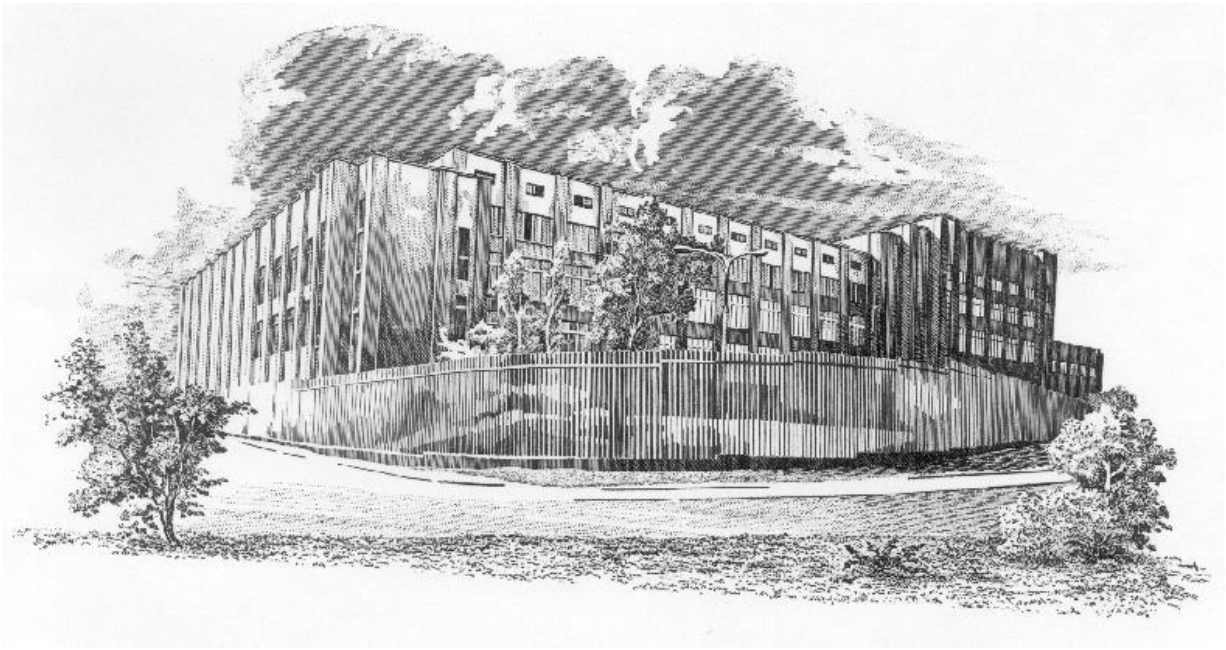
PWPW
POLSKA WYTWÓRNIA
PAPIERÓW WARTOŚCIOWYCH



# PWPW SmartApp-ID 5.0 (SIGN configuration)

## Security Target Lite

This page is intentionally left blank.

# Table of contents

# List of tables

# List of figures

# 1 Introduction

## 1.1 References

### 1.1.1 Security Target reference

| | |
|---|---|
| ST title: | PWPW SmartApp-ID 5.0 (SIGN configuration): Security Target Lite |
| ST author: | Polska Wytwórnia Papierów Wartościowych S.A. |
| ST version: | 5.0.1.0 |
| ST date: | 2024-11-05 |
| Evaluation body: | TÜV Informationstechnik GmbH (TÜVIT) |
| Certification body: | TrustCB B.V. |
| Evaluation assurance level: | EAL4 augmented with the following assurance components ALC_DVS.2, ATE_DPT.2 and AVA_VAN.5 |

### 1.1.2 Target of evaluation reference

| | |
|---|---|
| TOE identification: | PWPW SmartApp-ID 5.0 (SIGN configuration) |
| TOE developer: | Polska Wytwórnia Papierów Wartościowych S.A. |
| TOE certification ID: | NSCIB-2300120 |
| TOE OS: | JCOP 4.5 P71 |
| TOE OS certification ID: | NSCIB-CC-2300127-01 |
| TOE HW: | NXP Secure Smart Card Controller N7122 with IC Dedicated Software and Crypto Library (R1/R2/R3) |
| TOE HW certification ID: | BSI-DSZ-CC-1149-V3-2023 |

## 1.2 Intended usage

In SIGN configuration the TOE provides functionalities of a secure signature creation device (SSCD). It is intended for the usage in cryptographic cards and other products providing the SSCD functionality.

## 1.3 Target of evaluation

### 1.3.1 Overview

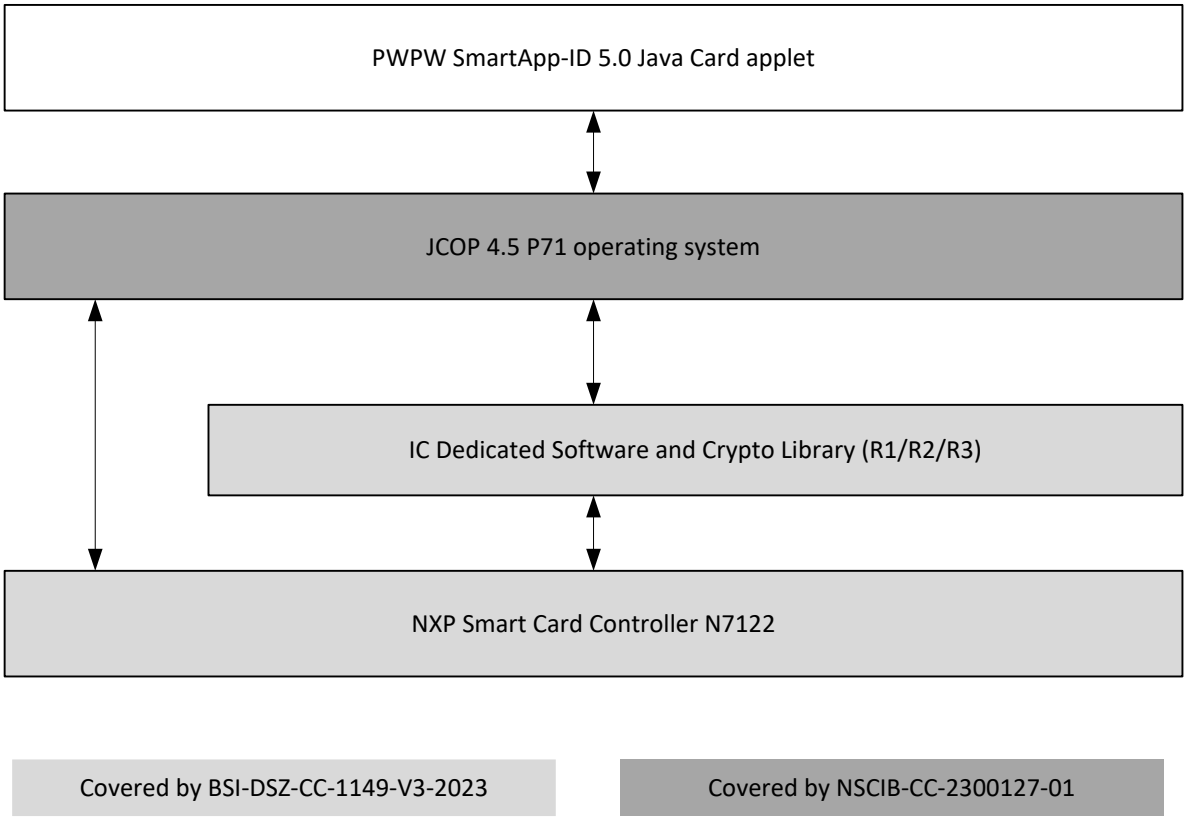
This security target defines the security objectives and requirements for the Secure Signature Creation Device (SSCD).

The TOE is a composite product. It comprises of a Java Card applet executed on top of the Common Criteria certified hardware and software.

The TOE consists of a Java Card (PWPW SmartApp-ID 5.0) applet executed by operating system (JCOP 4.5 P71) on microcontroller (NXP Smart Card Controller N7122) with IC Dedicated Software and Crypto Library (R1/R2/R3).

Section 1.3.3 describes components of the TOE.

**Figure 1.1: TOE components overview**




The TOE is a combination of hardware and software configured to securely create, use and manage signature creation data (SCD). Moreover, the TOE allows to securely store identification data of its holder.

The TOE protects the SCD during its whole life cycle as to be used in a signature creation process solely by its signatory. The TOE comprises all IT security functionality necessary to ensure the secrecy of the SCD and the security of the electronic signature.

The TOE protects personal data of its holder (including low sensitive biometric data) during its whole life cycle and contains authentication data used to verify terminals accessing these personal data.

The TOE provides the following functionalities:

1. to generate signature creation data (SCD) and the correspondent signature verification data (SVD),
2. to export the SVD for certification,
3. to import signature creation data (SCD),
4. to receive and store certificate info,
5. to switch the TOE from a non-operational state to an operational state,
6. if in an operational state, to create digital signatures for data with the following steps:
    a. to select an SCD if multiple are present in the SSCD,
    b. to authenticate the signatory and determine its intent to sign,
    c. to receive data to be signed or a unique representation thereof (DTBS/R),

    d. to apply an appropriate cryptographic signature creation function using the selected SCD to the DTBS/R;
7. to present personal data of its holder to authorized terminals,
8. to prove the identity as SSCD to external entities,
9. to protect confidentiality and integrity of data sent to/from the TOE.

PWPW SmartApp-ID 5.0 (SIGN configuration) supports also following security protocols:

1. Password Authenticated Connection Establishment
    a. Generic Mapping,
    b. Chip Authentication Mapping (PACE-CAM),
2. Chip Authentication.

The TOE is prepared for the signatory's (holder) use by:

1. generating or importing at least one SCD/SVD pair,
2. personalizing for the signatory by storing in the TOE:
    a. the signatory’s reference authentication data (RAD);
    b. optionally, certificate info for at least one SCD,
    c. certificate(s) for generated or imported SCD(s);
3. optionally, storing personal data of its holder.

The TOE supports ADFs creation and usage. MRTD functionality has its dedicated and only one ADF. SIGN is represented as well by ADF. It is possible to create only one ADF with SSCD functionality. Each ADF contains separated DF/EF structure. By selecting specified ADF, the applet’s security manager grants access to specific for MRTD or SSCD cryptographic functionalities. It is no possible to grant access from ADF.MRTD level to ADF.SSCD and vice versa.

***Developer note***
*MRTD functionality is out of scope of this document, it is described in a separate documentation [ASE MRTD].*

### 1.3.2 TOE definition

The TOE addressed by this security target is a secure signature creation device representing an integrated circuit (IC) programmed according to [EN 419211-2], [EN 419211-3], [EN 419211-4], [EN 419211-5], [EN 419211-6]. The communication between terminal and chip shall be protected by Password Authenticated Connection Establishment (PACE) according to [PP_PACE].

The TOE can be delivered in one of the following forms:

1. ICs (modules) ready for embedding;
2. ICs embedded in identity documents, plastic cards or other medium.

The TOE can provide one or both of the following interfaces:

1. contact,
2. contactless.

The TOE shall contain only one instance of the SmartApp-ID (the Java Card applet containing evaluated application).

Additional applets and their instances are not allowed. They cannot be loaded in post-issuance.

### 1.3.3 TOE components

PWPW SmartApp-ID 5.0 (SIGN configuration) is the TOE. It comprises of:

1. the micro controller with IC Dedicated Software,
2. the operating system,
3. the Java Card applet PWPW SmartApp-ID 5.0 containing SIGN application,
4. the guidance documentation.

#### 1.3.3.1 Microcontroller

The microcontroller has been certified according to the Common Criteria for Evaluation Assurance Level 6+. The exact reference to microcontroller certification is given in Table 1.1.

**Table 1.1: Microcontroller certification details**

| Developer | Name | Certification ID | EAL | Reference |
|---|---|---|---|---|
| NXP | NXP Smart Card Controller N7122 with IC Dedicated Software and Crypto Library (R1/R2/R3) | BSI-DSZ-CC-1149-V3-2023 | EAL 6+ | [ST_HW] |

The certification of the microcontroller includes IC Dedicated Software and embedded Crypto Library (R1/R2/R3).

#### 1.3.3.2 Operating system

The operating system has been certified according to the Common Criteria for Evaluation Assurance Level 6+. The exact reference to the operating system is given in Table 1.2.

**Table 1.2: Operating system certification details**

| Developer | Name | Certification ID | EAL | Reference |
|---|---|---|---|---|
| NXP | JCOP 4.5 P71 | NSCIB-CC-2300127-01 | EAL 6+ | [ST_OS] |

#### 1.3.3.3 Java Card applet

Components of the Java Card applet are identified in Table 1.3.

**Table 1.3: Components of the Java Card applet**

| Type | Developer | Name |
|---|---|---|
| Java Card applet | PWPW | pl.pwpw.smartapp.id.SFAccess |
| Java Card package | PWPW | pl.pwpw.smartapp.id.common |
| Java Card package | PWPW | pl.pwpw.smartapp.id.mrtd[1] |
| Java Card package | PWPW | pl.pwpw.smartapp.id.sign |

[1] Machine Readable Travel Document (MRTD) functionality is out of scope of this document, it is described in a separate documentation.

#### 1.3.3.4 Guidance documentation

The guidance documentation components are identified in Table 1.4.

**Table 1.4: Guidance documentation components**

| Type | Developer | Name |
|---|---|---|
| Document | PWPW | PWPW SmartApp-ID 5.0 (SIGN configuration): Preparative procedures |
| Document | PWPW | PWPW SmartApp-ID 5.0 (SIGN configuration): Operational user guidance |

***Developer note:***
*The exact versions of guidance documents are given in the certification report.*

### 1.3.4 TOE usage and security features for operational use

The operation environment of the SIGN consists of three distinct environments:

1. The secure preparation environment, where the SSCD interacts with a certification service provider (CSP) through:
    a. a certificate generation application (CGA) to obtain a certificate for the signature verification data (SVD) corresponding with the SCD the SSCD has generated; the preparation environment interacts further with the SSCD to personalize it with the initial value of the reference authentication data (RAD); and/or

***Developer note:***
1. *The SSCD and the CGA communicate through a trusted channel in order to protect the integrity and authenticity of the SVD exported from the SSCD and obtained certificate.*
2. *Description from point a. concerns the usage scenario for generating SCD by the TOE.*

    b. a SCD/SVD generation application to import the signature creation data (SCD) and a certificate generation application (CGA) to obtain a certificate for the signature validation data (SVD) corresponding to the SCD the certification service provider has generated; the SCD/SVD generation application transmits the SVD to the CGA; the preparation environment interacts further with the SSCD to personalize it with the initial value of the reference authentication data (RAD).

***Developer note:***
1. *The SSCD and SCD/SVD generation application communicate through a trusted channel in order to protect the integrity and authenticity of the imported SCD, the corresponding SVD and obtained certificate.*
2. *Description from point b. concerns the usage scenario for importing keys to the TOE.*

2. The signing environment where the SSCD interacts with a signer through a signature creation application (SCA) to sign data after authenticating the signer as its signatory. The signature creation application provides the data to be signed (DTBS), or a unique representation thereof (DTBS/R) as input to the SSCD signature creation function and obtains the resulting digital signature[2]).

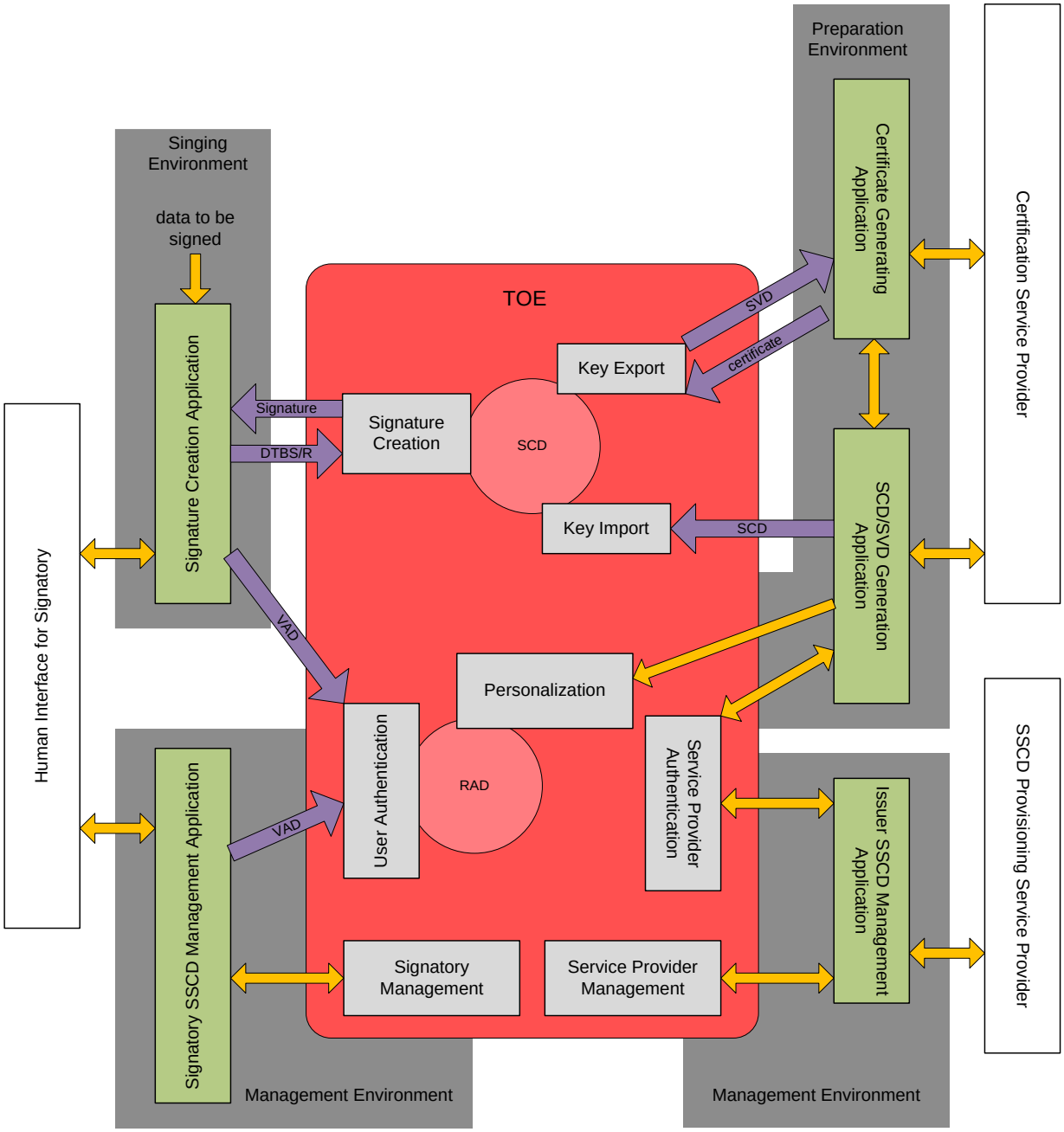
[2] At a pure functional level the SSCD creates a digital signature; for an implementation of the SSCD, in that meeting the requirements of the claimed PP and with the key certificate created as specified in the directive, Annex I, the result of the signing process can be used as to create a qualified electronic signature.

***Developer note:***
*The SSCD and the SCA communicate through a trusted channel in order to protect the integrity of the DTBS/R and resulting digital signature.*

3. The management environments where the SSCD interacts with the user or an SSCD provisioning service provider to perform management operations, e.g. for the signatory to reset a blocked RAD. A single device, e.g. a smart card terminal, may provide the required secure environment for management and signing.

Figure 1.2 illustrates the operational environment of the SSCD application.

**Figure 1.2: Operation environment of the SSCD application**




The SSCD stores signature creation data (SCD) and reference authentication data (RAD). The SSCD may store multiple instances of SCD. In this case, the SSCD provides a function to identify

each SCD and the SCA can provide an interface to the signer to select an SCD for use in the signature creation function of the SSCD. The SSCD protects the confidentiality and integrity of the SCD and restricts its use in signature creation to its signatory. The digital signature created by the SSCD may be used to create an advanced electronic signature as defined in Article 5.1 of [Directive] and Article 26 of [EIDAS]. Determining the state of the certificate as qualified is beyond the scope of the claimed protection profiles.

The signature creation application (SCA) is assumed to protect the integrity of the input it provides to the SSCD signature creation function as being consistent with the user data authorized for signing by the signatory. Unless implicitly known to the SSCD, the SCA indicates the kind of the signing input (as DTBS/R) it provides and computes any hash value required. The SSCD may augment the DTBS/R with signature parameters it stores and then computes a hash value over the input as needed by the kind of input and the used cryptographic algorithm.

The SSCD stores signatory reference authentication data (RAD) to authenticate a user as its signatory. The RAD is a password (e.g. PIN). The SSCD protects the confidentiality and integrity of the RAD. The SSCD receives the VAD from the signature creation application (SCA). It is assumed that SCA protects the confidentiality and integrity of the VAD.

A certification service provider interacts with the SSCD in the secure preparation environment to perform any preparation function of the SSCD required before control of the SSCD is given to the legitimate user. These functions may include:

1. initializing the RAD,
2. generating a key pair,
3. storing personal information of the legitimate user and/or certificate info.

The TOE provides Chip Authentication Protocol mechanism to prove the identity as SSCD to external entities.

A typical example of an SSCD device is a smart card. In this case, a smart card terminal may be deployed that provides the required secure environment to handle a request for signatory authorization. A signature can be obtained on a document prepared by a signature creation application component running on a personal computer connected to the card terminal. The signature creation application, after presenting the document to the user and after obtaining the authorization PIN, initiates the digital signature creation function of the smart card through the terminal.

#### 1.3.4.1 Authentication mechanisms

Authentication mechanisms are differentiated by the user roles:

1. *Signatory* (the end user),
2. *Administrator* (MRTD *Personalization Agent* role equivalent).

*Signatory* can authenticate himself using PIN.

During preparation phase SCP03 protocol is used for *Administrator* authentication. In Operational phase PACE-PIN and PACE-PUK are reserved for *Administrator* authentication.

***Developer note:***
*Administrator authentication with PACE-PIN or PACE-PUK involves establishing the PACEv2 protocol.*

The information on used protocols and cryptographic algorithms is given in Annex A.

Implementation of all cryptographic algorithms is provided by the platform.

#### 1.3.4.2 Cryptographic functions

Cryptographic functions are used to:

- generate cryptographic keys,
- create digital signatures,
- decrypt messages,
- perform key agreement,
- destruct cryptographic keys,
- generate random numbers.

The information on supported cryptographic algorithms is given in Annex A.

Implementation of all cryptographic algorithms is provided by the platform.

Random numbers are generated with the generator provided by the platform and compliant to the PTG.2 class specified in [AIS20/AIS31].

#### 1.3.4.3 Protection against interference, logical tampering and bypass

The platform protects the SmartApp-ID against malfunctions that are caused by exposure to operating conditions. This includes hardware resets and operation outside the intended environment characterized, among others, in terms of temperature and Vcc.

The platform provides protection against physical attack and performs self-tests as described in [ST_OS].

The SmartApp-ID applet uses secure values, redundant storage mechanisms and checksums to protect sensitive data. For objects available in Java Card API platform mechanisms and data types are used (e.g. for cryptographic keys storage). For simple data types like *byte*, *short* or byte arrays mechanisms are implemented at the applet level (see [ADV_ARC]).

The SmartApp-ID uses duplicated condition checks and counters to protect the execution flow.

The SmartApp-ID uses the dedicated counter to limit the number of potential attacks and to block itself, when the security is threatened.

#### 1.3.4.4 Access control, storage and protection of data

Security attribute based access control is used. Access control is enforced by dedicated, internal functions of the SSCD application. These functions check if access requests are compliant to rules defined in the functional specification.

All cryptographic keys are stored in dedicated structures, which are provided and protected by the platform. These structures are Java Card objects derived from the type Key.

#### 1.3.4.5 Trusted channel

Establishing a trusted channel is required for interactions (communication) between the TOE and external applications.

Trusted channel protects confidentiality and integrity of the communication by means of encryption and message authentication codes respectively.

Protection of confidentiality is based on:

- AES in CBC mode with 128, 192 or 256 bit session key or,
- Triple-DES in CBC mode with 112 bit session key.

Protection of integrity is based on:

- AES in CMAC mode with 128, 192 or 256 bit session key or,
- Triple-DES in Retail mode with 112 bit session key.

The information on the used protocol and cryptographic algorithms is given in Annex A.

Implementation of AES and Triple-DES primitives, CBC mode of operation for AES and Triple-DES, Retail mode of operation for Triple-DES and CMAC mode of operation for AES are provided by the platform.

#### 1.3.4.6 Security and life cycle management

The PWPW SmartApp-ID 5.0 (SIGN configuration) applet life cycle consists of the following states:

1. Testing,
2. Configuration,
3. Pre-personalization,
4. Personalization,
5. Operational use – Activated,
6. Operational use – Deactivated,
7. Terminated.

Testing state covers syntax tests of the TOE.

***Developer note:***
*In the Testing state, all operations are available. Verification of the applet current state, authentication flags and file access conditions are disabled. This mode of operation shall only be used during TOE development. Testing life cycle is permanently disabled after TOE delivery.*

Initial life cycle state of the applet is Configuration. Moving to next state is possible using SWITCH LIFECYCLE command. It is not possible to move back to the previous state (e.g. from Operational use to Personalization).

Transition to Terminated state is irreversible and renders the TOE unusable.

***Developer note:***
*The PWPW SmartApp-ID 5.0 (SIGN configuration) applet does not enable any mechanism allowing to revert its life cycle phase.*

Preparation, including the TOE personalization, is performed using the commands available in the preparation phase.

During operational use phase communication between the TOE and external applications is restricted to the use of a trusted channel in case of sensitive data exchange.

The test features of the platform are protected by ways described in the platform documentation.

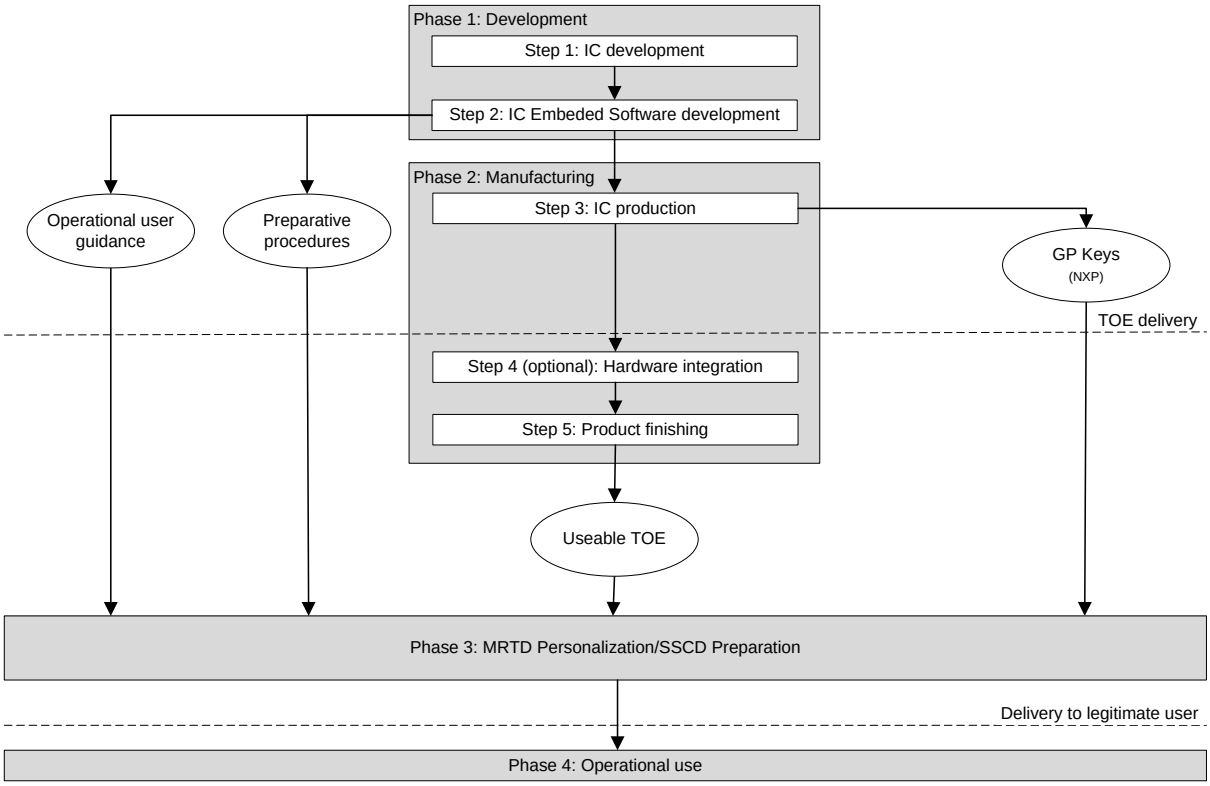
The platform protects the TOE against malfunctions that are caused by exposure to operating conditions.

The cryptographic keys stored on TOE are protected from disclosure.

### 1.3.5 Life cycle

The TOE life-cycle is described in terms of the four life-cycle phases. (With respect to the [PP_IC], the TOE life-cycle is additionally subdivided into 7 steps.). The TOE life cycle is presented in Figure 1.3.

**Figure 1.3: TOE life-cycle**




#### 1.3.5.1 Phase 1: Development

(Step1) The TOE is developed in Phase 1. The IC developer develops the integrated circuit, the IC Dedicated Software and the guidance documentation associated with these TOE components.

***Developer note:***
*NXP is the IC developer.*
*The IC Dedicated Software is developed by NXP.*

(Step2) The software developer uses the guidance documentation for the integrated circuit and the guidance documentation for relevant parts of the IC Dedicated Software and develops the IC Embedded Software (operating system). Then another developer (PWPW) uses the operating system, its documentation and develops the Java Card applet and the guidance documentation associated with these TOE components.

***Developer note:***
*The operating system is developed by NXP.*
*Operating system guidance documentation is developed by NXP.*
*Java Card applet is developed by PWPW.*
*Java Card applet guidance documentation is developed by PWPW.*

The manufacturing documentation of the IC including the IC Dedicated Software, the Embedded Software and the Java Card applet are securely delivered to the IC Manufacturer.

#### 1.3.5.2 Phase 2: Manufacturing

(Step3) In a first step, the TOE integrated circuit is produced containing the SSCD's chip Dedicated Software, Embedded Software and the Java Card applet in the non-volatile memory (ROM). The IC Manufacturer writes the IC Identification Data onto the chip to control the IC as SSCD during the IC manufacturing and the delivery process to the SSCD manufacturer or IC Packaging Manufacturer (depending on the optional step 4)..

If necessary, the IC manufacturer adds the parts of the IC Embedded Software in the non-volatile programmable memories (i.e. EEPROM).

***Developer note:***
*NXP is the IC Manufacturer.*

(Step4 optional) The IC Packaging Manufacturer combines the IC with hardware for the contact-based/contactless interface in the SSCD unless the SSCD consists of the card only.

IC with integrated peripherals is delivered to the SSCD manufacturer.

(Step5) The SSCD manufacturer (i) adds the IC Embedded Software or part of it in the non-volatile programmable memories (i.e. EEPROM or FLASH) if necessary, (ii) creates the SIGN application instance, and (iii) equips SSCD's chips with pre-personalization Data.

The SSCD together with the IC Identifier is securely delivered from the SSCD manufacturer to the SSCD provisioning service provider. The SSCD manufacturer also provides the relevant parts of the guidance documentation to the SSCD provisioning service provider.

#### 1.3.5.3 Phase 3: Preparation

***Developer note:***
*All preparation actions related to SSCD functionality could only be performed in the ADF.SSCD.*

(Step6) The preparation phase of the TOE lifecycle is processing the TOE from the customer's acceptance of the delivered TOE to a state ready for operation by the signatory. The customer receiving the TOE from the manufacturer is the SSCD-provisioning service that prepares and provides the SSCD to subscribers.

An SSCD provisioning service provider having accepted the TOE from the IC Manufacturer prepares the TOE for use and delivers it to its legitimate user. The preparation phase ends when the legitimate user has received the TOE from the SSCD provisioning service provider and any SCD it might already hold have been enabled for use in signing.

During preparation of the TOE, as specified above, an SSCD provisioning service provider performs the following tasks:

1. Obtain information on the intended recipient of the device as required for the preparation process and for identification as a legitimate user of the applet.

2. Establish the trusted channel with the applet.
3. Generate a PIN and store this data as RAD in the applet and prepare information about the VAD for delivery to the legitimate user.
4. Optionally, store the certificate info inside the applet which may be mandatory for certificate usage in operational environment. These data are out of the evaluation scope.
5. Generate a certificate for at least one SCD according to one of the following scenarios:
    a. Scenario 1 – generate key pair inside the TOE:
        - invoke security function responsible for generating an SCD/SVD pair inside the TOE and exporting the SVD.
        - optionally, read the certificate info from the TOE;
        - in the preparation environment, create a certificate request containing the exported SVD;
        - obtain a certificate for the SVD exported from the applet;
        - store the obtained certificate inside the TOE.
    b. Scenario 2 – import key pair to the TOE:
        - in the preparation environment, generate an SCD/SVD pair;
        - invoke security function (of the SSCD applet) responsible for importing an SCD/SVD;
        - optionally, read the certificate info from the SSCD applet;
        - in the preparation environment, create a certificate request containing the generated SVD;
        - obtain a certificate for the generated SVD;
        - store the obtained certificate inside the SSCD applet.
6. Deliver the VAD info to the legitimate user.
7. Once all required certificates are stored inside the SSCD applet, its preparation is finished.

The SVD certification task of an SSCD provisioning service provider as specified in the claimed PP may support a centralized, pre-issuing key generation process, with at least one key generated and certified, before delivery to the legitimate user. Alternatively or additionally, that task may support key generation by the signatory after delivery and outside the secure preparation environment. An applet may support both key generation processes, for example with a first key generated centrally and additional keys generated by the signatory in the operational use stage.

***Developer note:***
*The TOE supports both one-time and multiple key generation for security environment. Configuration shall be set during TOE preparation. After leaving preparation phase it is not possible to change security environment configuration.*

Once the TOE preparation is completed, the SSCD provisioning service provider delivers it to the legitimate user. It ends the preparation phase.

#### 1.3.5.4 Phase 4: Operational use

(Step7) In this lifecycle stage the signatory can use the TOE to create advanced electronic signatures.

The operational use of the SSCD applet begins when the signatory has obtained both the VAD and the TOE. Enabling the applet for signing requires at least one set of SCD stored in its memory and setting the RAD (i.e. the PIN) known only to the legitimate user.

The signatory can also interact with the SSCD application to perform management tasks, e.g. reset a RAD value or use counter if the password/PIN in the reference data has been lost or blocked. Such management tasks require a secure environment.

The signatory can render an SCD in the SSCD application permanently unusable.

The applet supports functions to generate additional signing keys and to securely obtain certificates for the new keys. For an additional key the signatory is allowed to choose the kind of certificate (qualified, or not) to obtain for the SVD of the new key. The signatory is also allowed to choose some of the data in the certificate request for instance to use a pseudonym instead of the legal name in the certificate[3]. If the conditions to obtain a qualified certificate are met the new key can also be used to create qualified electronic signatures.

The TOE life cycle as SSCD ends when all set of SCD stored in the SSCD application are destructed. This may include deletion of the corresponding certificates or rendering all SCD in the SSCD permanently unusable.

[3] The certificate request in this case will contain the name of the signatory as the requester, as for instance it may be signed by the signatory's existing SCD.

# 2 Conformance claims

## 2.1 Common Criteria conformance claims

This security target claims conformance to:

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model; CCMB-2017-04-001; Version 3.1, Revision 5, April 2017 [CC-Part1]
- Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components; CCMB-2017-04-002; Version 3.1, Revision 5, April 2017 [CC-Part2]
- Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance requirements; CCMB-2017-04-003; Version 3.1, Revision 5, April 2017 [CC-Part3]

as follows:

- Part 2 extended
- Part 3 conformant

*Common Methodology for Information Technology Security Evaluation, Evaluation methodology; CCMB-2017-04-004; Version 3.1, Revision 5, April 2017* [CC-CEM] has to be taken into account.

## 2.2 Protection profile claims

This security target claims strict conformance to the following Common Criteria protection profiles:

- EN 419211-2:2013: Protection profiles for secure signature creation device - Part 2: Device with key generation; BSI-CC-PP-0059-2009-MA-02 [EN 419211-2],
- EN 419211-3:2013: Protection profiles for secure signature creation device - Part 3: Device with key import; BSI-CC-PP-0075-2012-MA-01 [EN 419211-3],
- EN 419211-4:2013: Protection profiles for secure signature creation device - Part 4: Extension for device with key generation and trusted channel to certificate generation application; BSI-CC-PP-0071-2012-MA-01 [EN 419211-4],
- EN 419211-5:2013: Protection profiles for secure signature creation device - Part 5: Extension for device with key generation and trusted channel to signature creation application; BSI-CC-PP-0072-2012-MA-01 [EN 419211-5],
- EN 419211-6:2014: Protection profiles for secure signature creation device - Part 6: Extension for device with key import and trusted channel to signature creation application; BSI-CC-PP-0076-2013-MA-01 [EN 419211-6].

The TOE also covers additional functionalities, i.e.: secure messaging based on PACE and CA. To keep this document consistent, appropriate definitions were included from:

- Common Criteria Protection Profile Machine Readable Travel Document using Standard Inspection Procedure with PACE (PACE PP), BSI-CC-PP-0068-V2-2011-MA-01, Version 1.01, 22nd July 2014 ([PP_PACE]),

- Common Criteria Protection Profile Machine Readable Travel Document with „ICAO Application", Extended Access Control with PACE (EAC PP), BSI-CC-PP-0056-V2-2012, version 1.3.2, 5th December 2012 ([PP_EAC]).

## 2.3 Package claim

This security target is conformant to the assurance package EAL 4 augmented with the following assurance components ALC_DVS.2, ATE_DPT.2 and AVA_VAN.5.

## 2.4 Conformance claims rationale

This security target uses definitions of assets, objects, users, subjects, external entities, threat agents, threats, organizational security policies and assumptions given in the claimed protection profiles (see section 3 for details).

This security target uses security objectives given in the claimed protection profiles (see section 4 for details). No security objective is modified.

***Developer note:***
*The OT.SCD/SVD_Auth_Gen of the [EN 419211-2] is correspondent to the OT.SCD/SVD_Gen in [EN 419211-4] and [EN 419211-5].*

This Security Target does not contain OE.SSCD_Prov_Service, OE.HID_VAD and OE.DTBS_Protect.

OE.SSCD_Prov_Service specified in [EN 419211-2] and [EN 419211-3] has been replaced with OE.Dev_Prov_Service as required by [EN 419211-4].

OE.HID_VAD specified in [EN 419211-2] and [EN 419211-3] has been split into OE.HID_TC_VAD_Exp and OT.TOE_TC_VAD_Imp as required by [EN 419211-5] and [EN 419211-6].

OE.DTBS_Protect specified in [EN 419211-2] has been split into OE.SCA_TC_DTBS_Exp and OT.TOE_TC_DTBS_Imp as required by [EN 419211-5] and [EN 419211-6].

This security target uses only extended components given in the claimed protection profiles (see section 5 for details). No extended component is modified.

This security target uses SFRs given in the claimed protection profiles (see section 6 for details). Only operations of the SFRs (assignment, iteration, selection and refinement) explicitly permitted by the claimed protection profiles are done.

Following SFRs from [PP_PACE] and [PP_EAC] are used in this security target in terms of the PACE and CA implementation: FCS_CKM.1/DH_PACE, FCS_CKM.1/CA, FCS_COP.1/PACE_ENC, FCS_COP.1/PACE_MAC, FCS_COP.1/CA_ENC, FCS_COP.1/CA_MAC, FCS_RND.1, FIA_UID.1/PACE, FIA_UAU.1/PACE, FIA_UAU.4/PACE, FIA_UAU.5/PACE, FIA_UAU.6/PACE, FIA_UAU.6/EAC, FMT_SMR.1/PACE, FMT_MTD.1/KEY_READ, FMT_MTD.1/CAPK, FMT_LIM.1, FMT_LIM.2 and FTP_ITC.1/PACE.

One additional SFR is introduced, i.e. FCS_CKM.1.1/CAPK. It is done with the adherence to [PP_EAC].

**FCS_CKM.1/CAPK Cryptographic key generation – Chip Authentication key pair**

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation]: fulfilled by FCS_COP.1/CA_ENC and FCS_COP.1/CA_MAC

FCS_CKM.4 Cryptographic key destruction: fulfilled by FCS_CKM.4.

*FCS_CKM.1.1/CAPK*

*The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm Generating ECDH / ECDSA keys with Brainpool curve or NIST curve (for length 521 bits) and specified cryptographic key sizes of 224, 256, 320, 384, 512 and 521 bits that meet the following: [ISO15946-1], [ISO15946-3], [TR02102-1] and [TR03110-3].*

All **application notes** given in the claimed protection profiles are considered and addressed. Moreover, all **application notes** requiring security target writer actions are commented with **developer notes**.

# 3 Security problem definition

This security target claims strict conformance to [EN 419211-2], [EN 419211-3], [EN 419211-4], [EN 419211-5] and [EN 419211-6]. All definitions of assets, object, threat agents, threats, organizational security policies and assumptions given in these protection profiles are included to the security target. The definitions are taken over as described in the protection profiles, therefore they are not repeated here.

## 3.1 Assets and objects

All assets and objects defined in the claimed protection profiles are used in this security target.

None of the assets and objects taken from these protection profiles have been modified.

Table 3.1 presents included definitions of primary assets from claimed protection profiles.

**Table 3.1: Assets from claimed protection profiles**

| Assets | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| SCD | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| SVD | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| DTBS and DTBS/R | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |

Additionally to the assets defined from the claimed protection profiles, assets described in Table 3.2 has been introduced in this security target to cover secure messaging functionality implemented by the TOE using PACE and Chip Authentication protocols.

**Table 3.2: Additional assets for Secure Messaging**

| Assets | Involved protection profiles | Usage scenario |
|---|---|---|
| Accessibility to the TOE functions and data only for authorized subjects | [PP_PACE] | key generation, key import |
| Genuineness of the TOE | [PP_PACE] | key generation, key import |
| TOE internal secret cryptographic keys | [PP_PACE] | key generation, key import |
| TOE internal non-secret cryptographic material | [PP_PACE] | key generation, key import |
| Travel document communication establishment authorization data | [PP_PACE] | key generation, key import |
| Authenticity of the travel document's chip | [PP_EAC] | key generation, key import |

***Developer note:***
*Additional assets listed in Table 3.2 are used in this security target only in terms of the PACE and CA protocols as the TOE is the SSCD with support of MRTD-typical trusted channel implementation.*

## 3.2 Users, subjects and external entities

All users, subjects and external entities defined in the claimed protection profiles are used in this security target.

Table 3.3 presents included definitions of subjects from claimed protection profiles.

**Table 3.3: Users, subjects and external entities from claimed protection profiles**

| Definitions | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| User | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| Administrator | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| Signatory | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |

Additionally to the definitions of subjects from the claimed protection profiles, subjects described in Table 3.4 has been introduced in this security target to cover secure messaging functionality implemented by the TOE using PACE and Chip Authentication protocols.

**Table 3.4: Additional users, subjects and external entities for Secure Messaging**

| Definitions | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| Travel document holder | [PP_PACE] | User role equivalent | key generation, key import |
| Terminal | [PP_PACE] | | key generation, key import |
| Basic Inspection System with PACE (BIS-PACE) | [PP_PACE] | | key generation, key import |
| Personalization Agent | [PP_PACE] | Administrator role equivalent | key generation, key import |
| Inspection System | [PP_EAC] | | key generation, key import |

***Developer note:***
*Additional definitions listed in Table 3.4 are used in this security target only in terms of the PACE and CA protocols as the TOE is the SSCD with support of MRTD-typical trusted channel implementation.*

## 3.3 Threat agents

Only threat agents defined in the claimed protection profiles are used in this security target. No additional threat agent is introduced.

Table 3.5 presents included definitions of threat agents.

**Table 3.5: Threat agents from claimed protection profiles**

| Threat agents | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| Attacker | [EN 419211-2], [EN 419211-3], [PP_PACE] | identical in both SSCD protection profiles<br>definition from [PP_PACE] does not contradict SSCD definition | key generation, key import |

## 3.4 Threats

All threats defined in the claimed protection profiles are used in this security target. None of the threats taken from these protection profiles has been modified.

Table 3.6 presents included definitions of threats from claimed protection profiles.

**Table 3.6: Threats from claimed protection profiles**

| Threats | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| T.SCD_Divulg | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| T.SCD_Derive | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| T.Hack_Phys | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| T.SVD_Forgery | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| T.SigF_Misuse | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| T.DTBS_Forgery | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| T.Sig_Forgery | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |

Additionally to the threats from the claimed protection profiles, threats described in Table 3.7 has been introduced in this security target to cover secure messaging functionality implemented by the TOE using PACE and Chip Authentication protocols.

**Table 3.7: Additional threats for Secure Messaging**

| Threats | Involved protection profiles | Usage scenario |
|---|---|---|
| T.Skimming | [PP_PACE] | key generation, key import |
| T.Eavesdropping | [PP_PACE] | key generation, key import |
| T.Abuse-Func | [PP_PACE] | key generation, key import |
| T.Information_Leakage | [PP_PACE] | key generation, key import |
| T.Phys-Tamper | [PP_PACE] | key generation, key import |
| T.Malfunction | [PP_PACE] | key generation, key import |
| T.Counterfeit | [PP_EAC] | key generation, key import |

***Developer note:***
*Additional threats listed in Table 3.7 are used in this security target only in terms of the PACE and CA protocols as the TOE is the SSCD with support of MRTD-typical trusted channel implementation.*

## 3.5 Organizational security policies

Only organizational security policies defined in the claimed protection profiles are used in this security target.

Table 3.8 presents included definitions of organizational security policies from claimed protection profiles.

**Table 3.8: Organizational security policies from claimed protection profiles**

| Organizational Security Policies | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| P.CSP_Qcer | [EN 419211-2], [EN 419211-3] | information about SVD origin added in [EN 419211-2]<br>with regards to content identical in both protection profiles | key generation, key import |
| P.Qsign | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| P.Sigy_SSCD | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| P.Sig_Non-Repud | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |

Additionally to the organizational security policies from the claimed protection profiles, organizational security policies described in Table 3.9 has been introduced in this security target to cover secure messaging functionality implemented by the TOE using PACE and Chip Authentication protocols.

**Table 3.9: Additional organizational security policies for Secure Messaging**

| Organizational Security Policies | Involved protection profiles | Usage scenario |
|---|---|---|
| P.Pre-Operational | [PP_PACE] | key generation, key import |
| P.Terminal | [PP_PACE] | key generation, key import |

***Developer note:***
*Additional organizational security policies listed in Table 3.9 are used in this security target only in terms of the PACE and CA protocols as the TOE is the SSCD with support of MRTD-typical trusted channel implementation.*

## 3.6 Assumptions

All assumptions defined in the claimed protection profiles are used in this security target.

None of the assumptions taken from these protection profiles has been modified.

No additional assumption has been introduced.

Table 3.10 presents included definitions of assumptions.

**Table 3.10: Assumptions from claimed protection profiles**

| Assumptions | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| A.CGA | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| A.SCA | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| A.CSP | [EN 419211-3] | | key import |

# 4 Security objectives

This security target claims strict conformance to [EN 419211-2], [EN 419211-3], [EN 419211-4], [EN 419211-5] and [EN 419211-6]. All definitions of security objectives given in these protection profiles are included to the security target. The definitions are taken over as described in the protection profiles, therefore they are not repeated here.

## 4.1 Security objectives for the target of evaluation

All security objectives for the target of evaluation defined in the claimed protection profiles are used in this security target.

None of the security objectives for the target of evaluation taken from these protection profiles has been modified.

The following definitions of security objectives for the target of evaluation are included:

- *OT.Lifecycle_Security,*

***Application note 1 from [EN 419211-2]:***
***Application note 1 from [EN 419211-3]:***
*The TOE may contain more than one set of SCD. There is no need to destroy the SCD in case of repeated SCD generation. The signatory shall be able to destroy the SCD stored in the SSCD, e.g. after the (qualified) certificate for the corresponding SVD has been expired.*

***Developer note:***
*The TOE allows generating/importing up to 31 (thirty-one) SCD/SVD pairs. The signatory can destroy any of them at any time.*

- *OT.SCD_Secrecy,*

***Application note 2 from [EN 419211-2]:***
***Application note 2 from [EN 419211-3]:***
*The TOE shall keep the confidentiality of the SCD at all times, in particular during SCD/SVD generation, signature creation operation, storage and secure destruction.*

***Developer note:***
*The TOE is a Java Card applet and uses dedicated Java Card mechanisms to store and manipulate the SCD/SVD pairs. These mechanisms ensure confidentiality and integrity of SCD/SVD pairs. The TOE does not allow to export SCDs.*

- *OT.Sig_Secure,*
- *OT.Sigy_SigF,*
- *OT.DTBS_Integrity_TOE,*
- *OT.EMSEC_Design,*
- *OT.Tamper_ID,*
- *OT.Tamper_Resistance*
- *OT.TOE_TC_VAD_Imp,*

***Application note 3 from [EN 419211-5]:***
***Application note 1 from [EN 419211-6]:***
*This security objective for the TOE is partly covering OE.HID_VAD. While OE.HID_VAD requires only the operational environment to protect VAD, OT.TOE_TC_VAD_Imp requires the HID and the TOE to implement a trusted channel for the protection of the VAD: the HID exports the VAD and establishes*

*one end of the trusted channel according to OE.HID_TC_VAD_Exp, the TOE imports VAD at the other end of the trusted channel according to OT.TOE_TC_VAD_Imp. Therefore this security objective for the TOE reassigns partly the VAD protection from the operational environment as described by OE.HID_VAD to the TOE as described by OT.TOE_TC_VAD_Imp and leaves only the necessary functionality by the HID.*

- *OT.TOE_TC_DTBS_Imp*,

***Application note 2 from [EN 419211-5]:***
***Application note 2 from [EN 419211-6]:***
*This security objective for the TOE is partly covering OE.DTBS_Protect. While OE.DTBS_Protect requires only the operational environment to protect DTBS, OT.TOE_TC_DTBS_Imp requires the SCA and the TOE to implement a trusted channel for the protection of the DTBS: the SCA exports the DTBS and establishes one end of the trusted channel according to OE.SCA_TC_DTBS_Exp, the TOE imports DTBS at the other end of the trusted channel according to OT.TOE_TC_DTBS_Imp. Therefore this security objective for the TOE reassigns partly the DTBS protection from the operational environment as described by OE.DTBS_Protect to the TOE as described by OT.TOE_TC_DTBS_Imp and leaves only the necessary functionality by the SCA.*

- *OT.SCD/SVD_Auth_Gen*,
- *OT.SCD_Unique*,
- *OT.SCD_SVD_Corresp*,
- *OT.TOE_SSCD_Auth*,
- *OT.TOE_TC_SVD_Exp*,
- *OT.SCD_Auth_Imp*.

Table 4.1 presents the summary for the TOE security objectives.

**Table 4.1: Security objectives from claimed protection profiles summary**

| Security objectives for the TOE | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| OT.Lifecycle_Security | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| OT.SCD_Secrecy | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| OT.Sig_Secure | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| OT.Sigy_SigF | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| OT.DTBS_Integrity_TOE | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| OT.EMSEC_Design | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| OT.Tamper_ID | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| OT.Tamper_Resistance | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |

**Table 4.1 (continued)**

| Security objectives for the TOE | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| OT.TOE_TC_VAD_Imp | [EN 419211-5], [EN 419211-6] | corresponds to OE.HID_TC_VAD_Exp from [EN 419211-5] and [EN 419211-6] | key generation, key import |
| OT.TOE_TC_DTBS_Imp | [EN 419211-5], [EN 419211-6] | corresponds to OE.SCA_TC_DTBS_Exp from [EN 419211-5] and [EN 419211-6] | key generation, key import |
| OT.SCD/SVD_Auth_gen | [EN 419211-2] | corresponds to OE.SCD/SVD_Auth_gen from [EN 419211-3] | key generation |
| OT.SCD_Unique | [EN 419211-2] | corresponds to OE.SCD_Unique from [EN 419211-3] | key generation |
| OT.SCD_SVD_Corresp | [EN 419211-2] | corresponds to OE.SCD_SVD_Corresp from [EN 419211-3] | key generation |
| OT.TOE_SSCD_Auth | [EN 419211-4] | | key generation |
| OT.TOE_TC_SVD_Exp | [EN 419211-4] | | key generation |
| OT.SCD_Auth_Imp | [EN 419211-3] | | key import |

Additionally to the TOE security objectives from the claimed protection profiles, objectives described in Table 4.2 has been introduced in this security target to cover secure messaging functionality implemented by the TOE using PACE and Chip Authentication protocols.

**Table 4.2: Additional TOE security objectives for Secure Messaging**

| Security objectives for the TOE | Involved protection profiles | Usage scenario |
|---|---|---|
| OT.Data_Integrity | [PP_PACE] | key generation, key import |
| OT.Data_Authenticity | [PP_PACE] | key generation, key import |
| OT.Data_Confidentiality | [PP_PACE] | key generation, key import |
| OT.Prot_Abuse-Func | [PP_PACE] | key generation, key import |
| OT.Prot_Inf_Leak | [PP_PACE] | key generation, key import |
| OT.Prot_Phys-Tamper | [PP_PACE] | key generation, key import |
| OT.Prot_Malfunction | [PP_PACE] | key generation, key import |
| OT.AC_Pers | [PP_PACE] | key generation, key import |
| OT.Chip_Auth_Proof | [PP_EAC] | key generation, key import |

***Developer note:***
*Additional TOE security objectives listed in Table 4.2 are used in this security target only in terms of the PACE and CA protocols as the TOE is the SSCD with support of MRTD-typical trusted channel implementation.*

## 4.2 Security objectives for operational environment

All security objectives for the operational environment defined in the claimed protection profiles are used in this security target.

None of the security objectives for the operational environment taken from these protection profiles has been modified.

*OE.SVD_Auth* definitions from [EN 419211-2] and [EN 419211-3] have been combined to make it relevant for both key import and key generation scenarios.

*OE.Dev_Prov_Service* from [EN 419211-4] substitutes *OE.SSCD_Prov_Service* from core protection profile ([EN 419211-2]).

*OE.HID_TC_VAD_Exp* and *OE.SCA_TC_DTBS_Exp* from [EN 419211-5] and [EN 419211 6] substitute *OE.HID_VAD* and *OE.DTBS_Protect* from core protection profile ([EN 419211-3]), respectively.

The following definitions of security objectives for the operational environment are included:

- *OE.SVD_Auth*,

The operational environment shall ensure the integrity (in case SCD generation by the TOE) and authenticity (in case of SCD import to the TOE) of the SVD sent to the CGA of the CSP. The CGA verifies the correspondence between the SCD in the SSCD of the signatory and the SVD in the qualified certificate.

- *OE.CGA_QCert*,
- *OE.DTBS_Intend*,

***Application note 3 from [EN 419211-2]:***
***Application note 3 from [EN 419211-3]:***
*The SCA should be able to support advanced electronic signatures. Currently, there are three formats defined by ETSI recognized as meeting the requirements needed by advanced electronic signatures: CAdES, XAdES and PAdES. These three formats mandate to include the hash of the signer's public key certificate in the data to be signed. In order to support for the mobility of the signer, it is recommended to store the certificate info on the SSCD for use by SCA and identification of the corresponding SCD if more than one SCD is stored on the SSCD.*

***Developer note:***
*The certificate info can be stored in the TOE. It is available to the SCA and can be included in the data to be signed if necessary. The SCA calculates the hash from the data to be signed and then sends the hash to the TOE. The TOE calculates the signature value for the received hash and returns it to the SCA. It is the responsibility of the SCA to format the signature value according to the requirements of appropriate standards.*

- *OE.Signatory,*
- *OE.HID_TC_VAD_Exp,*

***Application note 3 from [EN 419211-5]:***
*This security objective for the TOE is partly covering OE.HID_VAD. While OE.HID_VAD requires only the operational environment to protect VAD, OE.HID_TC_VAD_Exp requires the HID and the TOE to implement a trusted channel for the protection of the VAD: the HID exports the VAD and establishes one end of the trusted channel according to OE.HID_TC_VAD_Exp, the TOE imports VAD at the other end of the trusted channel according to OT.TOE_TC_VAD_Imp. Therefore this security objective for the TOE reassigns partly the VAD protection from the operational environment as described by OE.HID_VAD to the TOE as described by OT.TOE_TC_VAD_Imp and leaves only the necessary functionality by the HID.*

- *OE.SCA_TC_DTBS_Exp*,

***Application note 4 from [EN 419211-5]:***
***Application note 2 from [EN 419211-6]:***
*This security objective for the TOE is partly covering OE.DTBS_Protect. While OE.DTBS_Protect requires only the operational environment to protect DTBS, OE.SCA_TC_DTBS_Exp requires the SCA and the TOE to implement a trusted channel for the protection of the DTBS: the SCA exports the DTBS and establishes one end of the trusted channel according to OE.SCA_TC_DTBS_Exp, the TOE imports DTBS at the other end of the trusted channel according to OT.TOE_TC_DTBS_Imp. Therefore this security objective for the TOE reassigns partly the DTBS protection from the operational environment as described by OE.DTBS_Protect to the TOE as described by OT.TOE_TC_DTBS_Imp and leaves only the necessary functionality by the SCA.*

- *OE.Dev_Prov_Service*,

***Developer note:***
*OE.Dev_Prov_Service defined in [EN 419211-4] substitutes OE.SSCD_Prov_Service from [EN 419211-2].*

- *OE.CGA_SSCD_Auth,*
- *OE.CGA_TC_SVD_Imp,*
- *OE.SCD/SVD_Auth_Gen,*
- *OE.SCD_Secrecy,*
- *OE.SCD_Unique,*
- *OE.SCD_SVD_Corresp.*

Table 4.3 presents the security objectives for the operational environment summary.

**Table 4.3: Security objectives for the operational environment from claimed protection profiles summary**

| Security objectives for the operational environment | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| OE.SVD_Auth | [EN 419211-2], [EN 419211-3] | combined to make it relevant for both key import and key generation scenario | key generation, key import |
| OE.CGA_QCert | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| OE.DTBS_Intend | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| OE.Signatory | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| OE.HID_TC_VAD_Exp | [EN 419211-5], [EN 419211-6] | substitutes OE.HID_VAD from [EN 419211-2] and [EN 419211-3]<br>corresponds to OT.TOE_TC_VAD_Imp from [EN 419211-5] and [EN 419211-6] | key generation, key import |
| OE.SCA_TC_DTBS_Exp | [EN 419211-5], [EN 419211-6] | substitutes OE.DTBS_Protect from [EN 419211-2] and [EN 419211-3]<br>corresponds to OT.TOE_TC_DTBS_Imp from [EN 419211-5] and [EN 419211-6] | key generation, key import |
| OE.Dev_Prov_Service | [EN 419211-4] | substitutes OE.SSCD_Prov_Service from [EN 419211-2] | key generation |
| OE.CGA_SSCD_Auth | [EN 419211-4] | | key generation |
| OE.CGA_TC_SVD_Imp | [EN 419211-4] | | key generation |
| OE.SCD/SVD_Auth_gen | [EN 419211-3] | corresponds to OT.SCD/SVD_Auth_gen from [EN 419211-2] | key import |
| OE.SCD_Secrecy | [EN 419211-3] | | key import |
| OE.SCD_Unique | [EN 419211-3] | corresponds to OT.SCD_Unique from [EN 419211-2] | key import |
| OE.SCD_SVD_Corresp | [EN 419211-3] | corresponds to OT.SCD_SVD_Corresp from [EN 419211-2] | key import |

Additionally to the security objectives for the operational environment from the claimed protection profiles, objectives described in Table 4.4 has been introduced in this security target to cover secure messaging functionality implemented by the TOE using PACE and Chip Authentication protocols.

**Table 4.4: Additional security objectives for the operational environment for Secure Messaging**

| Security objectives for the TOE | Involved protection profiles | Usage scenario |
|---|---|---|
| OE.Personalisation | [PP_PACE] | key generation, key import |
| OE.Terminal | [PP_PACE] | key generation, key import |
| OE.Travel_Document_Holder | [PP_PACE] | key generation, key import |
| OE.Auth_Key_Travel_Document | [PP_EAC] | key generation, key import |

***Developer note:***
*Additional security objectives for the operational environment listed in Table 4.4 are used in this security target only in terms of the PACE and CA protocols as the TOE is the SSCD with support of MRTD-typical trusted channel implementation.*

## 4.3 Security objective rationale

All SSCD threats, organizational security policies and assumptions described in this security target are coming from [EN 419211-2], [EN 419211-3], [EN 419211-4], [EN 419211-5] and [EN 419211-6]. Therefore SSCD security objectives rationales given in the protection profiles remain in force.

As the TOE implements secure messaging functionality using PACE and Chip Authentication protocols, additional security objectives from [PP_PACE] and [PP_EAC] were introduced to cover PACE and CA protocols implementation. Therefore security objectives rationales given in [PP_PACE] and [PP_EAC] remain in force in terms of the trusted channel implementation.

Mapping between security problem definition and security objectives of the TOE for key import is given in Table 4.5 and for key generation in Table 4.6.

**Table 4.5: Mapping of security problem definition to security objectives of the TOE – key import**

| Security problem definition \ Security objective | OT.Lifecycle_Security | OT.SCD_Auth_Imp | OT.SCD_Secrecy | OT.Sig_Secure | OT.Sigy_SigF | OT.DTBS_Integrity_TOE | OT.EMSEC_Design | OT.Tamper_ID | OT.Tamper_Resistance | OT.TOE_TC_VAD_Imp | OT.TOE_TC_DTBS_Imp | OT.Chip_Auth_Proof | OT.AC_Pers | OT.Data_Integrity | OT.Data_Authenticity | OT.Data_Confidentiality | OT.Prot_Abuse-Func | OT.Prot_Inf_Leak | OT.Prot_Phys-Tamper | OT.Prot_Malfunction | OE.SCD/SVD_Auth_Gen | OE.SCD_Secrecy | OE.SCD_Unique | OE.SCD_SVD_Corresp | OE.CGA_Qcert | OE.SVD_Auth | OE.Dev_Prov_Service | OE.DTBS_Intend | OE.Signatory | OE.HID_TC_VAD_Exp | OE.SCA_TC_DTBS_Exp | OE.Auth_Key_Travel_Document | OE.Personalisation | OE.Terminal | OE.Travel_Document_Holder |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| T.SCD_Divulg | | X | X | | | | | | | | | | | | | | | | | | X | X | | | | | | | | | | | | | |
| T.SCD_Derive | | | | X | | | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | |
| T.Hack_Phys | | | X | | | | X | X | X | | | | | | | | | | | | | | | | | | | | | | | | | | |
| T.SVD_Forgery | | | | | | | | | | | | | | | | | | | | | | | | X | | X | | | | | | | | | |
| T.SigF_Misuse | X | | | | X | X | | | | X | X | | | | | | | | | | | | | | | | | X | X | X | X | | | | |
| T.DTBS_Forgery | | | | | | X | | | | | X | | | | | | | | | | | | | | | | | X | | | X | | | | |
| T.Sig_Forgery | | | | X | | | | | | | | | | | | | | | | | | | X | | X | | | | | | | | | | |
| T.Counterfeit | | | | | | | | | | | | X | | | | | | | | | | | | | | | | | | | | X | | | |
| T.Skimming | | | | | | | | | | | | | | X | X | X | | | | | | | | | | | | | | | | | | | X |
| T.Eavesdroping | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | | | | | | |
| T.Abuse-Func | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | | | | | |
| T.Information_Leakage | | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | | | | |
| T.Phys-Tamper | | | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | | | |
| T.Malfunction | | | | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | | |
| P.CSP_QCert | X | X | | | | | | | | | | | | | | | | | | | X | | | X | X | | | | | | | | | | |
| P.QSign | | | | X | X | | | | | | | | | | | | | | | | | | | | X | | | X | | | | | | | |
| P.Sigy_SSCD | X | X | X | X | X | X | X | | X | | | | | | | | | | | | X | X | X | | | | X | | | | | | | | |
| P.Sig_Non-Repud | X | | X | X | X | X | X | X | X | X | X | | | | | | | | | | X | X | X | X | X | X | X | X | X | X | X | | | | |
| P.Pre-Operational | | | | | | | | | | | | | X | | | | | | | | | | | | | | | | | | | | X | | |
| P.Terminal | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | X | |
| A.CGA | | | | | | | | | | | | | | | | | | | | | | | | | X | X | | | | | | | | | |
| A.SCA | | | | | | | | | | | | | | | | | | | | | | | | | | | | X | | | | | | | |
| A.CSP | | | | | | | | | | | | | | | | | | | | | X | X | X | X | | | | | | | | | | | |

**Table 4.6: Mapping of security problem definition to security objectives of the TOE – key generation**

| Security problem definition / Security objective | OT.Lifecycle_Security | OT.SCD/SVD_Auth_Gen | OT.SCD_Unique | OT.SCD_SVD_Corresp | OT.SCD_Secrecy | OT.Sig_Secure | OT.Sigy_SigF | OT.DTBS_Integrity_TOE | OT.EMSEC_Design | OT.Tamper_ID | OT.Tamper_Resistance | OT.TOE_TC_VAD_Imp | OT.TOE_TC_DTBS_Imp | OT.TOE_SSCD_Auth | OT.TOE_TC_SVD_Exp | OT.Chip_Auth_Proof | OT.AC_Pers | OT.Data_Integrity | OT.Data_Authenticity | OT.Data_Confidentiality | OT.Prot_Abuse-Func | OT.Prot_Inf_Leak | OT.Prot_Phys-Tamper | OT.Prot_Malfunction | OE.CGA_Qcert | OE.SVD_Auth | OE.Dev_Prov_Service | OE.DTBS_Intend | OE.Signatory | OE.CGA_SSCD_Auth | OE.CGA_TC_SVD_Imp | OE.HID_TC_VAD_Exp | OE.SCA_TC_DTBS_Exp | OE.Auth_Key_Travel_Document | OE.Personalisation | OE.Terminal | OE.Auth_Key_Travel_Document |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| T.SCD_Divulg | | | | | X | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| T.SCD_Derive | | X | | | | X | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| T.Hack_Phys | | | | | X | | | | X | X | X | | | | | | | | | | | | | | | | | | | | | | | | | | |
| T.SVD_Forgery | | | | X | | | | | | | | | | | X | | | | | | | | | | X | | | | | | X | | | | | | |
| T.SigF_Misuse | X | | | | | | X | X | | | | X | X | | | | | | | | | | | | | | | X | X | | | X | X | | | | |
| T.DTBS_Forgery | | | | | | | | X | | | | | | | | | | | | | | | | | | | | X | | | | | X | | | | |
| T.Sig_Forgery | | | X | | | X | | | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | |
| T.Counterfeit | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | | | | | X | | | |
| T.Skimming | | | | | | | | | | | | | | | | | | X | X | X | | | | | | | | | | | | | | | | | X |
| T.Eavesdroping | | | | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | | | | |
| T.Abuse-Func | | | | | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | | | |
| T.Information_Leakage | | | | | | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | | |
| T.Phys-Tamper | | | | | | | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | |
| T.Malfunction | | | | | | | | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | |
| P.CSP_QCert | X | | | X | | | | | | | | | | X | | | | | | | | | | | X | | | | | X | | | | | | | |
| P.QSign | | | | | | X | X | | | | | | | | | | | | | | | | | | X | | | X | | | | | | | | | |
| P.Sigy_SSCD | X | X | X | | X | X | X | X | X | | X | | | X | X | | | | | | | | | | | | X | | | X | X | | | | | | |
| P.Sig_Non-Repud | X | | X | X | X | X | X | X | X | X | X | X | X | X | X | | | | | | | | | | X | X | X | X | X | X | X | X | X | | | | |
| P.Pre-Operational | | | | | | | | | | | | | | | | | X | | | | | | | | | | | | | | | | | | X | | |
| P.Terminal | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | X | |
| A.CGA | | | | | | | | | | | | | | | | | | | | | | | | | X | X | | | | | | | | | | | |
| A.SCA | | | | | | | | | | | | | | | | | | | | | | | | | | | | X | | | | | | | | | |

# 5 Extended components definition

This security target claims strict conformance to [EN 419211-2], [EN 419211-3], [EN 419211-4], [EN 419211-5] and [EN 419211-6]. All definitions of extended components given in these protection profiles are included to the security target. As the TOE implements secure messaging functionality using PACE and Chip Authentication protocols, additional extended components from [PP_PACE] and [PP_EAC] were introduced to cover PACE and CA protocols implementation. The definitions are taken over as described in the protection profiles, therefore they are not repeated here.

The following definitions of extended components are included:

- FPT_EMS.1 from [EN 419211-2],
- FIA_API from [EN 419211-4],
- FCS_RND.1 from [PP_PACE],
- FMT_LIM from [PP_PACE].

# 6 Security requirements

This section defines the functional requirements for the TOE and the assurance requirements for the TOE.

## 6.1 Security functional requirements

The permitted operations (assignment, iteration, selection and refinement) of the SFR, that have been made by the PP author are denoted as underlined text.

The refinements of the SFR that have been made by the PP author are denoted as **bolded text**.

The permitted operations (assignment, iteration, selection and refinement) of the SFR, that have been filled in by the ST author are denoted as *underlined and italic text*.

Table 6.1 presents the summary of the functional requirements for the TOE.

**Table 6.1: Security functional requirements summary**

| Security requirements | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| FCS_CKM.1/RSA | [EN 419211-2] | refines FCS_CKM.1 from [EN 419211-2] | key generation |
| FCS_CKM.1/ECDSA | [EN 419211-2] | refines FCS_CKM.1 from [EN 419211-2] | key generation |
| FCS_CKM.1/DH_PACE | [PP_PACE] | | key generation, key import |
| FCS_CKM.1/CA | [PP_EAC] | | key generation, key import |
| FCS_CKM.1/CAPK | | | key generation, key import |
| FCS_CKM.4 | [EN 419211-2], [EN 419211-3], [PP_PACE] | differences in application notes | key generation, key import |
| FCS_COP.1/RSA | [EN 419211-2], [EN 419211-3] | refines FCS_COP.1 from [EN 419211-2] and [EN 419211-3]<br>differences in application notes | key generation, key import |
| FCS_COP.1/ECDSA | [EN 419211-2], [EN 419211-3] | refines FCS_COP.1 from [EN 419211-2] and [EN 419211-3]<br>differences in application notes | key generation, key import |
| FCS_COP.1/PACE_ENC | [PP_PACE] | | key generation, key import |
| FCS_COP.1/PACE_MAC | [PP_PACE] | | key generation, key import |
| FCS_COP.1/CA_ENC | [PP_EAC] | | key generation, key import |
| FCS_COP.1/CA_MAC | [PP_EAC] | | key generation, key import |

**Table 6.1 (continued)**

| Security requirements | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| FCS_RND.1 | [PP_PACE] | | key generation, key import |
| FDP_ACC.1/Signature_Creation | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FDP_ACC.1/SCD/SVD_Generation | [EN 419211-2] | | key generation |
| FDP_ACC.1/SVD_Transfer | [EN 419211-2] | | key generation |
| FDP_ACC.1/SCD_Import | [EN 419211-3] | | key import |
| FDP_ACF.1/Signature_Creation | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FDP_ACF.1/SCD/SVD_Generation | [EN 419211-2] | | key generation |
| FDP_ACF.1/SVD_Transfer | [EN 419211-2] | | key generation |
| FDP_ACF.1/SCD_Import | [EN 419211-3] | | key import |
| FDP_UIT.1/DTBS | [EN 419211-5], [EN 419211-6] | identical in both protection profiles | key generation, key import |
| FDP_RIP.1 | [EN 419211-2], [EN 419211-3], [PP_PACE] | identical in both SSCD protection profiles<br>definition extended to cover objects from [PP_PACE] | key generation, key import |
| FDP_SDI.2/Persistent | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FDP_SDI.2/DTBS | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FDP_DAU.2/SVD | [EN 419211-4] | | key generation |
| FDP_ITC.1/SCD | [EN 419211-3] | | key import |
| FDP_UCT.1/SCD | [EN 419211-3] | | key import |
| FIA_UID.1 | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FIA_UID.1/PACE | [PP_PACE], [PP_EAC] | TA related statements were excluded from SFR definition as TA protocol is out of this certification scope | key generation, key import |
| FIA_UAU.1 | [EN 419211-4], [EN 419211-5], [EN 419211-6] | substitutes FIA_UAU.1 of [EN 419211-2] and [EN 419211-3]<br>combined to make it relevant for both key import and key generation scenario | key generation, key import |
| FIA_UAU.1/PACE | [PP_PACE], [PP_EAC] | TA related statements were excluded from SFR definition as TA protocol is out of this certification scope | key generation, key import |

**Table 6.1 (continued)**

| Security requirements | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| FIA_UAU.4/PACE | [PP_PACE], [PP_EAC] | TA related statements were excluded from SFR definition as TA protocol is out of this certification scope | key generation, key import |
| FIA_UAU.5/PACE | [PP_PACE], [PP_EAC] | TA related statements were excluded from SFR definition as TA protocol is out of this certification scope | key generation, key import |
| FIA_UAU.6/PACE | [PP_PACE] | | key generation, key import |
| FIA_UAU.6/EAC | [PP_EAC] | | key generation, key import |
| FIA_AFL.1 | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FIA_API.1 | [EN 419211-4], [PP_EAC] | | key generation |
| FMT_SMR.1 | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FMT_SMR.1/PACE | [PP_PACE], [PP_EAC] | TA related statements were excluded from SFR definition as TA protocol is out of this certification scope | key generation, key import |
| FMT_SMF.1 | [EN 419211-2], [EN 419211-3], [PP_PACE] | combined to make it relevant for both key import and key generation scenario<br>definition extended to cover objects from [PP_PACE] | key generation, key import |
| FMT_MOF.1 | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FMT_MSA.1/Admin | [EN 419211-2], [EN 419211-3] | combined to make it relevant for both key import and key generation scenario | key generation, key import |
| FMT_MSA.1/Signatory | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FMT_MSA.2 | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FMT_MSA.3 | [EN 419211-2], [EN 419211-3] | combined to make it relevant for both key import and key generation scenario | key generation, key import |

**Table 6.1 (continued)**

| Security requirements | Involved protection profiles | Comment | Usage scenario |
|---|---|---|---|
| FMT_MSA.4 | [EN 419211-2], [EN 419211-3] | combined to make it relevant for both key import and key generation scenario | key generation, key import |
| FMT_MTD.1/Admin | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FMT_MTD.1/Signatory | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FMT_MTD.1/KEY_READ | [PP_PACE], [PP_EAC] | | key generation, key import |
| FMT_MTD.1/CAPK | [PP_EAC] | | key generation, key import |
| FMT_LIM.1 | [PP_PACE], [PP_EAC] | | key generation, key import |
| FMT_LIM.2 | [PP_PACE], [PP_EAC] | | key generation, key import |
| FPT_EMS.1 | [EN 419211-2], [EN 419211-3], [PP_PACE], [PP_EAC] | identical in both SSCD protection profiles<br>definition extended to cover objects from [PP_PACE] and [PP_EAC] | key generation, key import |
| FPT_FLS.1 | [EN 419211-2], [EN 419211-3], [PP_PACE] | identical in both SSCD protection profiles<br>definition extended to cover objects from [PP_PACE] | key generation, key import |
| FPT_PHP.1 | [EN 419211-2], [EN 419211-3] | identical in both protection profiles | key generation, key import |
| FPT_PHP.3 | [EN 419211-2], [EN 419211-3], [PP_PACE] | identical in all protection profiles | key generation, key import |
| FPT_TST.1 | [EN 419211-2], [EN 419211-3], [PP_PACE] | identical in all protection profiles | key generation, key import |
| FTP_ITC.1/VAD | [EN 419211-5], [EN 419211-6] | identical in both protection profiles | key generation, key import |
| FTP_ITC.1/DTBS | [EN 419211-5], [EN 419211-6] | identical in both protection profiles | key generation, key import |
| FTP_ITC.1/SVD | [EN 419211-4] | | key generation |
| FTP_ITC.1/SCD | [EN 419211-3] | | key import |
| FTP_ITC.1/PACE | [PP_PACE] | | key generation, key import |

### 6.1.1 Class FCS: Cryptographic support

***Application note 4 from [EN 419211-2]:***
*Member states of the European Union have specified entities as responsible for accreditation and supervision of the evaluation process for products conforming to this standard and for determining*

*admissible algorithms and algorithm parameters ([Directive]: 1.1b and 3.4). The ST writer shall consult with these entities to learn of admissible algorithms and cryptographic key sizes and other parameters or applicable standards.*

***Developer note:***
*The above application note has been considered by the ST writer.*

#### 6.1.1.1 FCS_CKM.1: Cryptographic key generation

**FCS_CKM.1/RSA [key generation]**

*FCS_CKM.1.1/RSA*

The TSF shall generate an **SCD/SVD pair** in accordance with a specified cryptographic key generation algorithm *key generation algorithm for RSA (with CRT)* and specified cryptographic key sizes *RSA (with CRT): 1024, 2048 and 4096 bits* that meet the following: *[TR02102-1]*.

***Developer note:***
1. *RSA-1024 keys are supported for the backward compatibility, nevertheless these keys **are not** recommended according to the [TR02102-1].*
2. *For a period of use beyond 2023 the present Technical Report Guidance [TR02102-1] recommends using a key length of 3000 bits in order to achieve a compatible level of security for all asymmetric mechanisms.*

**FCS_CKM.1/ECDSA [key generation]**

*FCS_CKM.1.1/ECDSA*

The TSF shall generate an **SCD/SVD pair** in accordance with a specified cryptographic key generation algorithm *key generation for EC* and specified cryptographic key sizes *224, 256, 320, 384, 512, 521 bits* that meet the following: *[TR02102-1]*.

***Application note 5 from [EN 419211-2]:***
*The ST writer shall perform the missing operations in the element FCS_CKM.1.1. The refinement in the element FCS_CKM.1.1 substitutes "cryptographic keys" by "SCD/SVD pairs" because it clearly addresses the SCD/SVD key generation.*

***Developer note:***
1. *The ST writer has performed the missing assignments.*
2. *Elliptic curves with 224 bit sizes are supported for the backward compatibility, nevertheless these curves **are not** recommended according to the [TR02102-1].*

**FCS_CKM.1/DH_PACE [key generation, key import]**

*FCS_CKM.1.1/DH_PACE*

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *ECDH compliant to [TR03111]* and specified cryptographic key sizes *of 112, 128, 192, 256 bits* that meet the following: [Doc9303-P11].

***Developer note:***
1. *Session keys of 112 bits length are generated when secure messaging is based on Triple-DES.*

2. *Session keys of 128, 192 and 256 bits lengths are generated when secure messaging is based on AES.*
3. *The complete list of supported elliptic curves is given in A.2.*

***Application note 26 from [PP_PACE]:***
*The TOE generates a shared secret value K with the terminal during the PACE protocol, see [Doc9303-P11]. This protocol may be based on the Diffie-Hellman Protocol compliant to PKCS#3 (i.e. modulo arithmetic based cryptographic algorithm, cf. [PKCS#3]) or on the ECDH compliant to TR-03111 [TR03111] (i.e. the elliptic curve cryptographic algorithm ECKA, cf. [Doc9303-P11] and [TR03111] for details). The shared secret value K is used for deriving the AES or DES session keys for message encryption and message authentication (PACE-KMAC, PACE KENC) according to [Doc9303-P11] for the TSF required by FCS_COP.1/PACE_ENC and FCS_COP.1/PACE_MAC.*

***Developer note:***
*The TOE uses ECDH to generate a shared secret value. Then, the shared secret value is used for deriving the Triple-DES or AES session keys for message encryption and message authentication.*

***Application note 27 from [PP_PACE]:***
*FCS_CKM.1/DH_PACE implicitly contains the requirements for the hashing functions used for key derivation by demanding compliance to [Doc9303-P11].*

***Developer note:***
1. *The TOE uses SHA-1 to derive 112 (Triple-DES) and 128 (AES) bits session keys.*
2. *The TOE uses SHA-256 to derive 192 (AES) and 256 (AES) bits session keys.*

### FCS_CKM.1/CA [key generation, key import]

*FCS_CKM.1.1/CA*

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *based on an ECDH protocol* and specified cryptographic key sizes *of 112 bits, 128 bits, 192 bits, 256 bits* that meet the following: *based on an ECDH protocol compliant to [TR03111].*

***Developer note:***
1. *Session keys of 112 bits length are generated when secure messaging is based on Triple-DES.*
2. *Session keys of 128, 192 and 256 bits lengths are generated when secure messaging is based on AES.*
3. *The complete list of supported elliptic curves is given in A.2.*

***Application note 12 from [PP_EAC]:***
*FCS_CKM.1/CA implicitly contains the requirements for the hashing functions used for key derivation by demanding compliance to [TR03110-1].*

***Developer note:***
1. *The TOE uses SHA-1 to derive 112 (Triple-DES) and 128 (AES) bits session keys.*
2. *The TOE uses SHA-256 to derive 192 (AES) and 256 (AES) bits session keys.*

***Application note 13 from [PP_EAC]:***
*The TOE generates a shared secret value with the terminal during the Chip Authentication Protocol Version 1, see [TR03110-1]. This protocol may be based on the Diffie-Hellman Protocol compliant to PKCS#3 (i.e. modulo arithmetic based cryptographic algorithm, cf. [PKCS#3]) or on the ECDH compliant to TR-03111 (i.e. an elliptic curve cryptography algorithm) (cf. [TR03111], for details). The shared secret value is used to derive the Chip Authentication Session Keys used for encryption and MAC computation for secure messaging (defined in Key Derivation Function [TR03110-1]).*

***Developer note:***
*The TOE uses ECDH to generate a shared secret value. Then, the shared secret value is used for deriving the Triple-DES or AES session keys for message encryption and message authentication.*

***Application note 14 from [PP_EAC]:***
*The TOE shall implement the hash function SHA-1 for the cryptographic primitive to derive the keys for secure messaging from any shared secrets of the Authentication Mechanisms. The Chip Authentication Protocol v.1 may use SHA-1 (cf. [TR03110-1]). The TOE may implement additional hash functions SHA-224 and SHA-256 for the Terminal Authentication Protocol v.1 (cf. [TR03110-1] for details).*

***Developer note:***
1. *The TOE uses SHA-1 to derive 112 (Triple-DES) and 128 (AES) bits session keys for secure messaging.*
2. *According to requirements given in the section A.2.3 of [TR03110-3], the bit-length of the hash function shall be greater or equal to the bit-length of the derived key. That is why, the Chip Authentication Protocol implemented by the TOE uses SHA-256 to derive session keys of 192 (AES) and 256 (AES) bits lengths for secure messaging.*
3. *The Terminal Authentication protocol implementation is out of the certification scope.*
4. *Elliptic curves with 224 bit sizes are supported for the backward compatibility, nevertheless these curves are not recommended according to the [TR02102-1].*

***Application note 15 from [PP_EAC]:***
*The TOE shall destroy any session keys in accordance with FCS_CKM.4 from [PP_PACE] after (i) detection of an error in a received command by verification of the MAC and (ii) after successful run of the Chip Authentication Protocol v.1. (iii) The TOE shall destroy the PACE Session Keys after generation of a Chip Authentication Session Keys and changing the secure messaging to the Chip Authentication Session Keys. (iv) The TOE shall clear the memory area of any session keys before starting the communication with the terminal in a new after reset-session as required by FDP_RIP.1. Concerning the Chip Authentication keys FCS_CKM.4 is also fulfilled by FCS_CKM.1/CA.*

***Developer note:***
*Session keys are cleared by the application once a secure messaging session is broken due to:*
- *receiving APDU in a plain text,*
- *unsuccessful MAC verification,*
- *unsuccessful APDU decryption,*
- *establishing new secure messaging keys (starting a new session),*
- *card reset resulting with the application selection.*

### FCS_CKM.1/CAPK [key generation, key import]

*FCS_CKM.1.1/CAPK*

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *Generating ECDH / ECDSA keys with Brainpool curve or NIST curve (for length 521 bits)* and specified cryptographic key sizes *of 224, 256, 320, 384, 512 and 521 bits* that meet the following: *[ISO15946-1], [ISO15946-3], [TR03110-1] and [TR03110-3].*

***Developer note:***
1. *The complete list of supported elliptic curves is given in A.2.*

2. *The Chip Authentication key pair can either be generated in the TOE or imported by the Manufacturer or Administrator (MRTD's Personalization Agent role equivalent) (see FMT_MTD.1/CAPK). This SFR has been included as required by [PP_EAC] (see application note after FMT_MTD.1/CAPK). This SFR has been included in this security target in addition to the SFRs defined by the protection profiles claimed in clause 2.2. This extension does not conflict with the strict conformance to the claimed protection profiles.*
3. *Elliptic curves with 224 bit sizes are supported for the backward compatibility, nevertheless these curves are not recommended according to the [TR02102-1].*

#### 6.1.1.2 FCS_CKM.4: Cryptographic key destruction

**FCS_CKM.4 [key generation, key import]**

*FCS_CKM.4.1*

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

***Application note 6 from [EN 419211-2]:***
*The ST writer shall perform the missing operations in the element FCS_CKM.4.1. The specified cryptographic key destruction methods include but are not limited to overwriting the cryptographic key with any fixed or random data e.g. by generation of a new key.*

***Developer note:***
*The ST writer has performed the missing assignments.*

***Application note 4 from [EN 419211-3]:***
*The ST writer shall perform the missing operations in the element FCS_CKM.4.1. The cryptographic key SCD will be destroyed on demand of the signatory. The signatory may want to destruct the SCD stored in the SSCD e.g. after the qualified certificate for the corresponding SVD is not valid any more.*

***Developer note:***
*The ST writer has performed the missing assignments.*

***Application note 28 from [PP_PACE]:***
*The TOE shall destroy the PACE session keys after detection of an error in a received command by verification of the MAC. The TOE shall clear the memory area of any session keys before starting the communication with the terminal in a new after-reset-session as required by FDP_RIP.1.*

***Developer note:***
*Session keys are cleared by the application once a secure messaging session is broken due to:*

- *receiving APDU in a plain text,*
- *unsuccessful MAC verification,*
- *unsuccessful APDU decryption,*
- *establishing new secure messaging keys (starting a new session),*
- *the application selection,*
- *card reset resulting with the application selection.*

#### 6.1.1.3 FCS_COP.1: Cryptographic operation

**FCS_COP.1/RSA [key generation, key import]**

*FCS_COP.1.1/RSA*

The TSF shall perform digital signature creation in accordance with a specified cryptographic algorithm *RSA* and cryptographic key sizes *1024, 2048 and 4096 bits* that meet the following: *[JCAPI3] and [JCOP_UM].*

***Developer note:***
1. *RSA-1024 keys are supported for the backward compatibility, nevertheless these keys **are not** recommended according to the [TR02102-1].*
2. *For a period of use beyond 2023 the present Technical Report Guidance [TR02102-1] recommends using a key length of 3000 bits in order to achieve a compatible level of security for all asymmetric mechanisms.*

**FCS_COP.1/ECDSA [key generation, key import]**

*FCS_COP.1.1/ECDSA*

The TSF shall perform digital signature creation in accordance with a specified cryptographic algorithm *ECDSA-FP* and cryptographic key sizes *224, 256, 320, 384, 512, 521 bits* that meet the following: *[JCAPI3] and [JCOP_UM].*

***Developer note:***
*Elliptic curves with 224 bit sizes are supported for the backward compatibility, nevertheless these curves **are not** recommended according to the [TR02102-1].*

**FCS_COP.1/PACE_ENC [key generation, key import]**

*FCS_COP.1.1/PACE_ENC*

The TSF shall perform secure messaging – encryption and decryption in accordance with a specified cryptographic algorithm *Triple-DES and AES in CBC mode* and cryptographic key sizes *of 112, 128, 192, 256 bits* that meet the following: compliant to [Doc9303-P11].

***Developer note:***
1. *Session keys of 112 bits length are used for Triple-DES.*
2. *Session keys of 128, 192 and 256 bits lengths are used for AES.*

***Application note 29 from [PP_PACE]:***
*This SFR requires the TOE to implement the cryptographic primitive AES or 3DES for secure messaging with encryption of transmitted data and encrypting the nonce in the first step of PACE. The related session keys are agreed between the TOE and the terminal as part of the PACE protocol according to the FCS_CKM.1/DH_PACE (PACE-$K_{ENC}$).*

***Developer note:***
*The TOE uses secure messaging which is implemented by the underlying platform (see [ST_OS] for details).*

**FCS_COP.1/PACE_MAC [key generation, key import]**

*FCS_COP.1.1/PACE_MAC*

The TSF shall perform secure messaging – message authentication code in accordance with a specified cryptographic algorithm *Retail-MAC and CMAC* and cryptographic key sizes *of 112, 128, 192, 256 bits* that meet the following: compliant to [Doc9303-P11].

***Developer note:***
1. *Retail-MAC and session keys of 112 bits length are used when secure messaging is based on Triple-DES algorithm.*
2. *CMAC and session keys of 128, 192 and 256 bits lengths are used when secure messaging is based on AES algorithm.*

***Application note 30 from [PP_PACE]:***
*This SFR requires the TOE to implement the cryptographic primitive for secure messaging with message authentication code over transmitted data. The related session keys are agreed between the TOE and the terminal as part of either the PACE protocol according to the FCS_CKM.1/DH_PACE (PACE-KMAC). Note that in accordance with [Doc9303-P11] the (two-key) Triple-DES could be used in Retail mode for secure messaging.*

***Developer note:***
*The TOE uses secure messaging which is implemented by the underlying platform (see [ST_OS] for details).*

**FCS_COP.1/CA_ENC [key generation, key import]**

*FCS_COP.1.1/CA_ENC*

The TSF shall perform secure messaging – encryption and decryption in accordance with a specified cryptographic algorithm *Triple-DES and AES* and cryptographic key sizes *of 112, 128, 192, 256 bits* that meet the following: [Doc9303-P11].

***Developer note:***
1. *Session keys of 112 bits length are used for Triple-DES.*
2. *Session keys of 128, 192 and 256 bits lengths are used for AES.*

***Application note 16 from [PP_EAC]:***
*This SFR requires the TOE to implement the cryptographic primitives (e.g. Triple-DES and/or AES) for secure messaging with encryption of the transmitted data. The keys are agreed between the TOE and the terminal as part of the Chip Authentication Protocol Version 1 according to the FCS_CKM.1/CA.*

***Developer note:***
*The TOE uses secure messaging which is implemented by the underlying platform (see [ST_OS] for details).*

**FCS_COP.1/CA_MAC [key generation, key import]**

*FCS_COP.1.1/CA_MAC*

The TSF shall perform secure messaging – message authentication code in accordance with a specified cryptographic algorithm *Retail-MAC and CMAC* and cryptographic key sizes *of 112, 128, 192, 256 bits* that meet the following: [Doc9303-P11].

***Developer note:***
1. *Retail-MAC and session keys of 112 bits length are used when secure messaging is based on Triple-DES algorithm.*
2. *CMAC and session keys of 128, 192 and 256 bits lengths are used when secure messaging is based on AES algorithm.*

***Application note 18 from [PP_EAC]:***
*This SFR requires the TOE to implement the cryptographic primitive for secure messaging with encryption and message authentication code over the transmitted data. The key is agreed between the TSF by Chip Authentication Protocol Version 1 according to the FCS_CKM.1/CA. Furthermore the SFR is used for authentication attempts of a terminal as Personalization Agent by means of the authentication mechanism.*

***Developer note:***
1. *In this ST Administrator role is the MRTD's Personalization Agent role equivalent.*
2. *The TOE uses secure messaging which is implemented by the underlying platform (see [ST_OS] for details).*

***Application note 7 from [EN 419211-2]:***
*The ST writer shall perform the missing operations in the element FCS_COP.1.1. The operations in the element FCS_COP.1.1 shall be appropriate for the SCD/SVD pairs generated according to FCS_CKM.1. Note that for some cryptographic algorithm like RSA padding is important part of the signature creation algorithm.*

***Developer note:***
*In case of RSA signature creation the PKCS#1 RSASSA-PSS scheme is used.*

***Application note 5 from [EN 419211-3]:***
*The ST writer shall perform the missing operations in the element FCS_COP.1.1. The ST writer should consult the notified body or the certification body for the admissible algorithms, cryptographic key sizes and other parameters for algorithms, and standards for digital signature creation by SSCD. The operations in the element FCS_COP.1.1 shall be appropriate for the SCD imported according to FTP_ITC.1/SCD.*

***Developer note:***
*The ST writer has performed the missing assignments.*

### FCS_RND.1 [key generation, key import]

*FCS_RND.1.1*

The TSF shall provide a mechanism to generate random numbers that meet *class PTG.2 according to [AIS20/AIS31]*.

***Application note 31 from [PP_PACE]:***
*This SFR requires the TOE to generate random numbers (random nonce) used for the authentication protocol (PACE) as required by FIA_UAU.4/PACE.*

***Developer note:***
*The TOE uses random numbers generation which is implemented by the underlying platform (see [ST_OS] for details).*

Presented below are security functional requirements for the RNG class PTG.2 taken from [ST_HW]:

**FCS_RNG.1/PTG.2 Random number generation – PTG.2**
*FCS_RNG.1.1/PTG.2*

The TSF shall provide a physical random number generator that implements:
(PTG.2.1) - A total failure test detects a total failure of entropy source immediately when the RNG has started. When a total failure is detected, no random numbers will be output.
(PTG.2.2) - If a total failure of the entropy source occurs while the RNG is being operated, the RNG prevents the output of any internal random number that depends on after the total failure of the entropy source.
(PTG.2.3) - The online test shall detect non-tolerable statistical defects of the raw random number sequence (i) immediately when the RNG has started, and (ii) while the RNG is being operated. The TSF must not output any random numbers before the power-up online test has finished successfully or when a defect has been detected.
(PTG.2.4) - The online test procedure shall be effective to detect non-tolerable weaknesses of the random numbers soon.
(PTG.2.5) - The online test procedure checks the quality of the raw random number sequence. It is triggered at regular intervals or continuously. The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time.
FCS_RNG.1.2/PTG.2 The TSF shall provide octets of bits that meet:
(PTG.2.6) - Test procedure A does not distinguish the internal random numbers from output sequences of an ideal RNG.
(PTG.2.7) - The average Shannon entropy per internal random bit exceeds 0.997.

### 6.1.2 Class FDP: User data protection

The security attributes and related status for the subjects and objects are listed in Table 6.2.

**Table 6.2: Subjects and security attributes for access control**

| Subject or object the security attribute is associated with | Security attribute type | Value of the security attribute |
|---|---|---|
| S.User | Role | R.Admin, R.Sigy |
| S.User | SCD/SVD Management | authorized, not authorized |
| SCD | SCD Operational | no, yes |
| SCD | SCD identifier | arbitrary value |
| SVD | The claimed protection profiles do not define security attributes for SVD. | The claimed protection profiles do not define security attributes for SVD. |

***Application note 8 from [EN 419211-2]:***
***Application note 6 from [EN 419211-3]:***
*The writer of PP or ST may define additional objects and security attributes.*

***Developer note:***
*No additional objects or security attributes have been defined by the ST writer.*

#### 6.1.2.1 FDP_ACC.1: Subset access control

**FDP_ACC.1/Signature_Creation [key generation, key import]**

*FDP_ACC.1.1/Signature_Creation*

The TSF shall enforce the Signature Creation SFP on:

1. subjects: S.User;
2. objects: DTBS/R, SCD;
3. operations: signature creation.

**FDP_ACC.1/SCD/SVD_Generation [key generation]**

*FDP_ACC.1.1/SCD/SVD_Generation*

The TSF shall enforce the SCD/SVD Generation SFP on:

1. subjects: S.User,
2. objects: SCD, SVD,
3. operations: generation of SCD/SVD pair.

**FDP_ACC.1/SVD_Transfer [key generation]**

*FDP_ACC.1.1/SVD_Transfer*

The TSF shall enforce the SVD Transfer SFP on:

1. subjects: S.User;
2. objects: SVD;
3. operations: export.

**FDP_ACC.1/SCD_Import [key import]**

*FDP_ACC.1.1/SCD_Import*

The TSF shall enforce the SCD Import SFP on

1. subjects: S.User,
2. objects: SCD,
3. operations: import of SCD.

#### 6.1.2.2 FDP_ACF.1: Security attribute based access control

**FDP_ACF.1/Signature_Creation [key generation, key import]**

*FDP_ACF.1.1/Signature_Creation*

The TSF shall enforce the Signature Creation SFP to objects based on the following:

1. the S.User is associated with the security attribute “Role”; and
2. the SCD with the security attribute “SCD Operational”.

*FDP_ACF.1.2/Signature_Creation*

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

R.Sigy is allowed to create electronic signatures for DTBS/R with SCD which security attribute “SCD operational” is set to “yes”.

*FDP_ACF.1.3/Signature_Creation*

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none.

*FDP_ACF.1.4/Signature_Creation*

The TSF shall explicitly deny access of subjects to objects based on the following additional rules:

S.User is not allowed to create electronic signatures for DTBS/R with SCD which security attribute "SCD operational" is set to "no".

**FDP_ACF.1/SCD/SVD_Generation [key generation]**

*FDP_ACF.1.1/SCD/SVD_Generation*

The TSF shall enforce the SCD/SVD Generation SFP to objects based on the following: the user S.User is associated with the security attribute "SCD/SVD Management".

*FDP_ACF.1.2/SCD/SVD_Generation*

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

S.User with the security attribute "SCD/SVD Management" set to "authorized" is allowed to generate SCD/SVD pair.

*FDP_ACF.1.3/SCD/SVD_Generation*

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none.

*FDP_ACF.1.4/SCD/SVD_Generation*

The TSF shall explicitly deny access of subjects to objects based on the following additional rules:

S.User with the security attribute "SCD/SVD management" set to "not authorized" is not allowed to generate SCD/SVD pair.

**FDP_ACF.1/SVD_Transfer [key generation]**

*FDP_ACF.1.1/SVD_Transfer*

The TSF shall enforce the SVD Transfer SFP to objects based on the following:

1. the S.User is associated with the security attribute Role;
2. the SVD.

*FDP_ACF.1.2/SVD_Transfer*

The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *R.Admin and R.Sigy* is allowed to export SVD.

*FDP_ACF.1.3/SVD_Transfer*

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none.

*FDP_ACF.1.4/SVD_Transfer*

The TSF shall explicitly deny access of subjects to objects based on the following additional rules: none.

***Application note 9 from [EN 419211-2]:***
*The ST writer shall perform the operation in the element FDP_ACF.1.2/SVD_Transfer (SSCD) according to the access control rules provided by the TOE for SVD export. The access control rules may depend on TOE lifecycle as shown in the following examples:*

- *The Administrator is authorized to generate the SCD/SVD key pair according to FDP_ACF.1: SCD/SVD_Generation (SSCD) and to export the SVD before the signatory role (RAD) is created. This allows identification of a particular instance of the TOE by means of the SVD;*
- *The Administrator is authorized to generate the SCD/SVD key pair according to FDP_ACF.1: SCD/SVD_Generation (SSCD) and only the signatory is allowed to export the SVD to the CGA. This allows determination whether the signatory has control over the TOE instantiation and the certificate may be generated;*
- *The signatory is authorized to generate the SCD/SVD key pair according to FDP_ACF.1: SCD/SVD_Generation (SSCD) and to export the SVD to the CGA to apply for the certificate.*

*[EN 419211-2] does not require the TOE to protect the integrity and authenticity of the exported SVD public key but requires such protection by the operational environment. If the operational environment does not provide sufficient security measures for the CGA to ensure the authenticity of the public key the TOE shall implement additional security functions to support the export of public keys with integrity and data origin authentication. See [EN 419211-4] for additional requirements for use of an SSCD in an environment that cannot provide such protection.*

***Developer note:***
*The ST writer has performed the missing selection.*

**FDP_ACF.1/SCD_Import [key import]**

*FDP_ACF.1.1/SCD_Import*

The TSF shall enforce the SCD Import SFP to objects based on the following:

the S.User is associated with the security attribute “SCD/SVD Management”.

*FDP_ACF.1.2/SCD_Import*

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

S.User with the security attribute “SCD/SVD Management” set to “authorized” is allowed to import SCD.

*FDP_ACF.1.3/SCD_Import*

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none.

*FDP_ACF.1.4/SCD_Import*

The TSF shall explicitly deny access of subjects to objects based on the following additional rules:

S.User with the security attribute “SCD/SVD management” set to “not authorized” is not allowed to import SCD.

#### 6.1.2.3 FDP_UIT.1: Data exchange integrity

**FDP_UIT.1/DTBS [key generation, key import]**

*FDP_UIT.1.1/DTBS*

The TSF shall enforce the Signature Creation SFP to receive user data in a manner protected from modification and insertion errors.

*FDP_UIT.1.2/DTBS*

The TSF shall be able to determine on receipt of user data, whether modification and insertion has occurred.

#### 6.1.2.4 FDP_RIP.1: Subset residual information protection

**FDP_RIP.1 [key generation, key import]**

*FDP_RIP.1.1*

The TSF shall ensure that any previous information content of a resource is made unavailable upon the de-allocation of the resource from the following objects:

1. SCD,
2. Session Keys (immediately after closing related communication session),
3. the ephemeral private key-$SK_{PICC}$-PACE (by having generated a DH shared secret K[4]),

[4] according to [SAC]

4. none.

The following data persistently stored by the TOE shall have the user data attribute "integrity checked persistent stored data":

1. SCD;
2. SVD (if persistently stored by the TOE).

The DTBS/R temporarily stored by the TOE has the user data attribute "integrity checked stored data".

***Application note 42 from [PP_PACE]:***
*The functional family FDP_RIP possesses such a general character, so that it is applicable not only to user data (as assumed by the class FDP), but also to TSF-data; in this respect it is similar to the functional family FPT_EMS. Applied to cryptographic keys, FDP_RIP.1 requires a certain quality metric ('any previous information content of a resource is made unavailable') for key's destruction in addition to FCS_CKM.4 that merely requires a fact of key destruction according to a method/standard*

***Developer note:***
*The TOE uses dedicated Java Card objects provided by the platform (see [ST_OS]for details) to ensure secure deleting and de-allocation described above.*

#### 6.1.2.5 FDP_SDI.2: Stored data integrity monitoring and action

**FDP_SDI.2/Persistent [key generation, key import]**

*FDP_SDI.2.1/Persistent*

The TSF shall monitor user data stored in containers controlled by the TSF for integrity error on all objects, based on the following attributes: integrity checked stored data.

*FDP_SDI.2.2/Persistent*

Upon detection of a data integrity error, the TSF shall:

1. prohibit the use of the altered data;
2. inform the S.Sigy about integrity error.

**FDP_SDI.2/DTBS [key generation, key import]**

*FDP_SDI.2.1/DTBS*

The TSF shall monitor user data stored in containers controlled by the TSF for integrity error on all objects, based on the following attributes: integrity checked stored DTBS.

*FDP_SDI.2.2/DTBS*

Upon detection of a data integrity error, the TSF shall:

1. prohibit the use of the altered data;
2. inform the S.Sigy about integrity error.

***Application note 10 from [EN 419211-2]:***
***Application note 8 from [EN 419211-3]:***
*The integrity of TSF data like RAD shall be protected to ensure the effectiveness of the user authentication. This protection is a specific aspect of the security architecture (cf. ADV_ARC.1).*

#### 6.1.2.6 FDP_DAU.2: Data Authentication with Identity of Guarantor

**FDP_DAU.2/SVD [key generation]**

*FDP_DAU.2.1/SVD*

The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of SVD.

*FDP_DAU.2.2/SVD*

The TSF shall provide CGA with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

#### 6.1.2.7 FDP_ITC.1: Import of user data without security attributes

**FDP_ITC.1/SCD [key import]**

*FDP_ITC.1.1/SCD*

The TSF shall enforce the SCD Import SFP when importing user data, controlled under the SFP, from outside of the TOE.

*FDP_ITC.1.2/SCD*

The TSF shall ignore any security attributes associated with the ~~user data~~ **SCD** when imported from outside the TOE.

*FDP_ITC.1.3/SCD*

The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: *none*.

***Developer note:***
*The TOE enforces strict compliance to "SCD Import SFP" as stated in FDP_ITC.1.1/SCD. No additional rule is introduced, so FDP_ITC.1.3/SCD assigns "additional importation control rules" to "none".*

#### 6.1.2.8 FDP_UCT.1: Basic data exchange confidentiality

**FDP_UCT.1/SCD [key import]**

*FDP_UCT.1.1/SCD*

The TSF shall enforce the SCD Import SFP to receive ~~user data~~ **SCD** in a manner protected from unauthorized disclosure.

***Application note 7 from [EN 419211-3]:***
*The component FDP_UCT.1/SCD requires the TSF to ensure the confidentiality of the SCD during import. The refinement substituting "user data" by "SCD" highlights that confidentiality of other imported user data like DTBS is not required.*

### 6.1.3 Class FIA: Identification and authentication

#### 6.1.3.1 FIA_UID.1: Timing of identification

**FIA_UID.1 [key generation, key import]**

*FIA_UID.1.1*

The TSF shall allow:

1. self-test according to FPT_TST.1;
2. *none*.

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

*FIA_UID.1.2*

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

***Application note 11 from [EN 419211-2]:***
***Application note 9 from [EN 419211-3]:***
*The ST writer shall perform the missing operation in the element FIA_UID.1.1. The list of additional TSF-mediated actions may be empty (i.e. assignment "none") or include TSF-mediated actions like establishing a trusted path between the user using the HI of an external device. The TOE may identify the user by default or by selection of the role and RAD against which the authentication will be performed. Identification by default is normally linked to the TOE lifecycle, e.g. the TOE may identify by default the Administrator before the signatory's RAD is created and the signatory if signatory's RAD exists. In case of multi-application smart cards (i.e. the smart card provides more than the signature creation application) the user identifies themselves as signatory by selection of the signature application directory file and therefore the PIN authentication will be performed against the signatory PIN. The user may identify themselves as Administrator by selection of an authentication key as Administrator and therefore authentication will be performed by external authenticate or mutual device authentication.*

***Developer note:***
*The ST writer has performed the missing assignment. The TOE identifies the user by the role and RAD against which the authentication will be performed. The user is identified as Signatory if authentication with PIN has been performed. In case of authentication with Administrator Key (see 1.3.4.1), the user is identified as Administrator.*

**FIA_UID.1/PACE [key generation, key import]**

*FIA_UID.1.1/PACE*

The TSF shall allow:

1. to establish a communication channel,
2. carrying out the PACE Protocol according to [Doc9303-P11],
3. to carry out the Chip Authentication Protocol v.1 according to [TR03110-1],
4. *none*.

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

*FIA_UID.1.2/PACE*

The TSF shall require each user to be successfully identified before allowing any other TSF mediated actions on behalf of that user.

***Application note 33 from [PP_PACE]:***
*User identified after a successfully performed PACE protocol is a PACE authenticated BIS PACE. Please note that neither CAN nor MRZ effectively represent secrets (but other PACE passwords may do so), but are restricted-revealable; i.e. it is either the travel document holder itself or an authorized other person or device (BIS-PACE).*

***Developer note:***
*The TOE supports the following PACE passwords: MRZ, CAN, PIN and PUK. PIN and PUK are only known to the Signatory (MRTD's travel document holder role equivalent).*

***Application note 20 from [PP_EAC]:***
*The SFR FIA_UID.1/PACE in [PP_EAC] covers the definition in [PP_PACE] and extends it by EAC aspect 4. This extension does not conflict with the strict conformance to [PP_PACE].*

***Developer note:***
*Only Chip Authentication protocol definition is used in this ST as Terminal Authentication protocol is out of the certification scope.*

***Application note 21 from [PP_EAC]:***
*In the Phase 2 "Manufacturing of the TOE" the Manufacturer is the only user role known to the TOE which writes the Initialization Data and/or Pre-personalization Data in the audit records of the IC. The travel document manufacturer may create the user role Personalization Agent for transition from Phase 2 to Phase 3 "Personalization of the travel document". The users in role Personalization Agent identify themselves by means of selecting the authentication key. After personalization in the Phase 3 the PACE domain parameters, the Chip Authentication data and Terminal Authentication Reference Data are written into the TOE. The Inspection System is identified as default user after power up or reset of the TOE i.e. the TOE will run the PACE protocol, to gain access to the Chip Authentication Reference Data and to run the Chip Authentication Protocol Version 1. After successful authentication of the chip the terminal may identify itself as (i) Extended Inspection System by selection of the templates for the Terminal Authentication Protocol Version 1 or (ii) if necessary and available by authentication as Personalization Agent (using the Personalization Agent Key).*

***Developer note:***
1. *In this ST Administrator role is the MRTD's Personalization Agent role equivalent.*
2. *In the Phase 2 of the life cycle, the Manufacturer is the only user role known to the TOE.*
3. *Transition from Phase 2 to Phase 3 of the life cycle creates the user role Administrator and involves permanent blocking of the user role Manufacturer.*

4. *Transition from Phase 3 to Phase 4 of the life cycle creates the user role Inspection System and permanently blocks the user role Administrator.*

***Application note 22 from [PP_EAC]:***
*User identified after a successfully performed PACE protocol is a terminal. Please note that neither CAN nor MRZ effectively represent secrets, but are restricted-revealable; i.e. it is either the travel document holder itself or an authorized other person or device (Basic Inspection System with PACE).*

***Developer note:***
*The TOE supports the following PACE passwords: MRZ, CAN, PIN and PUK. PIN and PUK are only known to the Signatory (MRTD's travel document holder role equivalent).*

***Application note 23 from [PP_EAC]:***
*In the life-cycle phase 'Manufacturing' the Manufacturer is the only user role known to the TOE. The Manufacturer writes the Initialization Data and/or Pre-personalization Data in the audit records of the IC. Please note that a Personalization Agent acts on behalf of the travel document issuer under his and CSCA and DS policies. Hence, they define authentication procedure(s) for Personalization Agents. The TOE must functionally support these authentication procedures being subject to evaluation within the assurance components ALC_DEL.1 and AGD_PRE.1. The TOE assumes the user role 'Personalization Agent', when a terminal proves the respective Terminal Authorization Level as defined by the related policy (policies).*

***Developer note:***
1. *In this ST Administrator role is the MRTD's Personalization Agent role equivalent.*
2. *Administrator is authenticated to the TOE using Global Platform with Secure Channel Protocol '03' (SCP03). Global Platform functionality was fully implemented by the platform.*

#### 6.1.3.2 FIA_UAU.1: Timing of authentication

**FIA_UAU.1 [key generation, key import]**

*FIA_UAU.1.1*

The TSF shall allow:

1. self-test according to FPT_TST.1;
2. identification of the user by means of TSF required by FIA_UID.1;
3. *establishing a trusted channel between the CGA and the TOE by means of TSF required by FTP_ITC.1/SVD;*
4. *establishing a trusted channel between the HID and the TOE by means of TSF required by FTP_ITC.1/VAD;*
5. *none.*

on behalf of the user to be performed before the user is authenticated.

***Application note 12 from [EN 419211-2]:***
***Application note 10 from [EN 419211-3]:***
*The ST writer shall perform the missing operation in the element FIA_UAU.1.1. The list of additional TSF-mediated actions may be empty (i.e. assignment "none") or include TSF-mediated actions like establishing a trusted path between the user using the HI of an external device.*

***Application note 1 from [EN 419211-4]:***
*The ST writer shall perform the missing operation in the element FIA_UAU.1.1. The list of additional TSF-mediated actions may be empty (i.e. assignment "none"). [EN 419211-4] performed the*

*operation of the point (3) in the element FIA_UAU.1.1 of [EN 419211-2] by adding the establishment of a trusted channel to the CGA.*

***Application note 5 from [EN 419211-5]:***
*The ST writer shall perform the missing operation in the element FIA_UAU.1.1. The list of additional TSF-mediated actions may be empty (i.e. assignment "none"). [EN 419211-5] performed the operation of the point (4) in the element FIA_UAU.1.1 of [EN 419211-4] by adding the establishment of a trusted channel to HID.*

***Application note 5 from [EN 419211-6]:***
*The ST writer shall perform the missing operation in the element FIA_UAU.1.1. The list of additional TSF-mediated actions may be empty (i.e. assignment "none"). [EN 419211-6] performed the operation of the point (4) in the element FIA_UAU.1.1 of [EN 419211-4] by adding the establishment of a trusted channel to HID.*

***Developer note:***
*The ST writer has performed the missing assignments.*

*FIA_UAU.1.2*

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

**FIA_UAU.1/PACE [key generation, key import]**

*FIA_UAU.1.1/PACE*

The TSF shall allow:

1. to establish a communication channel,
2. carrying out the PACE Protocol according to [Doc9303-P11],
3. to identify themselves by selection of the authentication key,
4. to carry out the Chip Authentication Protocol Version 1 according to [TR03110-1],
5. *none*.

on behalf of the user to be performed before the user is authenticated.

*FIA_UAU.1.2/PACE*

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

***Application note 34 from [PP_PACE]:***
*The user authenticated after a successfully performed PACE protocol is a PACE authenticated BIS-PACE. Please note that neither CAN nor MRZ effectively represent secrets (but other PACE passwords may do so), but are restricted-revealable; i.e. it is either the travel document holder itself or an authorized other person or device (BIS-PACE). If PACE was successfully performed, secure messaging is started using the derived session keys (PACE-$K_{MAC}$, PACE $K_{ENC}$), cf. FTP_ITC.1/PACE.*

***Developer note:***
*The TOE supports the following PACE passwords: MRZ, CAN, PIN and PUK. PIN and PUK are only known to the Signatory (MRTD's travel document holder role equivalent).*

***Application note 24 from [PP_EAC]:***
*The SFR FIA_UAU.1/PACE in [PP_EAC] covers the definition in [PP_PACE] and extends it by EAC aspect 5. This extension does not conflict with the strict conformance to [PP_PACE].*

***Developer note:***
*Only Chip Authentication protocol definition is used in this ST as Terminal Authentication protocol is out of the certification scope.*

***Application note 25 from [PP_EAC]:***
*The user authenticated after a successfully performed PACE protocol is a terminal. Please note that neither CAN nor MRZ effectively represent secrets, but are restricted-revealable; i.e. it is either the travel document holder itself or an authorized other person or device (BIS PACE). If PACE was successfully performed, secure messaging is started using the derived session keys (PACE-$K_{MAC}$, PACE-$K_{ENC}$), cf. FTP_ITC.1/PACE.*

***Developer note:***
*The TOE supports the following PACE passwords: MRZ, CAN, PIN and PUK. PIN and PUK are only known to the Signatory (MRTD's travel document holder role equivalent).*

### FIA_UAU.4/PACE [key generation, key import]

*FIA_UAU.4.1/PACE*

The TSF shall prevent reuse of authentication data related to:

1. PACE Protocol according to [Doc9303-P11],
2. Authentication Mechanism based on *Global Platform SCP03,*
3. none.

***Application note 35 from [PP_PACE]:***
*For the PACE protocol, the TOE randomly selects a nonce s of 128 bits length being (almost) uniformly distributed.*

***Developer note:***
*As input of a generic mapping function required by the PACE protocol and used by the TOE has to be of the same length as an elliptic curve base point order, the selected nonce is extended with the leading zeros to the required length.*

***Application note 26 from [PP_EAC]:***
*The SFR FIA_UAU.4.1 in [PP_EAC] covers the definition in [PP_PACE] and extends it by the EAC aspect 3. This extension does not conflict with the strict conformance to [PP_PACE]. The generation of random numbers (random nonce) used for the authentication protocol (PACE) and Terminal Authentication as required by FIA_UAU.4/PACE is required by FCS_RND.1 from [PP_PACE].*

***Developer note:***
*Only Chip Authentication protocol definition is used in this ST as Terminal Authentication protocol is out of the certification scope.*

***Application note 27 from [PP_EAC]:***
*The authentication mechanisms may use either a challenge freshly and randomly generated by the TOE to prevent reuse of a response generated by a terminal in a successful authentication attempt. However, the authentication of Personalization Agent may rely on other mechanisms ensuring protection against replay attacks, such as the use of an internal counter as a diversifier.*

***Developer note:***
1. *In this ST Administrator role is the MRTD's Personalization Agent role equivalent.*

2. *All authentication mechanisms listed in FIA_UAU.4.1/PACE use challenges freshly and randomly generated by the TOE.*

**FIA_UAU.5/PACE [key generation, key import]**

*FIA_UAU.5.1/PACE*

The TSF shall provide:

1. PACE Protocol according to [Doc9303-P11],
2. Secure messaging in MAC-ENC mode according to [Doc9303-P11],
3. Symmetric Authentication Mechanism based on *Global Platform SCP03,*
4. none.

to support user authentication.

*FIA_UAU.5.2/PACE*

The TSF shall authenticate any user's claimed identity according to the following rules:

1. Having successfully run the PACE protocol the TOE accepts only received commands with correct message authentication code sent by means of secure messaging with the key agreed with the terminal by means of the PACE protocol.
2. The TOE accepts the authentication attempt as Personalization Agent by the Authentication Mechanism with Personalization Agent Key(s).
3. After run of the Chip Authentication Protocol Version 1 the TOE accepts only received commands with correct message authentication code sent by means of secure messaging with key agreed with the terminal by means of the Chip Authentication Mechanism v1.
4. *none*.

***Developer note:***
*In this ST Administrator role is the MRTD's Personalization Agent role equivalent.*

***Application note 36 from [PP_PACE]:***
*Please note that Passive Authentication does not authenticate any TOE's user, but provides evidence enabling an external entity (the terminal connected) to prove the origin of e-passport application.*

***Developer note:***
*Passive Authentication mechanism is out of the certification scope.*

***Application note 28 from [PP_EAC]:***
*The SFR FIA_UAU.5.1/PACE in [PP_EAC] covers the definition in [PP_PACE] and extends it by EAC aspects 4), 5), and 6). The SFR FIA_UAU.5.2/PACE in [PP_EAC] covers the definition in [PP_PACE] and extends it by EAC aspects 2), 3), 4) and 5). These extensions do not conflict with the strict conformance to [PP_PACE].*

***Developer note:***
*Only Chip Authentication protocol definition is used in this ST as Terminal Authentication protocol is out of the certification scope.*

**FIA_UAU.6/PACE [key generation, key import]**

*FIA_UAU.6.1/PACE*

The TSF shall re-authenticate the user under the conditions each command sent to the TOE after successful run of the PACE protocol shall be verified as being sent by the PACE terminal.

***Application note 37 from [PP_PACE]:***
*The PACE protocol specified in [Doc9303-P11] starts secure messaging used for all commands exchanged after successful PACE authentication. The TOE checks each command by secure messaging in encrypt-then-authenticate mode based on CMAC or Retail-MAC, whether it was sent by the successfully authenticated terminal (see FCS_COP.1/PACE_MAC for further details). The TOE does not execute any command with incorrect message authentication code. Therefore, the TOE re-authenticates the terminal connected, if a secure messaging error occurred, and accepts only those commands received from the initially authenticated terminal.*

***Developer note:***
1. *The TOE uses Retail-MAC or CMAC to verify APDUs protected with secure messaging.*
2. *Once APDU with incorrect MAC is received, the TOE breaks secure messaging session.*

**FIA_UAU.6/EAC [key generation, key import]**

*FIA_UAU.6.1/EAC*

The TSF shall re-authenticate the user under the conditions each command sent to the TOE after successful run of the Chip Authentication Protocol Version 1 shall be verified as being sent by the Inspection System.

***Application note 29 from [PP_EAC]:***
*The Password Authenticated Connection Establishment and the Chip Authentication Protocol specified in [Doc9303] include secure messaging for all commands exchanged after successful authentication of the Inspection System. The TOE checks by secure messaging in MAC_ENC mode each command based on a corresponding MAC algorithm whether it was sent by the successfully authenticated terminal (see FCS_COP.1/CA_MAC for further details). The TOE does not execute any command with incorrect message authentication code. Therefore the TOE re-authenticates the user for each received command and accepts only those commands received from the previously authenticated user.*

***Developer note:***
1. *The TOE uses Retail-MAC or CMAC to verify APDUs protected with secure messaging.*
2. *Once APDU with incorrect MAC is received, the TOE breaks secure messaging session.*

### 6.1.3.3 FIA_AFL.1: Authentication failure handling

**FIA_AFL.1 [key generation, key import]**

*FIA_AFL.1.1*

The TSF shall detect when an administrator configurable positive integer within the range from 1 to 59 unsuccessful authentication attempts occur related to consecutive failed authentication attempts.

***Developer note:***
*The SCP03 protocol implemented by the platform is used to authenticate Administrator role in preparation phase. Maximum value of the Authentication Retry Counter (ARC) is equal to 59 and it cannot be modified. To protect against brute force attacks the additional delay is added after three unsuccessful authentication attempts.*

*FIA_AFL.1.2*

When the defined number of unsuccessful authentication attempts has been met, the TSF shall block RAD.

***Application note 13 from [EN 419211-2]:***
***Application note 11 from [EN 419211-3]:***
*The ST writer shall perform the missing operation in the element FIA_AFL.1.1. The assignment shall be consistent with the implemented authentication mechanism and the resistant against attacks with high attack potential.*

***Developer note:***
*The ST writer has performed missing selection and assignment.*

#### 6.1.3.4 FIA_API.1: Authentication Proof of Identity

**FIA_API.1 [key generation]**

*FIA_API.1.1*

The TSF shall provide Chip Authentication Protocol Version 1 according to [TR03110-1] to prove the identity of the SSCD.

***Application note 2 from [EN 419211-4]:***
*The ST writer shall perform the missing operation in the element FIA_API.1.1. Via the authentication mechanism to be assigned here the TOE has to be able to authenticate itself as SSCD to the CGA, using authentication data implemented in the TOE during pre-initialization phase.*

***Application note 30 from [PP_EAC]:***
*This SFR requires the TOE to implement the Chip Authentication Mechanism v.1 specified in [TR03110-1]. The TOE and the terminal generate a shared secret using the Diffie-Hellman Protocol (DH or ECDH) and two session keys for secure messaging in ENC_MAC mode according to [Doc9303]. The terminal verifies by means of secure messaging whether the travel document's chip was able or not to run his protocol properly using its Chip Authentication Private Key corresponding to the Chip Authentication Key (EF.DG14).*

***Developer note:***
*The TOE implements the Chip Authentication Mechanism v.1 based on ECDH.*

### 6.1.4 Class FMT: Security management

#### 6.1.4.1 FMT_SMR.1: Security roles

**FMT_SMR.1 [key generation, key import]**

*FMT_SMR.1.1*

The TSF shall maintain the roles R.Admin and R.Sigy.

*FMT_SMR.1.2*

The TSF shall be able to associate users with roles.

**FMT_SMR.1/PACE [key generation, key import]**

*FMT_SMR.1.1/PACE*

The TSF shall maintain the roles

1. Manufacturer,
2. Personalization Agent,
3. Terminal,
4. PACE authenticated BIS-PACE,
5. none.

*FMT_SMR.1.2/PACE*

The TSF shall be able to associate users with roles.

***Application note 47 from [PP_PACE]:***
*For explanation on the role Manufacturer and Personalization Agent please refer to the glossary. The role Terminal is the default role for any terminal being recognized by the TOE as not PACE authenticated BIS-PACE ('Terminal' is used by the travel document presenter).*
*The TOE recognizes the travel document holder or an authorized other person or device (BIS PACE) by using PACE authenticated BIS-PACE (FIA_UAU.1/PACE).*

***Application note 37 from [PP_EAC]:***
*The SFR FMT_SMR.1.1/PACE in [PP_EAC] covers the definition in [PP_PACE] and extends it by 5) to 8). This extension does not conflict with the strict conformance to [PP_PACE].*

***Developer note:***
*Terminal Authentication protocol is out of the certification scope.*

### 6.1.4.2 FMT_SMF.1: Security management functions

**FMT_SMF.1 [key generation, key import]**

*FMT_SMF.1.1*

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

1. creation and modification of RAD,
2. enabling the signature creation function,
3. modification of the security attribute SCD/SVD management, SCD operational,
4. change the default value of the security attribute SCD Identifier,
5. Initialization,
6. Pre-personalization,
7. Personalization,
8. Configuration.

***Developer note:***
*Points 5-8 are used in terms of the PACE and CA protocols implementation as the TOE is the SSCD with support of MRTD-typical trusted channel implementation.*

***Application note 14 from [EN 419211-2]:***
***Application note 12 from [EN 419211-3]:***
*The ST writer shall perform the missing operation in the element FMT_SMF.1.1. The list of other security management functions to be provided by the TSF may be empty (i.e. assignment "none").*

***Developer note:***
*The ST writer has performed the missing assignment.*

#### 6.1.4.3 FMT_MOF.1: Management of security functions behaviour

**FMT_MOF.1 [key generation, key import]**

*FMT_MOF.1.1*

The TSF shall restrict the ability to enable the functions signature creation function to R.Sigy.

#### 6.1.4.4 FMT_MSA.1: Management of security attributes

**FMT_MSA.1/Admin [key generation, key import]**

*FMT_MSA.1.1/Admin*

The TSF shall enforce the SCD/SVD Generation SFP and SCD Import SFP to restrict the ability to modify and *none* the security attributes SCD/SVD management to R.Admin.

**FMT_MSA.1/Signatory [key generation, key import]**

*FMT_MSA.1.1/Signatory*

The TSF shall enforce the Signature Creation SFP to restrict the ability to modify the security attributes SCD operational to R.Sigy.

#### 6.1.4.5 FMT_MSA.2: Secure security attributes

**FMT_MSA.2 [key generation, key import]**

*FMT_MSA.2.1*

The TSF shall ensure that only secure values are accepted for SCD/SVD Management and SCD operational.

***Application note 15 from [EN 419211-2]:***
***Application note 13 from [EN 419211-3]:***
*The ST writer shall define which values of the security attribute SCD/SVD Management are secure for the TOE and the intended TOE lifecycle. E.g. if the TOE supports generation of SCD/SVD pairs by the signatory and a trusted channel for export of the SVD to the CGA then the subject S.Sigy may or may not be assigned the security attribute SCD/SVD Management to "yes". If the TOE supports the generation of the SCD/SVD pair in the preparation phase in secure environment only the TSF should enforce the assignment of the security attribute SCD/SVD Management of S.Admin to "yes" and of S.Sigy to "no".*

***Developer note:***
*All values defined for security attributes in Table 6.2 are secure for the TOE.*

#### 6.1.4.6 FMT_MSA.3: Static attribute initialization

**FMT_MSA.3 [key generation, key import]**

*FMT_MSA.3.1*

The TSF shall enforce the SCD/SVD Generation SFP, SVD Transfer SFP, SCD Import SFP and Signature Creation SFP to provide restrictive default values for security attributes that are used to enforce the SFP.

*FMT_MSA.3.2*

The TSF shall allow the R.Admin to specify alternative initial values to override the default values when an object or information is created.

#### 6.1.4.7 FMT_MSA.4: Security attribute value inheritance

**FMT_MSA.4 [key generation, key import]**

*FMT_MSA.4.1*

The TSF shall use the following rules to set the value of security attributes:

1. If S.Admin successfully generates an SCD/SVD pair without S.Sigy being authenticated the security attribute “SCD operational of the SCD” shall be set to “no” as a single operation.
2. If S.Sigy successfully generates an SCD/SVD pair the security attribute “SCD operational of the SCD” shall be set to “yes” as a single operation.
3. If S.Admin imports SCD while S.Sigy is not currently authenticated, the security attribute “SCD operational” of the SCD shall be set to “no” after import of the SCD as a single operation.
4. If S.Admin imports SCD while S.Sigy is currently authenticated, the security attribute “SCD operational” of the SCD shall be set to “yes” after import of the SCD as a single operation.

***Developer note:***
*As stated in section 1.3.4.1 PACE-PIN and PACE-PUK are reserved for S.Admin role authentication in Operational phase. If at least one of those passwords is authenticated and S.Sigy authentication conditions are not met, the security attribute “SCD operational” of the SCD shall is set to “no” after import or generation of the SCD.*

***Application note 16 from [EN 419211-2]:***
*The TOE may not support generating an SVD/SCD pair by the signatory alone, in which case rule (2) is not relevant.*

***Developer note:***
*The above Application note contains only information for the security target writer – no action is required.*

#### 6.1.4.8 FMT_MTD.1: Management of TSF data

**FMT_MTD.1/Admin [key generation, key import]**

*FMT_MTD.1.1/Admin*

The TSF shall restrict the ability to create the RAD to R.Admin.

**FMT_MTD.1/Signatory [key generation, key import]**

*FMT_MTD.1.1/Signatory*

The TSF shall restrict the ability to modify and *unblock* the RAD to R.Sigy.

***Application note 17 from [EN 419211-2]:***
***Application note 14 from [EN 419211-3]:***
*The ST writer shall perform the missing operation in the element FMT_MTD.1.1. The missing assignment may be "unblock" or "none".*

***Developer note:***
*The ST writer has performed the missing assignment.*

**FMT_MTD.1/KEY_READ [key generation, key import]**

*FMT_MTD.1.1/KEY_READ*

The TSF shall restrict the ability to read the:

1. PACE passwords,
2. Chip Authentication Private Key,
3. Personalization Agent Keys

to none.

***Application note 45 from [PP_EAC]:***
*The SFR FMT_MTD.1/KEY_READ in [PP_EAC] covers the definition in [PP_PACE] and extends it by additional TSF data. This extension does not conflict with the strict conformance to [PP_PACE].*

***Developer note:***
*In this ST Administrator role is the MRTD's Personalization Agent role equivalent.*

**FMT_MTD.1/CAPK [key generation, key import]**

*FMT_MTD.1.1/CAPK*

The TSF shall restrict the ability to *create, load* the Chip Authentication Private Key to *Personalization Agent*.

***Application note 44 from [PP_EAC]:***
*The component FMT_MTD.1/CAPK is refined by (i) selecting other operations and (ii) defining a selection for the operations "create" and "load" to be performed by the ST writer. The verb "load" means here that the Chip Authentication Private Key is generated securely outside the TOE and written into the TOE memory. The verb "create" means here that the Chip Authentication Private*

*Key is generated by the TOE itself. In the latter case the ST writer shall include an appropriate instantiation of the component FCS_CKM.1/CA as SFR for this key generation. The ST writer shall perform the assignment for the authorized identified roles in the SFR component FMT_MTD.1/CAPK.*

***Developer note:***

1. *In this ST Administrator role is the MRTD's Personalization Agent role equivalent.*
2. *The following operations have been selected: 'load', 'create'.*
3. *Due to selecting the 'create' operation, the following instantiation of the component FCS_CKM.1/CA (as SFR) has been done: FCS_CKM.1/CAPK.*

#### 6.1.4.9 FMT_LIM.1: Limited capabilities

**FMT_LIM.1 [key generation, key import]**

*FMT_LIM.1.1*

The TSF shall be designed in a manner that limits their capabilities so that in conjunction with 'Limited availability (*FMT_LIM.2*)' the following policy is enforced: Deploying test features after TOE delivery do not allow:

1. User Data to be manipulated and disclosed,
2. TSF data to be manipulated or disclosed,
3. software to be reconstructed,
4. substantial information about construction of TSF to be gathered which may enable other attacks, and
5. sensitive User Data to be disclosed.

#### 6.1.4.10 FMT_LIM.2: Limited availability

**FMT_LIM.2.1 [key generation, key import]**

*FMT_LIM.2.1*

The TSF shall be designed in a manner that limits their availability so that in conjunction with 'Limited capabilities (*FMT_LIM.1*)' the following policy is enforced: Deploying test features after TOE delivery do not allow:

1. User Data to be manipulated and disclosed,
2. TSF data to be manipulated or disclosed,
3. software to be reconstructed,
4. substantial information about construction of TSF to be gathered which may enable other attacks, and
5. sensitive User Data to be disclosed.

***Application note 39 form [PP_EAC] (includes Application note 48 from [PP_PACE]):***
*The formulation of "Deploying Test Features …" in FMT_LIM.2.1 might be a little bit misleading since the addressed features are no longer available (e.g. by disabling or removing the respective functionality). Nevertheless the combination of FMT_LIM.1 and FMT_LIM.2 is introduced to provide an optional approach to enforce the same policy.*
*Note that the term "software" in item 3 of FMT_LIM.1.1 and FMT_LIM.2.1 refers to both IC Dedicated and IC Embedded Software.*

***Developer note:***
*Test features are available only in Testing life cycle which is available only during Development Phase.*

### 6.1.5 Class FPT: Protection of the TSF

#### 6.1.5.1 FPT_EMS.1: TOE Emanation

**FPT_EMS.1 [key generation, key import]**

*FPT_EMS.1.1*

The TOE shall not emit *electromagnetic emissions , variations in power consumption or timing during command execution* in excess of *levels that could be measured or analysed in the current state of art* enabling access to:

1. RAD,
2. SCD,
3. Chip Authentication Session Keys,
4. PACE session keys (PACE-$K_{MAC}$, PACE-$K_{ENC}$),
5. the ephemeral private key ephem-$SK_{PICC}$-PACE,
6. Personalization Agent Key(s),
7. Chip Authentication Private Key.

*FPT_EMS.1.2*

The TSF shall ensure *that any users* are unable to use the following interface *contact and/or contactless interface and circuit contacts* to gain access to:

1. RAD,
2. SCD,
3. Chip Authentication Session Keys,
4. PACE session keys (PACE-$K_{MAC}$, PACE-$K_{ENC}$),
5. the ephemeral private key ephem-$SK_{PICC}$-PACE,
6. Personalization Agent Key(s),
7. Chip Authentication Private Key.

***Application note 18 from [EN 419211-2]:***
***Application note 15 from [EN 419211-3]:***
*The TOE shall prevent attacks against the SCD and other secret data where the attack is based on external observable physical phenomena of the TOE. Such attacks may be observable at the interfaces of the TOE or may origin from internal operation of the TOE or may origin by an attacker that varies the physical environment under which the TOE operates. The set of measurable physical phenomena is influenced by the technology employed to implement the TOE. Examples of measurable phenomena are variations in the power consumption, the timing of transitions of internal states, electromagnetic radiation due to internal operation, radio emission.*
*Due to the heterogeneous nature of the technologies that may cause such emanations, evaluation against state-of-the-art attacks applicable to the technologies employed by the TOE is assumed. Examples of such attacks are, but are not limited to, evaluation of TOE's electromagnetic radiation, simple power analysis (SPA), differential power analysis (DPA), timing attacks, etc.*

***Developer note:***
*The TOE uses security mechanisms provided by the hardware (see 1.3.3.1 for hardware details) to ensure protection against attacks described above.*

***Application note 51 from [PP_PACE]:***
*The TOE shall prevent attacks against the listed secret data where the attack is based on external observable physical phenomena of the TOE. Such attacks may be observable at the interfaces of the TOE or may be originated from internal operation of the TOE or may be caused by an attacker that varies the physical environment under which the TOE operates. The set of measurable physical phenomena is influenced by the technology employed to implement the smart card. The travel document's chip has to provide a smart card contactless interface, but may have also (not used by the terminal, but maybe by an attacker) sensitive contacts according to ISO/IEC 7816-2 as well. Examples of measurable phenomena include, but are not limited to variations in the power consumption, the timing of signals and the electromagnetic radiation due to internal operations or data transmissions.*

***Developer note:***
*The TOE uses security mechanisms provided by the hardware (see 1.3.3.1 for hardware details) to ensure protection against attacks described above.*

***Application note 47 from [PP_EAC]:***
*The SFR FPT_EMS.1.1 in [PP_EAC] covers the definition in [PP_PACE] and extends it by EAC aspects 1., 5. and 6. The SFR FPT_EMS.1.2 in [PP_EAC] covers the definition in [PP_PACE] and extends it by EAC aspects 4) and 5). These extensions do not conflict with the strict conformance to [PP_PACE].*

***Application note 48 from [PP_EAC]:***
*The ST writer shall perform the operation in FPT_EMS.1.1 and FPT_EMS.1.2. The TOE shall prevent attacks against the listed secret data where the attack is based on external observable physical phenomena of the TOE. Such attacks may be observable at the interfaces of the TOE or may be originated from internal operation of the TOE or may be caused by an attacker that varies the physical environment under which the TOE operates. The set of measurable physical phenomena is influenced by the technology employed to implement the smart card. The travel document's chip can provide a smart card contactless interface and contact based interface according to ISO/IEC 7816-2 [ISO_7816-2] as well (in case the package only provides a contactless interface the attacker might gain access to the contacts anyway). Examples of measurable phenomena include, but are not limited to variations in the power consumption, the timing of signals and the electromagnetic radiation due to internal operations or data transmissions.*

***Developer note:***
*The TOE uses security mechanisms provided by the hardware (see 1.3.3.1 for hardware details) to ensure protection against attacks described above.*

#### 6.1.5.2 FPT_FLS.1: Failure with preservation of secure state

**FPT_FLS.1 [key generation, key import]**

*FPT_FLS.1.1*

The TSF shall preserve a secure state when the following types of failures occur:

1. self-test according to FPT_TST fails;
2. exposure to operating conditions causing a TOE malfunction,
3. failure detected by TSF according to *FPT_TST.1,*
4. *none.*

***Application note 19 from [EN 419211-2]:***
***Application note 16 from [EN 419211 3]:***
*The ST writer shall perform the missing assignment in the element FPT_FLS.1.1. The assignment (1) addresses failures detected by a failed self-test and requiring appropriate action to prevent security violation. When the TOE is in a secure state the TSF shall not perform any cryptographic operations and all data output interfaces shall be inhibited by the TSF.*

***Developer note:***
*The ST writer has performed the missing assignment.*

#### 6.1.5.3 FPT_PHP.1: Passive detection of physical attack

**FPT_PHP.1 [key generation, key import]**

*FPT_PHP.1.1*

The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF.

*FPT_PHP.1.2*

The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.

#### 6.1.5.4 FPT_PHP.3: Resistance to physical attack

**FPT_PHP.3 [key generation, key import]**

*FPT_PHP.3.1*

The TSF shall resist *physical manipulation and physical probing* to the *TSF* by responding automatically such that the SFRs are always enforced.

***Application note 20 from [EN 419211-2]:***
***Application note 17 from [EN 419211 3]:***
*The TOE will implement appropriate measures to continuously counter physical tampering which may compromise the SCD. The "automatic response" in the element FPT_PHP.3.1 means (1) assuming that there might be an attack at any time and (2) countermeasures are provided at any time. Due to the nature of these attacks the TOE can by no means detect attacks on all of its elements (e.g. the TOE is destroyed). But physical tampering shall not reveal information of the SCD. E.g. the TOE may be physically tampered in power-off state of the TOE (e.g. a smart card), which does not allow TSF for overwriting the SCD but leads to physical destruction of the memory and all information therein about the SCD. In case of physical tampering the TFS may not provide the intended functions for SCD/SVD pair generation or signature creation but ensures the confidentiality of the SCD by blocking these functions. The SFR FPT_PHP.1 requires the TSF to react on physical tampering in a way that the signatory is able to determine whether the TOE was physical tampered or not. For example, the TSF may provide an appropriate message during start-up or the guidance documentation may describe a failure of TOE start-up as indication of physical tampering.*

***Developer note:***
*The ST writer has performed the missing assignments.*

#### 6.1.5.5 FPT_TST.1: TSF testing

**FPT_TST.1 [key generation, key import]**

*FPT_TST.1.1*

The TSF shall run a suite of self tests *before every use* to demonstrate the correct operation of the TSF.

*FPT_TST.1.2*

The TSF shall provide authorized users with the capability to verify the integrity of TSF data.

*FPT_TST.1.3*

The TSF shall provide authorized users with the capability to verify the integrity of TSF.

***Application note 21 from [EN 419211-2]:***
***Application note 18 from [EN 419211 3]:***
*The ST writer shall perform the operations in the element FPT_TST.1.1. The component FPT_TST.1 addresses only the self-test of the TSF. If the TSF relays on security feature of the hardware platform of part of the TOE the ST should consider inclusion FPT_TEE.1 to require the TSF to test these features for correct work of the dependent TSF.*

***Developer note:***
*The ST writer has performed the missing selection and assignment. Self tests required by FPT_TST.1.1 are performed by self-controlling security mechanisms of the platform as well as by additional mechanisms implemented in PWPW SmartApp-ID 5.0 applet itself. These mechanisms are described briefly in [ADV_ARC].*

***Application note 52 from [PP_PACE]:***
*If the travel document's chip uses state of the art smart card technology, it will run some self tests at the request of an authorized user and some self tests automatically. E.g. a self test for the verification of the integrity of stored TSF executable code required by FPT_TST.1.3 may be executed during initial start-up by the 'authorized user' Manufacturer in the life cycle phase 'Manufacturing'. Other self tests may automatically run to detect failures and to preserve the secure state according to FPT_FLS.1 in the phase 'operational use', e.g. to check a calculation with a private key by the reverse calculation with the corresponding public key as a countermeasure against Differential Failure Analysis.*

***Developer note:***
*1. The TOE uses security mechanisms provided by the platform (see [ST_OS] for hardware details) to ensure integrity of stored TSF executable code.*
*2. The TOE automatically verifies the integrity of the TSF-data before every use of these data.*

### 6.1.6 Class FTP: Trusted path/channels

#### 6.1.6.1 FTP_ITC.1: Inter-TSF trusted channel

**FTP_ITC.1/VAD [key generation, key import]**

*FTP_ITC.1.1/VAD*

The TSF shall provide a communication channel between itself and another trusted IT product **HID** 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/VAD*

The TSF shall permit the remote trusted IT product to initiate communication via the trusted channel.

*FTP_ITC.1.3/VAD*

The TSF **or the HID** shall initiate communication via the trusted channel for

1. User authentication according to FIA_UAU.1,
2. *none*.

***Application note 6 from [EN 419211-5]:***
***Application note 6 from [EN 419211-6]:***
*The component FTP_ITC.1/VAD requires the TSF to support a trusted channel established by the HID to send the VAD. The ST writer shall perform the missing operations in the element FTP_ITC.1.3. If the TSF does not enforce the use of trusted channel for other functions the operation in the element FTP_ITC.1.3 is "none". Note the VAD needs protection depending on the authentication methods employed: VAD for authentication by knowledge needs protection in confidentiality; VAD for biometric authentication may need protection in integrity only.*

***Developer note:***
*The ST writer has performed the missing assignment.*

**FTP_ITC.1/DTBS [key generation, key import]**

*FTP_ITC.1.1/DTBS*

The TSF shall provide a communication channel between itself and another trusted IT product **SCA** 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/DTBS*

The TSF shall permit the remote trusted IT product to initiate communication via the trusted channel.

*FTP_ITC.1.3/DTBS*

The TSF or the SCA shall initiate communication via the trusted channel for

1. signature creation,
2. *none*.

***Application note 7 from [EN 419211-5]:***
***Application note 7 from [EN 419211-6]:***
*The component FTP_ITC.1/DTBS requires the TSF to support a trusted channel established by the SCA to send the DTBS. The ST writer shall perform the missing operations in the element FTP_ITC.1.3. If the TSF does not enforce the use of trusted channel for other functions the operation in the element FTP_ITC.1.3 is "none".*

***Developer note:***
*The ST writer has performed the missing assignment.*

**FTP_ITC.1/SVD [key generation]**

*FTP_ITC.1.1/SVD*

The TSF shall provide a communication channel between itself and another trusted IT product **CGA** that is logically distinct from other communication channels and provides ensured identification of its end points and protection of the channel data from modification or disclosure.

*FTP_ITC.1.2/SVD*

The TSF shall permit another trusted IT product to initiate communication via the trusted channel.

*FTP_ITC.1.3/SVD*

The TSF **or the CGA** shall initiate communication via the trusted channel for

1. data Authentication with Identity of Guarantor according to FIA_API.1 and FDP_DAU.2/SVD
2. *none*.

***Application note 3 from [EN 419211-4]:***
*The component FPT_ITC.1/SVD requires the TSF to enforce a trusted channel established by the CGA to export the SVD to the CGA. The ST writer shall perform the missing operations in the element FTP_ITC.1.3. If the TSF does not enforce the use of trusted channel for other functions the operation in the element FTP_ITC.1.3 is "none".*

***Developer note:***
*The ST writer has performed the missing assignment.*

***Application note 4 from [EN 419211-4]:***
*If the ST writer requires the TSF to support (not to enforce) a trusted channel established by the CGA to export the SVD to the CGA than he or she shall use [EN 419211-4] and include a similar component FPT_ITC.1/SVD with assignment "none" in the element FPT_ITC.1.3/SVD.*

***Developer note:***
*The ST writer has performed the missing assignment. Trusted channel is established by PACEv2 and Chip Authentication version 1 protocols. Communication via trusted channel is protected by means of secure messaging mechanism specified for MRTDs. Implementation of PACEv2 and Chip Authentication version 1 protocols is certified according to the Common Criteria (more details in [ASE MRTD]).*

**FTP_ITC.1/SCD [key import]**

*FTP_ITC.1.1/SCD*

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/SCD*

The TSF shall permit another trusted IT product to initiate communication via the trusted channel.

*FTP_ITC.1.3/SCD*

The TSF shall initiate communication via the trusted channel for

1. Data exchange integrity according to FDP_UCT.1/SCD,
2. *none*.

***Application note 19 from [EN 419211-3]:***
*The component FPT_ITC.1 requires the TSF to support a trusted channel established to another trusted IT product generating the SCD/SVD pair for import the SCD as described by FDP_UCT.1/SCD. The ST writer shall perform the missing operations in the element FTP_ITC.1.3/SCD. If the TSF does not enforce the use of trusted channel for other functions the operation in the element FTP_ITC.1.3/SCD is "none".*

***Developer note:***
*The ST writer has performed the missing assignment.*

**FTP_ITC.1/PACE [key generation, key import]**

*FTP_ITC.1.1/PACE*

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/PACE*

The TSF shall permit another trusted IT product to initiate communication via the trusted channel.

*FTP_ITC.1.3/PACE*

The TSF shall ~~initiate~~ **enforce** communication via the trusted channel for any data exchange between the TOE and the Terminal.

***Application note 43 from [PP_PACE]:***
*The trusted IT product is the terminal. In FTP_ITC.1.3/PACE, the word "initiate" is changed to 'enforce", as the TOE is a passive device that can not initiate the communication. All the communication are initiated by the Terminal, and the TOE enforce the trusted channel.*

***Application note 44 from [PP_PACE]:***
*The trusted channel is established after successful performing the PACE protocol (FIA_UAU.1/PACE). If the PACE was successfully performed, secure messaging is immediately started using the derived session keys (PACE-$K_{MAC}$, PACE-$K_{ENC}$): this secure messaging enforces preventing tracing while Passive Authentication and the required properties of operational trusted channel; the cryptographic primitives being used for the secure messaging are as required by FCS_COP.1/PACE_ENC and FCS_COP.1/PACE_MAC. The establishing phase of the PACE trusted channel does not enable tracing due to the requirements FIA_AFL.1/PACE.*

***Developer note:***
*The TOE implements Secure Messaging according to [Doc9303].*

## 6.2 Security assurance requirements

The assurance requirements for the evaluation of the TOE and its development and operating environment are those taken from the assurance package EAL 4 and augmented by taking the following components ALC_DVS.2, ATE_DPT.2 and AVA_VAN.5.

## 6.3 Security requirements rationale

All SSCD security functional requirements and security assurance requirements described in this security target are coming from [EN 419211-2], [EN 419211-3], [EN 419211-4], [EN 419211-5] and [EN 419211-6]. The security requirement rationales stated in those documents applies to this security target.

As the TOE implements secure messaging functionality using PACE and Chip Authentication protocols, additional security functional requirements and security assurance requirements from [PP_PACE] and [PP_EAC] were introduced to cover PACE and CA protocols implementation. The security requirement rationales regarding PACE and CA protocols implementation stated in those documents applies to this security target.

The remaining security requirement *FCS_CKM.1* (related to cryptographic key generation) described in this security target was derived directly from [CC-Part2].

The security objective *OT.Chip_Auth_Proof* "Proof of travel document's chip authenticity" is ensured by the cryptographic key pair generation as required by *FCS_CKM.1/CAPK*. The *FCS_CKM.1/CAPK* requirement was described in chapter 6.1.1.1. NIST and Brainpool elliptic curves with cryptographic key sizes of 224, 256, 320, 384, 512 and 521 were selected for key pair generation. These algorithms are sufficient to generate strong enough key pairs used during Chip Authentication version 1, which will allow proving the travel document's authenticity.

***Developer note:***
*Elliptic curves with 224 bit sizes are supported for the backward compatibility, nevertheless these curves are not recommended according to the [TR02102-1].*

# 7 Target of evaluation summary specification

This section describes all security functions implemented by the TOE and maps their functionalities to SFRs. The mapping allows to demonstrate, that all SFRs defined in this security target have been addressed and each of them is covered by at least one security function.

Each security function has its representation in the module of the application with the signature functionality.

It is important to note that the PWPW SmartApp-ID 5.0 applet implements e-passport as well as signature functionalities. These functionalities were separated using ADFs structures. This separation is controlled by the proprietary concept called the Security Manager, which grants access based on selected ADF to only one functionality, either e-passport or signatory. It means that it is not possible to perform Terminal Authentication in the signatory ADF as well as it is not possible to perform signature operations in the e-passport ADF. Nevertheless there still exist common parts needed for both – e-passport and signature operations. Since this ST represents only signature functionality, therefore only common and related to signature security functionalities were described and mapped.

The common security functionalities are:

- SF.Access,
- SF.ChipAuthentication,
- SF.Configuration,
- SF.FileSystem,
- SF.GPAuthentication,
- SF.PACE,
- SF.PINManager,
- SF.Protection,
- SF.SEManager,
- SF.TrustedChannel.

The e-passport specific security functionalities (out of scope of this documentation) are:

- SF.TerminalAuthentication.

The signature specific security functionalities are:

- SF.KeyManager.

## 7.1 SFR to TSF mapping

**Table 7.1: Functional requirement to TOE security functionality mapping**

| TOE security functional requirement / TOE Security functionality | SF.Access | SF.ChipAuthentication | SF.Configuration | SF.FileSystem | SF.GPAuthentication | SF.PACE | SF.PINManager | SF.Protection | SF.SEManager | SF.Trustedchannel | SF.KeyManager |
|---|---|---|---|---|---|---|---|---|---|---|---|
| FCS_CKM.1/RSA | | | | | | | | | | | X |
| FCS_CKM.1/ECDSA | | | | | | | | | | | X |
| FCS_CKM.1/DH_PACE | | | | | | X | | | | | |
| FCS_CKM.1/CA | | X | | | | | | | | | |
| FCS_CKM.1/CAPK | | X | | | | | | | | | |
| FCS_CKM.4 | | | | | | | | | | | X |
| FCS_COP.1/RSA | | | | | | | | | | | X |
| FCS_COP.1/ECDSA | | | | | | | | | | | X |
| FCS_COP.1/PACE_ENC | | | | | | | | | | X | |
| FCS_COP.1/PACE_MAC | | | | | | | | | | X | |
| FCS_COP.1/CA_ENC | | | | | | | | | | X | |
| FCS_COP.1/CA_MAC | | | | | | | | | | X | |
| FCS_RND.1 | | X | | | | X | | X | | | |
| FDP_ACC.1/Signature_Creation | X | X | X | | | X | X | | X | X | X |
| FDP_ACC.1/SCD/SVD_Generation | X | X | X | | | X | X | | X | X | X |
| FDP_ACC.1/SVD_Transfer | X | X | X | | | X | X | | X | X | X |
| FDP_ACC.1/SCD_Import | X | X | X | | | X | X | | X | X | X |
| FDP_ACF.1/Signature_Creation | X | X | X | | | X | X | | X | X | X |
| FDP_ACF.1/SCD/SVD_Generation | X | X | X | | | X | X | | X | X | X |
| FDP_ACF.1/SVD_Transfer | X | X | X | | | X | X | | X | X | X |
| FDP_ACF.1/SCD_Import | X | X | X | | | X | X | | X | X | X |

**Table 7.1 (continued)**

| TOE security functional requirement / TOE Security functionality | SF.Access | SF.ChipAuthentication | SF.Configuration | SF.FileSystem | SF.GPAuthentication | SF.PACE | SF.PINManager | SF.Protection | SF.SEManager | SF.Trustedchannel | SF.KeyManager |
|---|---|---|---|---|---|---|---|---|---|---|---|
| FDP_UIT.1/DTBS | X | X | X | | | X | X | | X | X | X |
| FDP_RIP.1 | | | | | | | | | | | X |
| FDP_SDI.2/Persistent | X | | | X | | | X | X | | | X |
| FDP_SDI.2/DTBS | X | | | | | | | | | | X |
| FDP_DAU.2/SVD | | | | | | | | | | | X |
| FDP_ITC.1/SCD | | | | | | | | | | | X |
| FDP_UCT.1/SCD | | X | | | | X | | | | X | X |
| FIA_UID.1 | X | | X | | X | | X | | X | | |
| FIA_UID.1/PACE | X | X | | | | X | | | | X | |
| FIA_UAU.1 | X | X | | | | X | | | | X | |
| FIA_UAU.1/PACE | X | X | | | | X | | | | X | |
| FIA_UAU.4/PACE | | | | | X | X | | | | | |
| FIA_UAU.5/PACE | | X | | | X | X | | | | | |
| FIA_UAU.6/PACE | X | | | | | X | | | | X | |
| FIA_UAU.6/EAC | X | X | | | | | | | | X | |
| FIA_AFL.1 | | | | | X | | | | | | |
| FIA_API.1 | X | X | | X | | | | | | | |
| FMT_SMR.1 | X | | | | | | | | X | | |
| FMT_SMR.1/PACE | X | | | | | | | | | | |
| FMT_SMF.1 | X | | | | | | X | | X | | X |
| FMT_MOF.1 | X | | | | | | X | | X | | |

**Table 7.1 (continued)**

| TOE security functional requirement / TOE Security functionality | SF.Access | SF.ChipAuthentication | SF.Configuration | SF.FileSystem | SF.GPAuthentication | SF.PACE | SF.PINManager | SF.Protection | SF.SEManager | SF.Trustedchannel | SF.KeyManager |
|---|---|---|---|---|---|---|---|---|---|---|---|
| FMT_MSA.1/Admin | X | | | | X | | X | | X | | |
| FMT_MSA.1/Signatory | X | | | | | | X | | X | | |
| FMT_MTD.1/KEY_READ | X | X | | | X | X | | | | | |
| FMT_MTD.1/CAPK | X | X | | | | | | | | | |
| FMT_MSA.2 | X | | | | | | | | | | X |
| FMT_MSA.3 | X | | | | X | | X | | | | X |
| FMT_MSA.4 | X | | | | | | X | | X | | X |
| FMT_MTD.1/Admin | X | | | | | | X | | X | | |
| FMT_MTD.1/Signatory | X | | | | | | X | | X | | |
| FMT_LIM.1 | X | | | | | | | | X | | |
| FMT_LIM.2 | X | | | | | | | | X | | |
| FPT_EMS.1 | | X | | | X | X | X | X | | X | X |
| FPT_FLS.1 | | | X | | | | | X | | | |
| FPT_PHP.1 | | | | | | | | X | | | |
| FPT_PHP.3 | | | | | | | | X | | | |
| FPT_TST.1 | | | | | | | | X | | X | |
| FTP_ITC.1/VAD | | | | | | | | | | X | |
| FTP_ITC.1/DTBS | | | | | | | | | | X | |
| FTP_ITC.1/SVD | | | | | | | | | | X | |
| FTP_ITC.1/SCD | | | | | | | | | | X | |
| FTP_ITC.1/PACE | X | | | | | X | | | | X | |

## 7.2 SF.Access

The content is available in the complete Security Target documentation.

## 7.3 SF.ChipAuthentication

The content is available in the complete Security Target documentation.

## 7.4 SF.Configuration

The content is available in the complete Security Target documentation.

## 7.5 SF.FileSystem

The content is available in the complete Security Target documentation.

## 7.6 SF.GPAuthentication

The content is available in the complete Security Target documentation.

## 7.7 SF.PACE

The content is available in the complete Security Target documentation.

## 7.8 SF.PINManager

The content is available in the complete Security Target documentation.

## 7.9 SF.Protection

The content is available in the complete Security Target documentation.

## 7.10 SF.SEManager

The content is available in the complete Security Target documentation.

## 7.11 SF.TrustedChannel

The content is available in the complete Security Target documentation.

## 7.12 SF.KeyManager

The content is available in the complete Security Target documentation.

# 8 Statement of compatibility concerning the composite ST

## 8.1 Separation of the platform TSF

### 8.1.1 Security functionalities

Table 8.1 confronts the relevant security functionalities of the platform with the security functionalities of the composite TOE to separate them. The security functionalities provided by the platform are summarized based on [ST_OS] (section 8).

**Table 8.1: Platform security functionalities used by the TOE**

| Platform security functionality | Usage by the TOE | Remarks |
|---|---|---|
| SF.JVCM: Java Card Virtual Machine | Yes | SF.JVCM constitutes the runtime framework for the Java Card applet being part of the TOE. |
| SF.CONFIG: Configuration Management | Yes | SF.CONFIG provides means to store Initialization Data and Pre-personalization Data by the TOE. |
| SF.OPEN: Card Content Management | Yes | SF.OPEN is used to load and instantiate the Java Card applet being part of the TOE. |
| SF.CRYPTO: Cryptographic Functionality | Yes | All cryptographic functionality of the TOE is based on the SF.CRYPTO security functionality of the platform. No cryptographic algorithms are implemented by the Java Card applet itself. |
| SF.RNG: Random Number Generator | Yes | Random number generation functionality of the TOE is based on the SF.RNG security functionality of the platform. |
| SF.DATA_STORAGE: Secure Data Storage | Yes | SF.DATA_STORAGE is used to store cryptographic keys by the TOE. |
| SF.PUF: User Data Protection using PUF | No | SF.PUF is not used by the TOE. |
| SF.OM: Java Object Management | Yes | SF.OM is used by the TOE as it provides Java objects management functionalities to the SF.JVCM. |
| SF.MM: Memory Management | Yes | SF.MM is used by the TOE as it provides memory management functionalities for Java Card objects. |
| SF.PIN: PIN Management | No | SF.PIN is not used by the TOE. |

**Table 8.1 (continued)**

| Platform security functionality | Usage by the TOE | Remarks |
|---|---|---|
| SF.BIO: Biometric Template Management | No | SF.BIO is not used by the TOE. |
| SF.PERS_MEM: Persistent Memory Management | Yes | SF.PERS_MEM is used by the TOE as it provides atomic write operations and transaction management for the Java Card Runtime Environment. |
| SF.EDC: Error Detection Code API | Yes | SF.EDC is used by the TOE as it provides an Java Card API to perform integrity checks Java Card arrays |
| SF.HW_EXC: Hardware Exception Handling | Yes | SF.HW_EXC is used by the TOE as provides software exception handler to react on unforeseen events captured by the hardware (hardware exceptions). |
| SF.PID: Platform Identification | Yes | SF.PID is used to identify unambiguously the TOE. |
| SF.SMG_NSC: No Side-Channel | Yes | SF. SMG_NSC is used by the TOE as it provides resistance to side-channel attacks for SF.CRYPTO. |
| SF.ACC_SBX: Secure Box | No | SF.ACC_SBX is not used by the TOE. |
| SF.MOD_INVOC: Module Invocation | No | SF. MOD_INVOC is not used by the TOE. |
| SF.SENS_RES: Sensitive Result | Yes | SF. SENS_RES is used by the TOE as it provides secure results storage functionality for Java Card sensitive methods. |
| SF.OSU: OS Update | No | SF.OSU is not used by the TOE. |
| SF.MOD_DEL: Module Deletion | No | SF. MOD_DEL is not used by the TOE. |

***Note***
*SF.MOD_DEL functionality of the platform is not used as only modules necessary to proper TOE operation are loaded during TOE production process. Particularly the Configuration Module allowing to change platform configuration is not present.*

### 8.1.2 Security functional requirements

The following composite SFRs are platform related:

- FCS_CKM.1/RSA
- FCS_CKM.1/ECDSA
- FCS_CKM.1/DH_PACE
- FCS_CKM.1/CA

- FCS_CKM.1/CAPK
- FCS_CKM.4
- FCS_COP.1/RSA
- FCS_COP.1/ECDSA
- FCS_COP.1/PACE_ENC
- FCS_COP.1/PACE_MAC
- FCS_COP.1/CA_ENC
- FCS_COP.1/CA_MAC
- FCS_RND.1
- FDP_UIT.1/DTBS
- FDP_RIP.1
- FDP_SDI.2/Persistent
- FDP_SDI.2/DTBS
- FDP_UCT.1/SCD
- FIA_UID.1/PACE
- FIA_UAU.1/PACE
- FIA_UAU.4/PACE
- FMT_SMR.1/PACE,
- FPT_EMS.1
- FPT_FLS.1
- FPT_PHP.1
- FPT_PHP.3
- FPT_TST.1
- FTP_ITC.1/VAD
- FTP_ITC.1/DTBS
- FTP_ITC.1/SVD
- FTP_ITC.1/SCD
- FTP_ITC.1/PACE

Other SFRs of the composite ST are not related directly to the platform.

The following SFRs of the platform contribute to the composite SFRs:

- FAU_ARP.1
- FCS_CKM.1.1
- FCS_CKM.1.1[ECDSA]
- FCS_CKM.1.1[RSA]
- FCS_CKM.4.1
- FCS_COP.1.1[AES]
- FCS_COP.1.1[AESMAC]
- FCS_COP.1.1[AES_CMAC]
- FCS_COP.1.1[DESMAC]
- FCS_COP.1.1[ECSignature]
- FCS_COP.1.1[RSASignaturePKCS1]
- FCS_COP.1.1[TripleDES]
- FCS_COP.1.1[ECDHPACEKeyAgreement]
- FCS_COP.1.1[ECDH_P1363]
- FCS_RNG.1
- FIA_UID.1[SC]

- FIA_UAU.1[SC]
- FIA_UAU.4[SC]
- FDP_RIP.1[APDU]
- FDP_RIP.1[GlobalArray_Refined]
- FDP_RIP.1[bArray]
- FDP_RIP.1[KEYS]
- FDP_RIP.1[TRANSIENT]
- FDP_SDI.2[DATA]
- FDP_SDI.2[SENSITIVE_RESULT]
- FDP_UIT.1[CCM]
- FMT_SMR.1[SD]
- FPT_EMSEC.1
- FPT_FLS.1
- FPT_PHP.3

The other platform SFRs are not used.

Mapping of the platform SFRs to the composite SFRs is provided in Table 8.2.

**Table 8.2: SFRs mapping**

| Composite SFR | Platform SFR | Comments |
|---|---|---|
| FCS_CKM.1/RSA | FCS_CKM.1.1[RSA]<br>FCS_RNG.1 | RSA key pair generation functionality of the platform is used by the TOE. |
| FCS_CKM.1/ECDSA | FCS_CKM.1.1[ECDSA]<br>FCS_RNG.1 | EC key pair generation functionality of the platform is used by the TOE. |
| FCS_CKM.1/DH_PACE | FCS_COP.1.1[ECDHPACEKeyAgreement] | ECDH key agreement is performed twice during each PACE establishment. |
| FCS_CKM.1/CA | FCS_COP.1.1[ECDH_P1363] | ECDH key agreement is performed during each CA establishment. |
| FCS_CKM.1/CAPK | FCS_CKM.1.1 | Static EC key pair for CA can be generated during personalization of the TOE. |

**Table 8.2 (continued)**

| Composite SFR | Platform SFR | Comments |
|---|---|---|
| FCS_CKM.4 | FCS_CKM.4.1 | Secure messaging session keys are destroyed if:<br>• secure messaging has failed,<br>• new secure messaging was established,<br>• they are not needed any more.<br>The Signatory may destroy the SCD by using dedicated APDU command. |
| FCS_COP.1/RSA | FCS_COP.1.1[RSASignaturePKCS1] | RSA digital signature creation functionality of the platform is used by the TOE. |
| FCS_COP.1/ECDSA | FCS_COP.1.1[ECSignature] | EC digital signature creation functionality of the platform is used by the TOE. |
| FCS_COP.1/PACE_ENC | FCS_COP.1.1[TripleDES]<br>FCS_COP.1.1[AES] | Triple DES and AES encryption functionality of the platform is used for secure messaging. |
| FCS_COP.1/PACE_MAC | FCS_COP.1.1[DESMAC]<br>FCS_COP.1.1[AESMAC]<br>FCS_COP.1.1[AES_CMAC] | Triple DES and AES MAC generation and verification functionality of the platform is used for secure messaging. |
| FCS_COP.1/CA_ENC | FCS_COP.1.1[TripleDES]<br>FCS_COP.1.1[AES] | Triple DES and AES encryption functionality of the platform is used for secure messaging. |
| FCS_COP.1/CA_MAC | FCS_COP.1.1[DESMAC]<br>FCS_COP.1.1[AESMAC]<br>FCS_COP.1.1[AES_CMAC] | Triple DES and AES MAC generation and verification functionality of the platform is used for secure messaging. |
| FCS_RND.1 | FCS_RNG.1 | The TOE uses random numbers generation functionality of the platform. |
| FDP_UIT.1/DTBS | FCS_COP.1.1[DESMAC]<br>FCS_COP.1.1[AESMAC]<br>FCS_COP.1.1[AES_CMAC] | Triple DES and AES MAC generation and verification functionality of the platform is used for secure messaging. |

**Table 8.2 (continued)**

| Composite SFR | Platform SFR | Comments |
|---|---|---|
| FDP_RIP.1 | FCS_CKM.4<br>FDP_RIP.1[APDU]<br>FDP_RIP.1[GlobalArray_Refined]<br>FDP_RIP.1[bArray]<br>FDP_RIP.1[KEYS]<br>FDP_RIP.1[TRANSIENT] | Cryptographic key destruction functionality is used by the TOE upon key objects de-allocation. Moreover the TOE overwrites the erased key object data with zeros.<br>APDU buffer object, byte array objects, cryptographic key objects and any transient objects allocation and de-allocation functionalities of the platform are used by the TOE to perform corresponding operations. |
| FDP_SDI.2/Persistent | FDP_SDI.2[DATA]<br>FDP_SDI.2[SENSITIVE_RESULT] | The TOE uses functionality provided by the platform to verify data integrity. |
| FDP_SDI.2/DTBS | FDP_SDI.2[DATA]<br>FDP_SDI.2[SENSITIVE_RESULT] | The TOE uses functionality provided by the platform to verify data integrity. |
| FDP_UCT.1/SCD | FCS_COP.1.1[TripleDES]<br>FCS_COP.1.1[AES] | Triple DES and AES encryption functionality of the platform is used for secure messaging. |
| FIA_UID.1/PACE | FIA_UID.1[SC] | In personalization phase the TOE uses Global Platform SCP03 authentication mechanism implementation provided by the platform. |
| FIA_UAU.1/PACE | FIA_UAU.1[SC] | In personalization phase the TOE uses Global Platform SCP03 authentication mechanism implementation provided by the platform. |
| FIA_UAU.4/PACE | FIA_UAU.4[SC] | In personalization phase the TOE uses Global Platform SCP03 authentication mechanism implementation provided by the platform. |
| FMT_SMR.1/PACE | FMT_SMR.1[SD] | In personalization phase the TOE uses Global Platform SCP03 implementation provided by the platform to maintain Personalization Agent role. |

**Table 8.2 (continued)**

| Composite SFR | Platform SFR | Comments |
|---|---|---|
| FPT_EMS.1 | FPT_EMSEC.1 | The TOE uses functionality provided by the platform. |
| FPT_FLS.1 | FPT_FLS.1<br>FAU_ARP.1 | The TOE uses functionality provided by the platform to preserve a secure state in case of security violation detection . |
| FPT_PHP.1 | FAU_ARP.1 | The TOE uses functionality provided by the platform to detect potential security violation. |
| FPT_PHP.3 | FPT_PHP.3<br>FAU_ARP.1 | The TOE uses functionality provided by the platform to resist physical manipulation and probing and to detect potential security violation. |
| FPT_TST.1 | FAU_ARP.1<br>FDP_SDI.2[DATA]<br>FDP_SDI.2[SENSITIVE_RESULT] | The TOE uses functionality provided by the platform to verify data integrity and detect potential security violation. |
| FTP_ITC.1/VAD | FDP_UIT.1[CCM] | SCP03 protocol implementation of the platform is used to establish secure channel during TOE personalization. |
| FTP_ITC.1/DTBS | FCS_COP.1.1[TripleDES]<br>FCS_COP.1.1[AES]<br>FCS_COP.1.1[DESMAC]<br>FCS_COP.1.1[AESMAC]<br>FCS_COP.1.1[AES_CMAC] | Triple DES and AES encryption functionality of the platform is used for secure messaging.<br>Triple DES and AES MAC generation and verification functionality of the platform is used for secure messaging. |
| FTP_ITC.1/SVD | FDP_UIT.1[CCM]<br>FCS_COP.1.1[TripleDES]<br>FCS_COP.1.1[AES]<br>FCS_COP.1.1[DESMAC]<br>FCS_COP.1.1[AESMAC]<br>FCS_COP.1.1[AES_CMAC] | SCP03 protocol implementation of the platform is used to establish secure channel during TOE personalization.<br>Triple DES and AES encryption functionality of the platform is used for secure messaging.<br>Triple DES and AES MAC generation and verification functionality of the platform is used for secure messaging. |

**Table 8.2 (continued)**

| Composite SFR | Platform SFR | Comments |
|---|---|---|
| FTP_ITC.1/SCD | FDP_UIT.1[CCM]<br>FCS_COP.1.1[TripleDES]<br>FCS_COP.1.1[AES]<br>FCS_COP.1.1[DESMAC]<br>FCS_COP.1.1[AESMAC]<br>FCS_COP.1.1[AES_CMAC] | SCP03 protocol implementation of the platform is used to establish secure channel during TOE personalization.<br>Triple DES and AES encryption functionality of the platform is used for secure messaging.<br>Triple DES and AES MAC generation and verification functionality of the platform is used for secure messaging. |
| FTP_ITC.1/PACE | FDP_UIT.1[CCM] | In personalization phase the TOE uses Global Platform SCP03 implementation provided by the platform. |

## 8.2 Compatibility between the composite ST and the platform ST

### 8.2.1 Threats

The following threats of the TOE can be mapped to the threats of the platform:

- T.SCD_Divulg,
- T.SCD_Derive,
- T.Hack_Phys,
- T.SigF_Misuse,
- T.Skimming,
- T.Eavesdropping,
- T.Abuse-Func,
- T.Information_Leakage,
- T.Phys-Tamper,
- T.Malfunction,
- T.Counterfeit.

Other threats of the TOE cannot be mapped to the threats of the platform.

The following threats of the platform are relevant for the composite ST:

- T.CONFID-APPLI-DATA[REFIED],
- T.CONFID-JCS-CODE,
- T.CONFID-JCS-DATA,
- T.SID.1,
- T.UNAUTHORIZED_CARD_MNGT,
- T.COM_EXPLOIT,
- T.PHYSICAL,
- T.LIFE_CYCLE,
- T.MODULE_REPLACEMENT,
- T.OS_OPERATE,
- T.RESOURCES,
- T.SID.2.

Other threats of the platform are not related to the composite ST.

Mapping between threats of the platform and threats of the TOE is given in Table 8.3.

**Table 8.3: Mapping threats of the platform and of the TOE**

| Platform ST threats \ Composite ST threats | T.SCD_Divulg | T.SCD_Derive | T.Hack_Phys | T.SigF_Misuse | T.Skimming | T.Eavesdroping | T.Abuse-Func | T.Information_Leakage | T.Phys-Tamper | T.Malfunction | T.Counterfeit |
|---|---|---|---|---|---|---|---|---|---|---|---|
| T.COM_EXPLOIT | X | X | | | X | X | | | | | X |
| T.CONFID-APLI-DATA[REFINED] | X | | | | | | | | | X | |
| T.CONFID-JCS-DATA | | | | | | | | | | X | |
| T.LIFE_CYCLE | | | | | | | X | | | | |
| T.PHYSICAL | X | | X | | | | | X | X | X | |
| T.SID.1 | X | | | X | | | | | | | |
| T.CONFID-JCS-CODE | X | | | | | | | | | | |
| T.UNAUTHORIZED_CARD_MNGT | X | | | X | | | | | | | |
| T.MODULE_REPLACEMENT | | | | | | | | | | X | |
| T.OS_OPERATE | X | X | | X | | | | | | X | |
| T.RESOURCES | | | | | | | | | | X | |
| T.SID.2 | | | | | | | | | | X | |

### 8.2.2 Organizational security policies

The following organizational security policies of the TOE can be mapped to the organizational security policies of the platform:

- P.Pre-Operational.

Other organizational security policies of the TOE cannot be mapped to the organizational security policies of the platform.

The following organizational security policies of the platform are relevant for the composite ST:

- OSP.PROCESS-TOE.

Other organizational security policies of the platform are not related to the composite ST.

Mapping between organizational security policies of the platform and organizational security policies of the TOE is given in Table 8.4.

**Table 8.4: Mapping organizational security policies of the platform and of the TOE**

| Platform ST OSP | Composite ST OSP | P.Pre-Operational |
|---|---|---|
| OSP.PROCESS-TOE | | X |

### 8.2.3 Assumptions

Assumptions of the platform cannot be mapped to assumptions of the TOE.

### 8.2.4 Security objectives of the TOE

The following security objectives of the TOE can be mapped to the security objectives of the platform:

- OT.Lifecycle_Security,
- OT.SCD_Unique,
- OT.SCD_SVD_Corresp,
- OT.SCD_Secrecy,
- OT.Sig_Secure,
- OT.Sigy_SigF,
- OT.EMSEC_Design,
- OT.Tamper_ID,
- OT.Tamper_Resistance,
- OT.TOE_TC_SVD_Exp,
- OT.TOE_TC_VAD_Imp,
- OT.TOE_TC_DTBS_Imp,
- OT.Data_Integrity,
- OT.Data_Authenticity,
- OT.Data_Confidentiality,
- OT.Prot_Inf_Leak,
- OT.Prot_Phys-Tamper,
- OT.Prot_Malfunction,
- OT.AC_Pers,
- OT.Chip_Auth_Proof.

Other security objectives of the TOE cannot be mapped to the security objectives of the platform.

The following security objectives of the platform are relevant for the composite ST:

- OT.ALARM,
- OT.CIPHER.
- OT.RND,
- OT.KEY-MNGT,
- OT.PIN-MNGT,

- OT.TRANSACTION,
- OT.COMM_AUTH,
- OT.COMM_INTEGRITY,
- OT.COMM_CONFIDENTIALITY,
- OT.GLOBAL_ARRAYS_CONFID,
- OT.GLOBAL_ARRAYS_INTEG,
- OT.DOMAIN-RIGHTS,
- OT.OPERATE,
- OT.RESOURCES,
- OT.SCP.IC,
- OT.SCP.RECOVERY
- OT.SCP.SUPPORT.

Other security objectives of the platform are not related to the composite ST.

Mapping between security objectives of the platform and security objectives of the TOE is given in Table 8.5.

**Table 8.5: Mapping security objectives of the platform and of the TOE**

| Platform ST SO | Composite ST SO | OT.Lifecycle_Security | OT.SCD_Unique | OT.SCD_SVD_Corresp | OT.SCD_Secrecy | OT.Sig_Secure | OT.Sigy_SigF | OT.EMSEC_Design | OT.Tamper_ID | OT.Tamper_Resistance | OT.TOE_TC_SVD_Exp | OT.TOE_TC_VAD_Imp | OT.TOE_TC_DTBS_Imp | OT.Data_Integrity | OT.Data_Authenticity | OT.Data_Confidentiality | OT.Prot_Inf_Leak | OT.Prot_Phys-Tamper | OT.Prot_Malfunction | OT.AC_Pers | OT.Chip_Auth_Proof |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| OT.ALARM | | X | | | X | | | | X | | X | X | X | | | | X | | X | | |
| OT.CIPHER | | | | | X | X | | X | | | X | X | X | X | X | | | | | | X |
| OT.RND | | | X | | | X | | X | | | | | | X | X | X | | | | | |
| OT.KEY-MNGT | | X | X | X | X | X | X | | | | | | | X | X | | | | | | X |
| OT.PIN-MNGT | | | | | | | X | | | | | | | | | | | | | | |
| OT.TRANSACTION | | X | | | | | | | | | | | | | | | | | | | |
| OT.COMM_AUTH | | | | | X | | | | | | X | X | X | | X | | | | | X | |
| OT.COMM_INTEGRITY | | | | | | | | | | | X | X | X | X | | | | | | X | |
| OT.COMM_CONFIDENTIALITY | | | | | | | | | | | X | X | X | | | | | | | | |
| OT.GLOBAL_ARRAYS_CONFID | | | | | | | | | | | | | | | | X | | | | | |
| OT.GLOBAL_ARRAYS_INTEG | | | | | | | | | | | | | | X | | | | | | | |
| OT.DOMAIN-RIGHTS | | | | | | | | | | | | | | | | | | | | X | |
| OT.OPERATE | | | | | | | | | | | | | | | | | | | X | | |
| OT.RESOURCES | | | | | | | | | | | | | | | | | | | X | | |
| OT.SCP.IC | | | | | X | | | X | X | X | | | | | | | X | X | | | |
| OT.SCP.RECOVERY | | | | | | | | | | X | | | | | | | | | X | | |
| OT.SCP.SUPPORT | | | | | X | | | X | X | X | | | | | | | | | | | |

### 8.2.5 Security objectives of the operational environment

The following security objectives of the operational environment of the TOE can be mapped to the security objectives of the operational environment of the platform:

- OE.SVD_Auth,
- OE.Dev_Prov_Service,
- OE. HID_TC_VAD_Exp,
- OE. SCA_TC_DTBS_Exp,
- OE.HID_TC_VAD_Exp,
- OE.SCA_TC_DTBS_Exp,
- OE.SCD_Secrecy,
- OE.SCD_SVD_Corresp.
- OE.Personalization.

Other security objectives of the operational environment of the TOE cannot be mapped to the security objectives of the operational environment of the platform.

The following security objectives of the operational environment of the platform are relevant for the composite ST:

- OE.USE_DIAG,
- OE.USE_KEYS,
- OE.PROCESS_SEC_IC.

Other security objectives of the operational environment of the platform are not related to the composite ST.

Mapping between security objectives of the operational environment of the platform and security objectives of the operational environment of the TOE is given in Table 8.6.

**Table 8.6: Mapping security objectives of the operational environment of the platform and of the TOE**

| **Platform ST SO of the operational environment** | **Composite ST SO of the operational environment** | OE.Dev_Prov_Service | OE. HID_TC_VAD_Exp | OE. SCA_TC_DTBS_Exp | OE.HID_TC_VAD_Exp | OE.SCA_TC_DTBS_Exp | OE.SCD_Secrecy | OE.SCD_SVD_Corresp | OE.SVD_Auth | OE.Personalization |
|---|---|---|---|---|---|---|---|---|---|---|
| OE.USE_DIAG | | | X | X | X | X | X | X | X | X |
| OE.USE_KEYS | | | | | | | X | X | X | |
| OE.PROCESS_SEC_IC | | X | | | | | | | | |

# Annex A Cryptographic Disclaimer

## A.1 Supported mechanisms, protocols and algorithms

Table A.1 presents the cryptographic mechanisms supported by the TOE and lists all cryptographic algorithms used by those mechanisms.

**Table A.1: Cryptographic functionality**

| | Purpose | Cryptographic mechanism | Standard of implementation | Key size in Bits | Standard of Application | Comments |
|---|---|---|---|---|---|---|
| 1 | Key Agreement / Authentication | PACEv2 (Generic Mapping), PACE-CAM (Chip Authentication Mapping), PACE common: ECDH, ECDH key generation, Nonce Encryption, Authentication token | [TR03110-1], [TR03110-3], [Doc9303], [TR03111] (sec. 4.3.2.1), [IEEE1363], [RFC5639], [FIPS186-4], [ANSI X9.63] | [MRZ] = 160 [Nonce] = 128 Brainpool EC: 224, 256, 320, 384, 512 NIST EC: 224, 256, 384, 521 3DES session key: 112 AES session keys: 128, 192, 256 | [TR03110-1], [TR03110-3], [Doc9303] | Related SFRs: - FCS_CKM.1/DH_PACE, - FCS_COP.1/PACE_ENC, - FIA_UAU.1/PACE, - FIA_UAU.5/PACE, JCOP 4.5 P71 platform used for: - ECDH, - ECDH key generation |
| 2 | Key Agreement / Authentication | Chip Authentication v1 ECDH, ECDH key generation | [TR03110-1], [Doc9303], [TR03111] (sec. 4.3.2.1), [IEEE1363], [RFC5639], [FIPS186-4], [ANSI X9.63] | 224, 256, 320, 384, 512, 521 | [Doc9303], [TR03110-1] | Related SFRs: - FCS_CKM.1/CA, - FIA_UAU.5/PACE, - FIA_UAU.6/EAC, - FIA_API.1 JCOP 4.5 P71 platform used for: - ECDH, - ECDH key generation |
| 3 | Digital Signature / Client-Server Authentication / Decryption | Key pair generation | RSA: [ST_OS], [PKCS#1] ECC FP: [ANSI X9.62], [ISO15946-2] | RSA CRT: 1024, 2048, 4096 ECC FP: 256, 320, 384, 512, 521 | [TR SIGN] | JCOP 4.5 P71 platform implementation See A.2 for list of supported curves. |
| 4 | Digital Signature / Client-Server Authentication | Digital signature generation | RSA: [ST_OS], RSASSA-PSS from [PKCS#1] ECC FP: [ANSI X9.62], [ISO15946-2] | RSA CRT: 1024, 2048, 4096 ECC FP: 256, 320, 384, 512, 521 | [TR SIGN] | JCOP 4.5 P71 platform implementation Hash is calculated outside the TOE. See A.2 for list of supported curves. |
| 5 | Decryption | Encryption key decipherment | RSA: RSASSA-PSS from [PKCS#1] | RSA CRT: 1024, 2048, 4096 | [TR SIGN] | JCOP 4.5 P71 platform implementation |
| 6 | Key agreement | Encryption key agreement | ECDSA: [ANSI X9.62], [ISO15946-2] | ECC FP: 256, 320, 384, 512, 521 | [TR SIGN] | JCOP 4.5 P71 platform implementation See A.2 for list of supported curves. |
| 7 | Authentication / Confidentiality / Integrity | Personalization Agent authentication using Global Platform SCP03. | [GlobalPlatform] also see line 17 | 128, 192, 256 | [GlobalPlatform] | Related SFRs: - FIA_UAU.4/PACE, - FIA_UAU.5/PACE, JCOP 4.5 P71 platform implementation Supported SCP03 modes: - C-DECRYPTION, R-ENCRYPTION, C-MAC and R-MAC - C-DECRYPTION, C-MAC and R-MAC - C-MAC and R-MAC, - C-DECRYPTION and C-MAC - C-MAC - No secure messaging |
| 8 | Confidentiality | 3DES in CBC mode for Secure Messaging after PACE / CA establishment | [TR03110-1], [TR03110-3], [Doc9303], [ISO10116] also see line 16 | 112 | [TR03110-1], [TR03110-3], [Doc9303] | Related SFRs: - FCS_COP.1/PACE_ENC, - FCS_COP.1/CA_ENC, - FDP_UCT.1/TRM |
| 9 | Confidentiality | AES in CBC mode for Secure Messaging after PACE / CA establishment | [TR03110-1], [TR03110-3], [Doc9303], [ISO10116] also see line 17 | 128, 192, 256 | [TR03110-1], [TR03110-3], [Doc9303] | Related SFRs: - FCS_COP.1/PACE_ENC, - FCS_COP.1/CA_ENC, - FDP_UCT.1/TRM |
| 10 | Integrity | 3DES in Retail-MAC mode for Secure Messaging after PACE / CA establishment | [TR03110-1], [TR03110-3], [Doc9303], also see line 16 and 19 | 112 | [TR03110-1], [TR03110-3], [Doc9303] | Related SFRs: - FCS_COP.1/PACE_MAC, - FCS_COP.1/CA_MAC, - FDP_UIT.1/TRM The first steps (C1…Cn) represent the DES with 56 Bits in CBC mode cipher. The last two steps (finalization of the Retail-MAC token and signature using 3DES) correspond to 3DES with 112 Bits of security in CBC mode. |
| 11 | Integrity | CMAC-AES for Secure Messaging after PACE / CA establishment | [TR03110-1], [TR03110-3], [Doc9303], also see line 17 | 128, 192, 256 | [TR03110-1], [TR03110-3], [Doc9303] | Related SFRs: - FCS_COP.1/PACE_MAC, - FCS_COP.1/CA_MAC, - FDP_UIT.1/TRM |

| | Purpose | Cryptographic mechanism | Standard of implementation | Key size in Bits | Standard of Application | Comments |
|---|---|---|---|---|---|---|
| 12 | Key Derivation | PACE,<br>Chip Authentication v1, Key derivation using SHA-1 and SHA-256 | [TR03110-1],<br>[TR03110-3],<br>[Doc9303],<br>[TR03111]<br>also see line 18 | 3DES: 112<br>AES: 128, 192, 256 | [TR03110-1],<br>[TR03110-3],<br>[Doc9303] | Related SFRs:<br>- FCS_CKM.1/DH_PACE,<br>- FCS_CKM.1/CA |
| 13 | Trusted Channel | Secure Messaging in ENC and MAC modes (PACE) | [TR03110-1],<br>[TR03110-3],<br>[Doc9303] | N/A | [TR03110-1],<br>[TR03110-3],<br>[Doc9303] | Related SFRs:<br>- FTP_ITC.1/PACE,<br>- FDP_UCT.1/TRM,<br>- FDP_UIT.1/TRM |
| 14 | Trusted Channel | Secure Messaging in ENC and MAC modes (CA after PACE) | [TR03110-1],<br>[TR03110-3],<br>[Doc9303] | N/A | [TR03110-1],<br>[TR03110-3],<br>[Doc9303] | Related SFRs:<br>- FCS_CKM.1/CA<br>- FDP_UCT.1/TRM,<br>- FDP_UIT.1/TRM |
| 15 | Cryptographic Primitive | Hybrid Physical True Random Number Generator (PTG.2) | [AIS20/AIS31] | N/A | [TR03110-1],<br>[TR03110-3],<br>[Doc9303] | Related SFRs:<br>- FCS_RND.1<br>JCOP 4.5 P71 platform used for:<br>- HRNG |
| 16 | Cryptographic Primitive | 3DES in mode CBC | [NIST800-67],<br>[ISO18033-3],<br>[NIST800-38A],<br>[NIST800-67],<br>[ISO9797-1] | 112 | [TR03110-1],<br>[TR03110-3],<br>[Doc9303] | JCOP 4.5 P71 platform implementation |
| 17 | Cryptographic Primitive | AES in mode CBC | [FIPS197],<br>[ISO18033-3],<br>[NIST800-38A],<br>[NIST800-38B]<br>[ISO9797-1] | 128, 192, 256 | [TR03110-1],<br>[TR03110-3],<br>[Doc9303] | JCOP 4.5 P71 platform implementation |
| 18 | Cryptographic Primitive | SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 | [FIPS180-4] | N/A | [TR03110-1],<br>[TR03110-3],<br>[Doc9303] | JCOP 4.5 P71 platform implementation |
| 19 | Cryptographic Primitive | DES in CBC mode | [FIPS46-3] | 56 | [TR03110-1],<br>[TR03110-3],<br>[Doc9303] | JCOP 4.5 P71 platform implementation |

## A.2 Supported elliptic curves

The TOE supports the following elliptic curves:

- NIST P-224 (secp224r1) [FIPS186-4],
- BrainpoolP224r1 [RFC5639],
- NIST P-256 (secp256r1) [FIPS186-4],
- BrainpoolP256r1 [RFC5639],
- BrainpoolP320r1 [RFC5639],
- NIST P-384 (secp384r1) [FIPS186-4],
- BrainpoolP384r1 [RFC5639],
- BrainpoolP512r1 [RFC5639],
- NIST P-521 (secp521r1) [FIPS186-4].

***Developer note:***
*Elliptic curves with 224 bit sizes are supported for the backward compatibility, nevertheless these curves **are not** recommended according to the [TR02102-1].*

# Annex B Bibliography

## B.1 Common Criteria documents


| | |
|---|---|
| [CC-Part1] | Common Criteria for Information Technology Security Evaluation,<br>Part 1: Introduction and general model; CCMB-2017-04-001;<br>Version 3.1, Revision 5, April 2017 |
| [CC-Part2] | Common Criteria for Information Technology Security Evaluation,<br>Part 2: Security functional components; CCMB-2017-04-002;<br>Version 3.1, Revision 5, April 2017 |
| [CC-Part3] | Common Criteria for Information Technology Security Evaluation,<br>Part 3: Security assurance requirements; CCMB-2017-04-003;<br>Version 3.1, Revision 5, April 2017 |
| [CC-CEM] | Common Methodology for Information Technology Security Evaluation, Evaluation methodology; CCMB-2017-04-004; Version 3.1, Revision 5, April 2017 |
| [CC-Smartcard] | Common Criteria – Supporting Document Guidance – Smartcard Evaluation,<br>CCDB-2010-03-001, Version 2.0, February 2010 |


## B.2 Protection profiles


| | |
|---|---|
| [EN 419211-2] | EN 419211-2:2013: Protection profiles for secure signature creation device - Part 2: Device with key generation (BSI-CC-PP-0059-2009-MA-02) |
| [EN 419211-3] | EN 419211-3:2013: Protection profiles for secure signature creation device - Part 3: Device with key import (BSI-CC-PP-0075-2012-MA-01) |
| [EN 419211-4] | EN 419211-4:2013: Protection profiles for secure signature creation device - Part 4: Extension for device with key generation and trusted channel to certificate generation application (BSI-CC-PP-0071-2012-MA-01) |
| [EN 419211-5] | EN 419211-5:2013: Protection profiles for secure signature creation device - Part 5: Extension for device with key generation and trusted channel to signature creation application (BSI-CC-PP-0072-2012-MA-01) |
| [EN 419211-6] | EN 419211-6:2014: Protection profiles for secure signature creation device - Part 6: Extension for device with key import and trusted channel to signature creation application (BSI-CC-PP-0076-2013-MA-01) |
| [PP_PACE] | Common Criteria Protection Profile Machine Readable Travel Document using Standard Inspection Procedure with PACE (PACE PP), BSI-CC-PP-0068-V2-2011-MA-01, Version 1.01, 22nd July 2014 |
| [PP_EAC] | Common Criteria Protection Profile Machine Readable Travel Document with „ICAO Application", Extended Access Control with PACE (EAC PP), BSI-CC-PP-0056-V2-2012, version 1.3.2, 5th December 2012 |
| [PP_IC] | Security IC Platform Protection Profile with Augmentation Packages; registered and certified by BSI (Bundesamt für Sicherheit in der Informationstechnik) under the reference BSI-CC-PP-0084-2014, Version 1.0, January 2014 |

## B.3 SSCD specifications

| | |
|---|---|
| [Directive] | DIRECTIVE 1999/93/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 13 December 1999 on a Community framework for electronic signatures |
| [EIDAS] | REGULATION (EU) No 910/2014 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC |
| [EN 419212-2] | EN 419212-2:2014: Application Interface for smart card used as Secure Signature Creation Device – Part 2: Additional services |
| [TR SIGN] | ANSSI/BSI Technical Report: Signature creation and administration for eIDAS token, Version 1.0 Release Candidate, 2015-01-19 |

## B.4 MRTD specifications

| | |
|---|---|
| [Doc9303] | [Doc9303-P9], [Doc9303-P10], [Doc9303-P11] or [Doc9303-P12] |
| [Doc9303-P9] | ICAO Doc 9303: Machine Readable Travel Documents – Part 9: Deployment of Biometric Identification and Electronic Storage of Data in MRTDs, 8th edition, 2021 |
| [Doc9303-P10] | ICAO Doc 9303: Machine Readable Travel Documents – Part 10: Logical Data Structure (LDS) for Storage of Biometrics and Other Data in the Contactless Integrated Circuit (IC), 8th edition 2021 |
| [Doc9303-P11] | ICAO Doc 9303: Machine Readable Travel Documents – Part 11: Security Mechanisms for MRTDs, 8th edition 2021 |
| [Doc9303-P12] | ICAO Doc 9303: Machine Readable Travel Documents – Part 12: Public Key Infrastructure for MRTDs, 8th edition 2021 |
| [TR03110-1] | BSI Technical Guideline TR-03110-1: Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token – Part 1: eMRTDs with BAC/PACEv2 and EACv1, Version 2.20, 26 February 2015 |
| [TR03110-3] | BSI Technical Guideline TR-03110-3: Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token – Part 3: Common Specifications, Version 2.21, 21 December 2016 |

## B.5 Platform documentation

| | |
|---|---|
| [JCOP_UM] | JCOP 4.5 P71, User manual for JCOP 4.5 P71, Rev.1.7, DocNo 615217, 2022-10-13, NXP Semiconductors |
| [ST_HW] | NXP Secure Smart Card Controller N7122 with IC Dedicated Software and Crypto Library (R1/R2/R3), Security Target Lite, Rev. 1.8, 2023-12-01, BSI-DSZ-CC-1149 |
| [ST_OS] | Security Target Lite for JCOP 4.5 P71, Rev. 2.6, 2023-12-11, NSCIB-CC-2300127-01 |

## B.6 Cryptographic standards

| | |
|---|---|
| [AIS20/AIS31] | Wolfgang Killmann (T-Systems GEI GmbH), Werner Schindler (BSI), A proposal for: Functionality classes for random number generators, Version 2.0, 18 September 2011 |
| [ANSI X9.62] | Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA), 2005 |
| [ANSI X9.63] | Public Key Cryptography for the Financial Services Industry: Key Agreement and Key Transport using Elliptic Curve Cryptography, November 20, 2001 |
| [FIPS46-3] | Federal Information Processing Standards Publication 46-3: Data Encryption Standard (DES). 25th October 1999 |

| | |
|---|---|
| [FIPS180-4] | Federal Information Processing Standards Publication 180-4: Secure Hash Standard (SHS), National Institute of Standards and Technology, March 2012 |
| [FIPS186-4] | Federal Information Processing Standards Publication 186-4: Digital Signature Standard. July 2013 |
| [FIPS197] | Federal Information Processing Standards Publication 197: Advanced Encryption Standard (AES), National Institute of Standards and Technology, November 26th 2001 |
| [IEEE1363] | IEEE 1363-2000: IEEE Standard Specifications for Public-Key Cryptography, 2002-08-06 |
| [ISO9797-1] | ISO/IEC 9797-1:1999: Information technology – Security techniques – Message Authentication Codes (MACs) – Part 1: Mechanisms using a block cipher |
| [ISO10116] | ISO/IEC 10116:2017: Information Technology – Security Techniques – Modes of operation for an n-bit block cipher |
| [ISO11770-3] | ISO/IEC 11770-3: Information technology – Security techniques – Key management – Part 3: Mechanisms using asymmetric techniques, 2008 |
| [ISO15946-1] | ISO/IEC 15946-1:2016: Information technology – Security techniques – Cryptographic techniques based on elliptic curves – Part 1: General |
| [ISO15946-2] | ISO/IEC 15946-2:2002: Information technology – Security techniques – Cryptographic techniques based on elliptic curves – Part 2: Digital signatures |
| [ISO15946-3] | ISO/IEC 15946-3:2002 Information technology – Security techniques – Cryptographic techniques based on elliptic curves – Part 3: Key establishment |
| [ISO18033-3] | ISO/IEC 18033-3:2010: Information Technology – Security Techniques – Encryption Algorithms – Part 3: Block Ciphers |
| [NIST800-38A] | NIST Special Publication 800-38A: Recommendation for Block Cipher Modes of Operation Methods and Techniques, National Institute of Standards and Technology, December 2001 |
| [NIST800-38B] | NIST Special Publication 800-38B: Recommendation for Block Cipher Modes of Operation – The CMAC Mode for Authentication, U.S. Department of Commerce, National Institute of Standards and Technology, May 2005 |
| [NIST800-67] | NIST Special Publication 800-67: Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, National Institute of Standards and Technology, Version 1.2, Revised July 2011 |
| [PKCS#1] | PKCS #1 v2.1: RSA Cryptography Standard, RSA Laboratories, 14 June 2002 |
| [PKCS#3] | PKCS #3: Diffie-Hellman Key-Agreement Standard, An RSA Laboratories Note, Version 1.4, Revised, November 1, 1993 |
| [RFC5639] | Elliptic Curve Cryptography (ECC) Brainpool Standard curves and Curve Generation. March 2010 |
| [TR02102-1] | BSI Technical Guideline TR-02102-1: Cryptographic Mechanisms: Recommendations and Key Lengths, 2017-01 |
| [TR03111] | BSI Technical Guideline TR-03111: Elliptic Curve Cryptography, Version 1.11, 17.04.2009 |


## B.7 Other


| | |
|---|---|
| [ASE MRTD] | PWPW SmartApp-ID 5.0 (MRTD configuration): Security Target, version 5.0.6.0, certification ID: NSCIB-2300119 |
| [ADV_ARC] | PWPW SmartApp-ID 5.0: Security architecture, exact version of the document is given in the certification report |
| [JCAPI3] | Java Card 3 Platform, Application Programming Interface, Classic Edition, Version 3.0.5, May 2015, published by Oracle |
| [GlobalPlatform] | GlobalPlatform Card Specification 2.3, GlobalPlatform Inc., October 2015 |
| [ISO_7816-2] | ISO/IEC 7816-2:2007: Identification cards – Integrated circuit cards – Part 2: Cards with contacts – Dimensions and location of the contacts |

# Annex C Acronyms

## C.1 Organizations

| | |
|---|---|
| BSI | Bundesamt für Sicherheit in der Informationstechnik |
| NXP | NXP Semiconductors |
| PWPW | Polska Wytwórnia Papierów Wartościowych S.A. |
| TÜVIT | TÜV Informationstechnik GmbH |

## C.2 Terms

| | |
|---|---|
| ADF | application dedicated file |
| AS | application software |
| BS | basic software |
| CA | chip authentication |
| CC | common criteria |
| ES | embedded software |
| CGA | certificate generation application |
| CSP | certification service provider |
| DTBS | data to be signed |
| DTBS/R | unique representation of DTBS |
| EAL | evaluation assurance level |
| HID | human interface device |
| IC | integrated circuit |
| JCVM | java card virtual machine |
| MRTD | machine readable travel document |
| OSP | organization security policy |
| PACE | password authenticated connection establishment |
| PIN | personal identification number |
| PUK | personal unblocking key |
| PP | protection profile |
| RAD | reference authentication data |
| SAR | security assurance requirements |
| SCA | signature creation application |
| SCD | signature creation data |
| SDO | signature data object |

| | |
|---|---|
| SO | security objectives |
| SSCD | secure signature creation device |
| ST | security target |
| SVD | signature verification data |
| TOE | target of evaluation |
| TSF | TOE security function |
| VAD | verification authentication data |

# Annex D Glossary

## D.1 Security evaluation terms

*Application Note*

Optional informative part of the protection profile containing sensitive supporting information that is considered relevant or useful for the construction, evaluation, or use of the TOE.

*Common Criteria*

A set of rules and procedures for evaluating the security properties of a product.

*Evaluation Assurance Level*

A set of assurance requirements for a product, its manufacturing process and its security evaluation specified by Common Criteria.

*Protection Profile*

A document specifying security requirements for a class of products that conforms in structure and content to rules specified by Common Criteria.

*Security Target*

A document specifying security requirements for a particular product that conforms in structure and content to rules specified by Common Criteria, which may be based on one or more protection profiles.

*Target of Evaluation*

Abstract reference in a document, such as a protection profile, for a particular product that meets specific security requirements.

*Target of Evaluation Security Functions*

Functions implemented by the TOE to meet the requirements specified for it in a protection profile or security target.

*TSF Data*

Data created by and for the TOE, that might affect the operation of the TOE.

*User Data*

Data created by and for the user, that does not affect the operation of the TSF.

## D.2 Smartcard terms

*Integrated Circuit*

Electronic component(s) designed to perform processing and/or memory functions (i.e. the hardware component containing the micro-controller and IC dedicated software).

A typical IC comprises: a processing unit, security components, I/O ports and volatile and non-volatile memories. It also includes any IC designer/manufacturer proprietary IC dedicated software, required for testing purposes. This IC dedicated software may be either IC embedded software (also known as IC firmware) or security-relevant parts of tests programs outside the IC. The IC may include any IC pre-personalization data.

*IC Dedicated Software*

IC proprietary software embedded in a smartcard IC (also known as IC firmware) and developed by the IC Developer. Such software is required for testing purposes (IC Dedicated Test Software) but may provide additional services to facilitate usage of the hardware and/or to provide additional services.

*IC Dedicated Test Software*

That part of the IC Dedicated Software (refer to above) which is used to test the device but which does not provide functionality during Phases 4 to 7.

Phases of the smartcard life-cycle are described in [CC-Smartcard], Figure 4.

*IC Dedicated Support Software*

That part of the IC Dedicated Software (refer to above) which provides functions in Phases 4 to 7. The usage of parts of the IC Dedicated Software might be restricted to certain phases.

Phases of the smartcard life-cycle are described in [CC-Smartcard], Figure 4.

*Identification Data*

Any data defined by the Integrated Circuit manufacturer and injected into the non-volatile memory by the Integrated Circuit manufacturer (Phase 3). These data are for instance used for traceability.

Phases of the smartcard life-cycle are described in [CC-Smartcard], Figure 4.

*Basic Software*

Smartcard embedded software in charge of generic functions of the Smartcard IC such as an operating system, general routines and interpreters.

*Application Software*

Smartcard embedded software (may be in ROM or loaded onto a platform in EEPROM or FLASH memory). It is software dedicated to the applications.

*Embedded Software*

Software embedded in a smartcard IC but not developed by the IC Designer. This comprises embedded software in charge of generic functions of the Smartcard IC, such as an operating system, general routines and interpreters (Smartcard Basic Software - BS) and embedded software dedicated to applications (Smartcard Application Software - AS). The Smartcard Embedded Software is designed in Phase 1 and embedded into the Smartcard IC in Phase 3 or in later phases of the smartcard product life-cycle.

Phases of the smartcard life-cycle are described in [CC-Smartcard], Figure 4.

*Smartcard Personalization*

Final process under the responsibility of the card issuer, through which a smartcard is to be configured, security parameters loaded and secret keys set. At the end of the personalization process, the smartcard is irreversibly set into “user mode”. Hence it becomes fully operational and can be delivered to the end user.

*IC Platform*

Usually refers to a smartcard component which may undergo an evaluation process as a complete Target of Evaluation (TOE) in itself, but which is not an end-user product (i.e. a smartcard component without any Application Software loaded).

*IC Initialization*

Process of writing Initialization Data to the IC.

*IC Initialization Data*

Any data defined by the IC Manufacturer and injected into the non-volatile memory during the manufacturing process. These data are for instance used for traceability and for IC identification.

*IC Pre-personalization*

Process performed at the IC manufacturer site, through which customer data can be loaded into the IC, prior to the IC being irreversibly set into “issuer mode”.

*IC Pre-personalization Data*

Any data supplied by the software developer that is injected into the non-volatile memory by the Integrated Circuits manufacturer (Phase 3). These data are for instance used for traceability and/or to secure shipment between phases.

Phases of the smartcard life-cycle are described in [CC-Smartcard], Figure 4.

*Smartcard Product*

A product corresponds to a fully operational smartcard, composed of both IC and complete ES, including application software as appropriate.

*IC Developer*

The entity which develops the integrated circuit, the IC Dedicated Software (firmware) and the guidance documentation.

*IC Manufacturer*

Institution (or its agent) responsible for the IC manufacturing, testing, and pre-personalization.

*ES Developer or AS Developer*

Institution (or its agent) responsible for the smartcard Embedded Software or Application Software development and the specification of IC pre-personalization requirements.

*Card Manufacturer*

The customer of the IC Manufacturer who receives the TOE during TOE Delivery. The Card Manufacturer includes all roles after TOE Delivery up to Phase 7. The Card Manufacturer has the following roles: (i) the Smartcard Product Manufacturer (Phase 5); (ii) the Personalizer (Phase 6). If the TOE is delivered after Phase 3 in the form of wafers or sawn wafers (dice) he also assumes the role of the IC Packaging Manufacturer (Phase 4). Usually, the Card Manufacturer is also the ES or AS developer.

Phases of the smartcard life-cycle are described in [CC-Smartcard], Figure 4.

*Card Issuer*

Customer for a product who is in charge of the issuance of the product to the smartcard holders (end users).

## D.3 SSCD terms

*Administrator*

A user that performs TOE initialization, TOE personalization, or other TOE administrative functions.

*Advanced electronic signature*

An electronic signature which meets the following requirements:

1. it is uniquely linked to the signatory,
2. it is capable of identifying the signatory,
3. it is created using means that the signatory can maintain under his sole control,
4. it is linked to the data to which it relates in such a manner that any subsequent change of the data is detectable.

*Authentication data*

Information used to verify the claimed identity of a user.

*Certificate*

An electronic attestation which links the SVD to a person and confirms the identity of that person.

*Certificate info*

Information associated with a SCD/SVD pair that may be stored in a secure creation device.

*Certificate generation application*

A collection of application elements which requests the SVD from the SSCD to generate a certificate obtaining data to be included in the certificate and to create a digital signature of the certificate.

*Certification service provider*

An entity or a legal or natural person who issues certificates or provides other services related to electronic signatures.

*Data to be signed*

All electronic data to be signed (including both user message and signature attributes).

*Data to be signed or its unique representation*

Data received by a secure signature creation device as input in a single signature creation operation.

The DTBS/R is either:

1. a hash-value of the data to be signed (DTBS), or
2. an intermediate hash-value of the first part of the DTBS and the remaining part of the DTBS, or
3. the DTBS.

*Legitimate user*

An user of a secure signature creation device who gains possession of it from an SSCD provisioning service provider and who may be authenticated by the SSCD as its signatory.

*Notified body*

An organizational entity designated by a member state of the European Union as responsible for accreditation and supervision of the evaluation process for products conforming to the claimed protection profiles and for determining admissible algorithms and algorithm parameters.

*Qualified certificate*

A certificate which meets the requirements laid down in Annex I of [Directive] and is provided by a CSP who fulfils the requirements laid down in Annex II of [Directive].

*Qualified electronic signature*

An advanced signature that has been created with SSCD with a key certified with a qualified certificate according to [Directive], article 5, paragraph 1.

*Reference authentication data*

Data persistently stored by the TOE for authentication of a user as authorized for a particular role.

*Secure signature creation device*

Configured software or hardware which is used to implement the SCD and which meets the requirements laid down in Annex III of the [Directive].

*Signatory*

A person who holds a SSCD and acts either on his own behalf or on behalf of the natural or legal person or entity he represents.

*Signature attributes*

Additional information that is signed together with the user message.

*Signature creation application*

The application complementing an SSCD with a user interface with the purpose to create an electronic signature.

The signature creation application is software consisting of a collection of application components configured to:

1. present the data to be signed (DTBS) for review by the signatory,
2. obtain prior to the signature process a decision by the signatory,
3. send a DTBS/R to the TOE if the signatory indicates by specific unambiguous input or action its intent to sign,
4. process the electronic signature generated by the SSCD as appropriate, e.g. as attachment to the DTBS.

*Signature creation data*

Unique data, such as codes or private cryptographic keys, which are used by the signatory to create an electronic signature.

*Signature creation system*

The complete system that creates an electronic signature. The signature creation system consists of the SCA and the SSCD.

*Signature verification data*

Data, such as codes or public cryptographic keys, which are used for the purpose of verifying an electronic signature.

*Signed data object*

The electronic data to which the electronic signature has been attached to or logically associated with as a method of authentication.

*SSCD provisioning service*

A service to prepare and provide a SSCD to a subscriber and to support the signatory with certification of generated keys and administrative functions of the SSCD.

*U*ser

Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

*Verification authentication data*

Data provided as input to a secure signature creation device for authentication by cognition or by data derived from user's biometric characteristics.

# Annex E Revision history

| VERSION | CHANGES |
|---|---|
| 5.0.1.0 | The following document was prepared based on PWPW SmartApp-ID 5.0 (SIGN configuration): Security Target, v5.0.7.0. |