

[11] [assignment: *integrity errors*]
[12] [assignment: *user data attributes*]
[13] [selection: *transmit, receive*]
[14] [selection: *modification, deletion, insertion, replay*]
[15] [selection: *detect, prevent*]

FIA_UAU.3.2	The TSF shall *detect and prevent*[16] use of authentication data that has been copied from any other user of the TSF.

6.1.12 FMT_MTD.1/deletion of keys

The TOE shall meet the requirement "Management of TSF data (deletion of keys)" as specified below.

FMT_MTD.1/deletion of keys	**Management of TSF data (deletion of keys)**
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 the keys to Digital Key framework *and no other roles*[17].

6.1.13 FMT_SMR.1

The TOE shall meet the requirement "Security roles" as specified below.

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 Keyframework, Vehicle, *and no other roles*[18].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.

6.1.14 FPT_ITT.3/DIGTAL_KEY

The TOE shall meet the requirement "TSF data integrity monitoring (Digital Key)" as specified below.

FPT_ITT.3/DIGTAL_KEY	**TSF data integrity monitoring (Digital Key)**
Hierarchical to:	No other components.
Dependencies:	No dependencies.

[16] [selection: *detect, prevent*]
[17] [assignment: *the authorised identified roles*]
[18] [assignment: the authorised identified roles]

FPT_ITT.3.1/ DIGITAL_KEY	The TSF shall be able to detect *modification of data, substitution of data, re-ordering of data, deletion of data*[19] 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: *perform a security reset or raise a security exception*[20].

6.1.15 FPT_ITT.3/IMMO_TOKEN

The TOE shall meet the requirement "TSF data integrity monitoring (Immobilizer Token)" as specified below.

FPT_ITT.3/IMMO_TOKEN	**TSF data integrity monitoring (Immobilizer Token)**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_ITT.3.1/ IMMO_TOKEN	The TSF shall be able to detect *modification of data, substitution of data, re-ordering of data, deletion of data*[21] for TSF data transmitted between separate parts of the TOE.
FPT_ITT.3.2/ IMMO_TOKEN	Upon detection of a data integrity error, the TSF shall take the following actions: *perform a security reset or raise a security exception*[22].

6.1.16 FPT_TST.1

The TOE shall meet the requirement "TSF testing" as specified below.

FPT_TST.1	**TSF testing**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_TST.1.1	The TSF shall run a suite of self tests *during initial start-up and at the request of the authorised user*[23] to demonstrate the correct operation of *the TSF*[24].
FPT_TST.1.2	The TSF shall provide authorised users with the capability to verify the integrity of *TSF data*[25].

[19] [selection: *modification of data, substitution of data, re-ordering of data, deletion of data, [assignment: other integrity errors]*]

[20] [assignment: *specify the action to be taken*]

[21] [selection: *modification of data, substitution of data, re-ordering of data, deletion of data, [assignment: other integrity errors]*]

[22] [assignment: *specify the action to be taken*]

[23] [selection: *during initial start-up, periodically during normal operation, at the request of the authorised user, at the conditions[assignment: conditions under which self test should occur]*]

[24] [selection: *[assignment: parts of TSF], the TSF*]

[25] [selection: *[assignment: parts of TSF data], TSF data*]

FPT_TST.1.3 The TSF shall provide authorised users with the capability to verify the integrity of *TSF*[26].

6.2 Security Assurance Requirements

The following table lists all security assurance components that are valid for this Security Target.

Table 8. Security Assurance Requirements

Name	Title
ADV_ARC.1	Security architecture description
ADV_FSP.4	Complete functional specification
ADV_IMP.1	Implementation representation of the TSF
ADV_TDS.3	Basic modular design
AGD_OPE.1	Operational user guidance
AGD_PRE.1	Preparative procedures
ALC_CMC.4	Production support, acceptance procedures and automation
ALC_CMS.4	Problem tracking CM coverage
ALC_DEL.1	Delivery procedures
ALC_DVS.2	Sufficiency of security measures
ALC_LCD.1	Developer defined life-cycle model
ALC_TAT.1	Well-defined development tools
ASE_INT.1	ST introduction
ASE_CCL.1	Conformance claims
ASE_SPD.1	Security problem definition
ASE_OBJ.2	Security objectives
ASE_ECD.1	Extended components definition
ASE_REQ.2	Derived security requirements
ASE_TSS.2	TOE summary specification with architectural design summary
ATE_COV.2	Analysis of coverage
ATE_DPT.1	Testing: basic design
ATE_FUN.1	Functional testing
ATE_IND.2	Independent testing - sample
AVA_VAN.5	Advanced methodical vulnerability analysis

[26] [selection: *[assignment: parts of TSF], TSF*]

6.3 Security Requirements Rationale

6.3.1 Rationale for the Security Functional Requirements

Section 5.4.1 in the Protection Profile [6] provides a rationale for the mapping between security functional requirements and security objectives defined in the Protection Profile. This rationale given there is applicable to this Security Target and is not repeated here.

6.3.2 Dependencies of Security Functional Requirements

The Protection Profile [6] defines all security functional requirements and their dependencies, with the exception of two dependencies:

- Section 5.4.2 of the Protection Profile gives a rationale for the exclusion of dependency FIA_UID.1 (required by FMT_SMR.1). The rationale states that the Protection Profile (and therefore also the Security Target) does not require the identification of the roles to be assigned, as this is handled by the operational environment.
- Application Note 10 of the PP gives a rationale for the exclusion of dependency FMT_MSA.3 (required by FDP_ACF.1). This dependency is not required since no initialisation of attributes is necessary.

6.3.3 Rationale for the Security Assurance Requirements

The selection of security assurance requirements is identical to the underlying Protection Profile [6]. Section 5.4.3 of the Protection Profile gives a rationale for the chosen evaluation assurance level of EAL4, with ALC_DVS.2 and AVA_VAN.5 selected as augmentations. The rationale given there is applicable to this Security Target.

7 TOE Summary Specification

The TOE Security Functionality (TSF) introduced in this chapter realize the SFRs of the TOE. Table 9 below lists the TSF and gives a short description on how the SFRs are implemented.

7.1 TOE Security Functionality

Table 9. TOE Security Functionality

TSF	Description
TSF.RNG	**Random Number Generator (RNG)** The applet is using the RNG provided by the hardware platform for various functionality, such as generation of cryptographic keys, generation of random nonces, and generation of X.509 certificate serial number. The random numbers are generated as described in FCS_RNG.1. The RNG is also used for clearing key buffers, implementing the key destruction functionality claimed by FCS_CKM.4.
TSF.Crypto	**Cryptographic algorithms** TSF.Crypto provides the cryptographic algorithms described in FCS_CKM.1/ECC, FCS_CKM.2/ECDHE, FCS_COP.1/Hash, FCS_COP.1/HMAC, FCS_COP.1/Encryption/decryption, FCS_COP.1/CMAC and FCS_COP.1/ECDSA. These cryptographic primitives are not implemented by the applet but by the functionality provided by the underlying JCOP operating system and cryptographic library provided by the hardware platform. The FCS_CKM.1/Session_keys, FCS_CKM.1/Long_Term_key, and FCS_CKM.1/Secret_Shared_key generate 256-bit cryptographic keys using HKDF-SHA-256. This KDF is not provided by the platform but is implemented by the applet, using cryptographic primitives provided by the JCOP operating system and hardware platform. The applet also makes sure that cryptographic buffers and keys are destroyed in a secure manner, implementing the functionality claimed by FCS_CKM.4 and FDP_RIP.1.
TSF.Access-Ctrl	**Access Control** Section 5.2.3 of the Digital Key Protection Profile [6] defines an access control policy for the loading, installation, and removal of the applet based on GlobalPlatform [15] specifications. For this functionality the TOE depends on the JCOP operating system which implements these specifications through the Card Content Management (CCM). This covers the access control requirements defined in FDP_ACC.2 and FDP_ACF.1. For communication with the CCM a secure channel protocol needs to be established as defined by the SFRs FDP_IFC.2/SCP and FDP_IFF.1/SCP. For this functionality the applet also depends on the implementation provided by the JCOP operating system. Access to the applet itself is initiated through the use of its AID. The value used for the AID is defined by the Digital Key Specification [14], which the applet requires to be installed with through its user documentation. This meets the requirement stated by FMT_MTD.3
TSF.Auth	**Authentication** The applet is only aware of two entities it communicates with, being the Digital Key framework inside the phone and the vehicle. These two entities are both defined a role by FMT_SMR.1. The applet assures that forging of authentication data is prevented and detected as required by FIA_UAU.3. The applet only allows the Digital Key framework to delete a key and makes sure that a deletion attestation is created for the deleted key and that this key is deleted before the attestation is transferred. This covers the requirement FMT_SMF.1
TSF.Integrity	**Integrity Protection and Disclosure** The TOE protects its stored cryptographic keys against integrity errors by storing these keys as standard JavaCard key objects, using the underlying JCOP operating system. The JCOP operating system is then responsible for monitoring the integrity of the cryptographic keys. This implements the requirement defined by FDP_SDI.2. The integrity of the Digital Key and the immobiliser token is protected against modification when transmitted between the device and the vehicle by using a CMAC message authentication code. This

Table 9. TOE Security Functionality*...continued*

TSF	Description
	requirement is met by FPT_ITT.3/IMMO_TOKEN and FPT_ITT.3/DIGITAL_KEY. The encrypted secure channel that is used for this communication also ensures confidentiality of the immobiliser token and Digital Key. This is met by the requirement FPT_ITT.1/IMMO_TOKEN and FPT_ITT.1/DIGITAL_KEY. This functionality also meets FTP_ITC.1 and FPT_ITI.1.1/ Vehicle_Integrity. As communication between the TOE and a vehicle is protected, it is not possible for other entities to determine whether the communication was initiated by the same or another user. This meets the requirement FPR_UNL.1. The secure channel used to communicate with the CCM (see TSF.Access-Ctrl) also protects the integrity of the communication by detecting modification, deletion, insertion and replay errors. As stated earlier, the secure channel is provided by the underlying JCOP platform. This functionality implements the claims in FDP_UIT.1/CCM.
TSF.Protection	**Physical and Logical Protection** The applet implements secure coding techniques to protect against attacks, but largely depends on the platform for its protection against physical and logical attacks, such as fault injection and side-channel analysis. The hardware is protected by various techniques, such as active shielding, glue logic routing of signals, sensors such as voltage and light sensors, and scrambling of memories. The JCOP operating system is also securely implemented, and makes use of attack counters to respond to threats. This implements the requirement claimed by FPT_PHP.3. The applet and JCOP operating system also ensure that transaction data shared between the applet and the vehicle, and the applet and the Digital Key framework can not be replayed. Once replay is detected, the transaction is terminated. This meets the requirement FPT_RPL.1 The hardware platform also ensures that the TOE remains in a secure state after power-loss. This is required by FPT_RCV.2 Protection is also provided by performing self-tests during the start-up of the device and at the request of the JCOP operating system, implementing the requirement FPT_TST.1.

8 References

8.1 Evaluation documents

[1] A proposal for: Functionality classes for random number generators, Wolfgang Killmann, T-Systems GEI GmbH, Werner Schindler, Bundesamt für Sicherheit in der Informationstechnik (BSI), Version 2.0, 18 September 2011.

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

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

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

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

[6] Car Connectivity Consortium Digital Key, Protection Profile of the Digital Key Applet, Version 1.0, CCC-CP-023, 16 October 2023.

[7] Java Card System - Open Configuration Protection Profile, Certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-CC-PP-0099-V2-2020, Version 3.1, April 2020.

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

8.2 Developer documents

[9] UM11616 - CCC Digital Key Applet, User Manual, NXP Semiconductors, Docstore ID 675516, Revision 1.6, 7 December 2023.

[10] NXP JCOP 6.2 on SN220 Secure Element, Security Target, Product evaluation document, NXP Semiconductors, Revision 1.8, 24 November 2022.

[11] SN220 Series - Secure Element with Crypto Library, Security Target, Evaluation document, NXP Semiconductors, Revision 1.5, 29 September 2022.

8.3 Standards

[12] ANSI X9.62: Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA), American National Standard for Financial Services, 16 November 2005.

[13] Technical Guideline BSI TR-03111: Elliptic Curve Cryptography, Federal Office for Information Security (BSI), Version 2.10, 1 June 2018.

[14] Car Connectivity Consortium, Digital Key Release 3, Technical Specification, CCC-TS-101, Car Connectivity Consortium LLC, Version 1.1.1, 17 July 2023.

[15] GlobalPlatform. GlobalPlatform Card Specification 2.3.1, GPC_SPE_034, GlobalPlatform Inc., Mar 2018.

[16] FIPS PUB 180-4: Secure Hash Standard (SHS), Federal Information Processing Standards Publication, US Department of Commerce/National Institute of Standards and Technology, August 2015.

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

Legal information

Definitions

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

Disclaimers

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Trademarks

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

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

Tables

Figures

Contents

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

For more information, please visit: https://www.nxp.com