Security Target Lite

ASapp-QSCD Applet

Common Criteria version 3.1 revision 4
Assurance Level EAL 4+

Version	1
Date	2018-06-27
Reference	TCLE180049
Classification	**PUBLIC**

Table of Contents

List of Tables

List of Figures

Notations

Numerical values

Numbers are printed in decimal, hexadecimal or binary notation. Hexadecimal values are indicated with a 'h' suffix as in XXh, where X is a hexadecimal digit from 0 to F. Decimal values have no suffix.

Example: the decimal value 179 may be noted as the hexadecimal value B3h.

Denoted text

The text added to provide details on how the TOE implementation fulfils some security requirements is written in *italics* and is preceded by the numbered tag "Application Note".

Refinements to the security requirements are denoted by the tag "Refinement" and are written in **bold** text.

Selections and *assignments* made by the Protection Profile authors are written in underlined text.

Selections and *assignments* made by the authors of this ST are written in **underlined bold** text.

Iterations are denoted by showing a slash "/", and the iteration indicator after the component indicator.

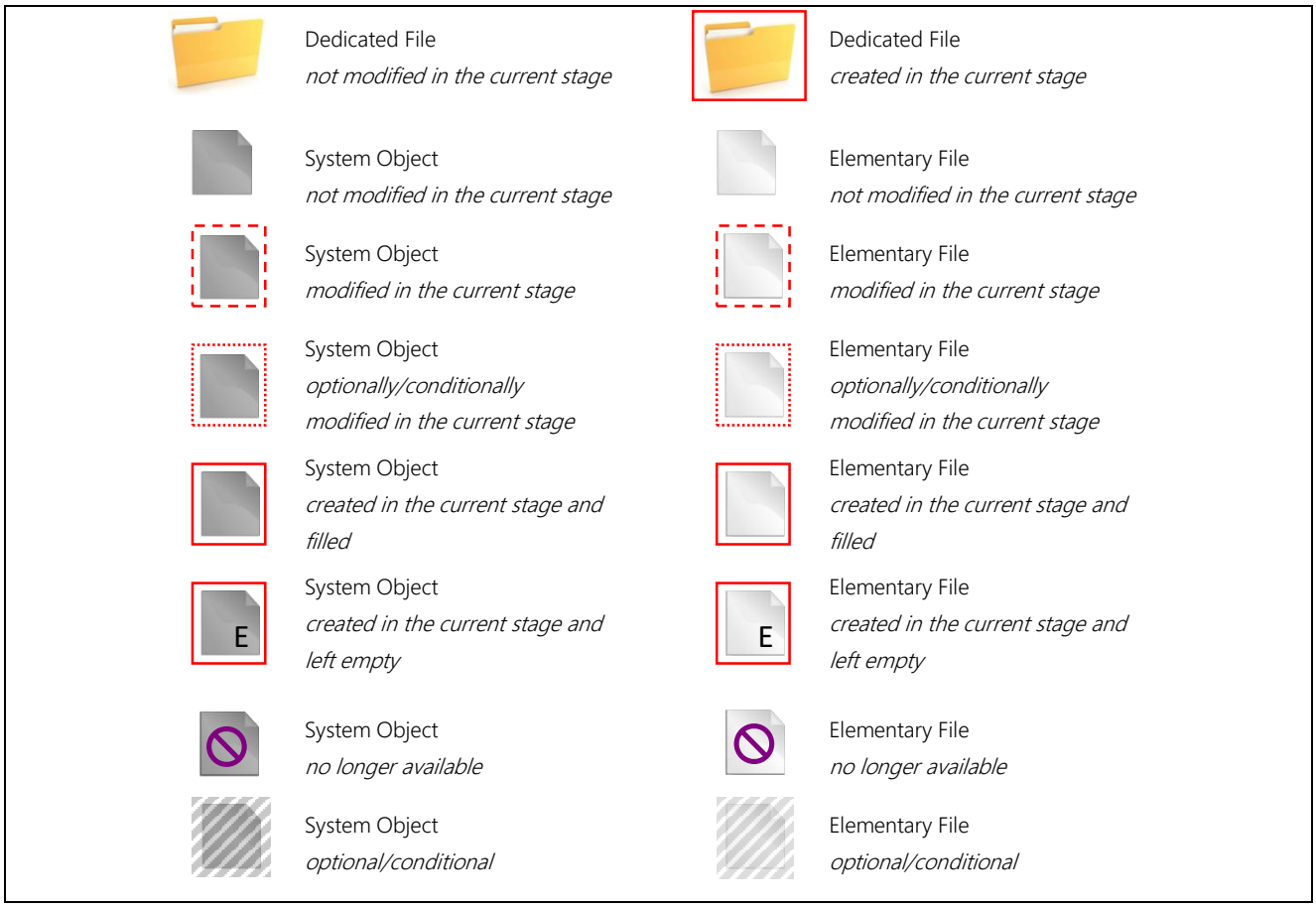
The original text of the selection and assignment components, as defined by the Common Criteria, is given by a footnote.

Key words

The words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY" and "OPTIONAL" are to be interpreted as described in RFC2119

Diagram legends

The following legend applies to the diagrams that illustrate the high-level objects present in the TOE persistent memory at the completion of the various stages of the TOE life cycle (cf. section 2.3):


Dedicated File
not modified in the current stage
Dedicated File
created in the current stage
System Object
not modified in the current stage
Elementary File
not modified in the current stage
System Object
modified in the current stage
Elementary File
modified in the current stage
System Object
optionally/conditionally
modified in the current stage
Elementary File
optionally/conditionally
modified in the current stage
System Object
created in the current stage and
filled
Elementary File
created in the current stage and
filled
E
System Object
created in the current stage and
left empty
E
Elementary File
created in the current stage and
left empty
System Object
no longer available
Elementary File
no longer available
System Object
optional/conditional
Elementary File
optional/conditional

1. Introduction

1.1 ST overview

This Security Target (ST) defines the security requirements, as well as the scope of the Common Criteria evaluation, for a Qualified Signature Creation Device (QSCD) Java Card™ applet, denominated *ASapp-QSCD*, compliant with the European Parliament Regulation No. 910/2014 [R13]. The ASapp-QSCD applet can optionally be configured as a PKCS #15 application [R28].

1.2 ST reference

Table 1-1 ST reference

Title	Security Target *ASapp-QSCD* Applet; Public Version
Version	1
Author	HID Global
Date	2018-06-27
Reference	TCLE180049

1.3 TOE reference

Table 1-2 TOE reference

Product Name	ASapp-QSCD
Product Version	1.0
TOE Identification Data	`41h 53h 61h 70h 70h 2Dh 51h 53h 43h 44h 5Fh 31h 5Fh 30h`
Evaluation Criteria	Common Criteria version 3.1 revision 4
Protection Profile	BSI-CC-PP-0059-2009-MA-01, BSI-CC-PP-0071-2012 , BSI-CC-PP-0072-2012
Evaluation Assurance Level	EAL 4 augmented by ALC_DVS.2 and AVA_VAN.5
Developer	HID Global
Evaluation Sponsor	HID Global
Evaluation Facility	CCLab Software Laboratory
Certification Body	OCSI
Certification ID	ASapp-QSCD (OSB)

The Target of Evaluation (TOE) is delivered as a chip ready for initialization. It is identified by the following string, which constitutes the TOE identifier:

ASapp-QSCD_1_0

(ASCII coding: `41h 53h 61h 70h 70h 2Dh 51h 53h 43h 44h 5Fh 31h 5Fh 30h`)

The last three bytes encode the applet version (1_0).

The ASCII encoding of the TOE identifier constitutes the TOE identification data, located in the persistent memory of the chip. Instructions for reading identification data are provided by the guidance documentation.

1.4 TOE overview

1.4.1 TOE type, usage, and major security features

The TOE is a combination of a certified integrated circuit chip P622J VB with its Java Card™ multi-application Chip Operating System (COS) named JCOP3 [R4] and an applet configured to securely create, use, and manage the Signature Creation Data (SCD). The QSCD 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 provides the following functions:

1. to generate Signature Creation Data (SCD) and the corresponding Signature Verification Data (SVD),
2. to export the SVD for certification to the CGA over a trusted channel,
3. to prove the identity as QSCD to external entities,
4. to, optionally, receive and store certificate info,
5. to switch the QSCD from a non-operational state to an operational state, and
6. if in an operational state, to create digital signatures for data with the following steps:
    a. select an SCD if multiple are present in the QSCD,
    b. authenticate the Signatory and determine its intent to sign,
    c. receive data to be signed or a unique representation thereof (DTBS/R) from the SCA over a trusted channel,

d. apply an appropriate cryptographic signature creation function to the DTBS/R using the selected SCD.

The TOE is prepared for the Signatory's use by:

1. generating at least one SCD/SVD pair, and
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 in the TOE.

After preparation, the SCD shall be in a non-operational state. Upon receiving a TOE, the Signatory shall verify its non-operational state and change the SCD state to operational.

After preparation, the intended legitimate user should be informed of the Signatory's Verification Authentication Data (VAD) required for use of the TOE in signing. The means of providing this information is expected to protect the confidentiality and the integrity of the corresponding Reference Authentication Data (RAD).

If the use of an SCD is no longer required, then it shall be destroyed.

1.4.2 Required non-TOE hardware/software/firmware

The TOE operates in the following operational environments:

- The preparation environment, where it interacts with a Certification Service Provider (CSP) through a Certificate Generation Application (CGA) to obtain a certificate for the Signature Verification Data (SVD) corresponding to the Signature Creation Data (SCD) generated by the TOE. The TOE exports the SVD through a trusted channel allowing the CGA to check its authenticity. The preparation environment interacts further with the TOE to personalize it with the initial value of the Reference Authentication Data (RAD).
- The signing environment, where it interacts with the signer through a Signature Creation Application (SCA) to sign data after authenticating the signer as its Signatory. The SCA provides the data to be signed or a unique representation thereof (DTBS/R) as input to the TOE signature creation function, and obtains the resulting digital signature. The TOE and the SCA communicate through a trusted channel to ensure the integrity of the DTBS/R.
- The management environment, where it interacts with the user to perform management operations, e.g. to reset a blocked RAD, after authenticating the user as its Signatory. A single device, e.g. a smart card terminal, may provide the required environment for management and signing.

Therefore, the use of the TOE requires any hardware, software, and firmware component of such operational environments, particularly a Certificate Generation Application (CGA) and a Signature Creation Application (SCA) supporting trusted channels with the TOE.

2. TOE description

2.1 TOE physical scope

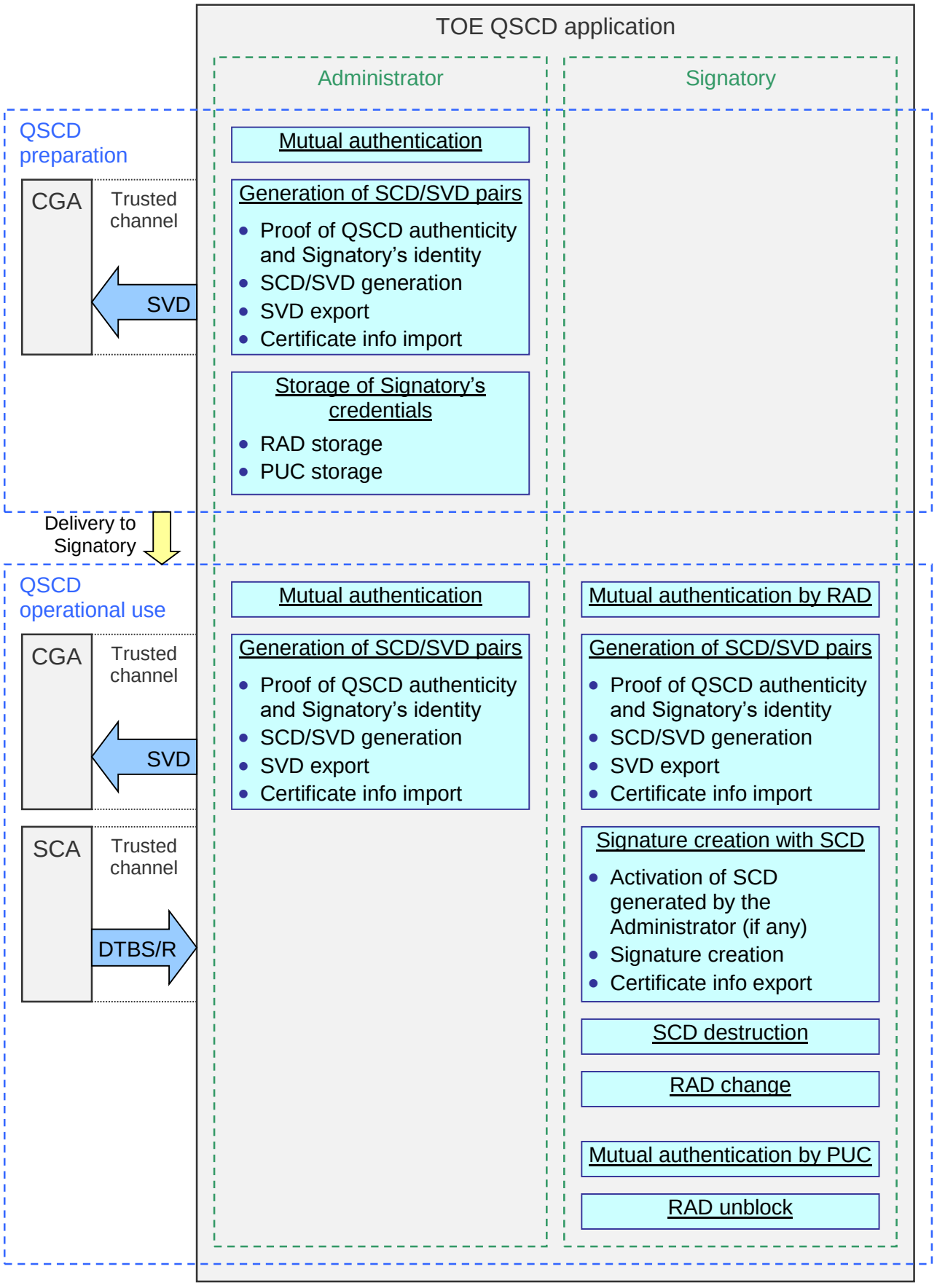
The TOE is comprised of the following parts:

- dual-interface chip NXP P6022J VB equipped with IC Dedicated Software (cf. Appendix A for more details);
- multi-application COS from NXP (JCOP3 [R20]), based on Java Card 3.0.4 [R15][R16][R17] and *GlobalPlatform* 2.2.1 [R12];
- a QSCD applet compliant with European Parliament Regulation No. 910/2014 [R13];
- guidance documentation [R1] [R2] [R3].

2.2 TOE logical scope

The QSCD application of the TOE supports the same QSCD life cycle phases, i.e. *QSCD preparation* and *QSCD operational use*, as well as the same QSCD roles, i.e. *Administrator* and *Signatory*, as those defined in the PPs [R9] [R10] [R11].
Figure 2-1 illustrates the operations supported by the QSCD application of the TOE, split according to the QSCD life cycle phases and the QSCD roles for which they are actually available.

Figure 2-1 QSCD applet operations split by QSCD life cycle phase and role



Here below, each of the main operations reported in Figure 2-1 is described in more detail.

2.2.1 Mutual authentication

As a precondition for gaining access to further operations, both the Administrator and the Signatory must perform a mutual authentication with respect to the QSCD application. The authentication procedure is comprised of the following two steps:

1. mutual authentication by means of a PACE authentication compliant with ICAO Doc 9303 [R18];
2. external authentication by means of the verification of a password over the trusted channel opened with PACE authentication.

All the algorithm combinations (i.e. key agreement algorithms, mapping algorithms, block ciphers) and the standardized domain parameters specified in ICAO Doc 9303 [R18] are supported for PACE authentication. All the encoding methods specified in PKCS #15 [R28] are supported as regards the passwords used in the password verification step.
The export of the SVD to the CGA upon key pair generation, as well as the import of the DTBS/R from the SCA upon signature creation, shall be executed over the trusted channel compliant with ICAO Doc 9303 [R18] opened by means of PACE authentication.
Table 2-1 identifies the credentials associated to either of the QSCD roles, through which they can perform their respective mutual authentication procedures.

Table 2-1 Mapping between QSCD roles and their credentials

QSCD roles	Credentials	Diversification
Administrator	• Administrator's PACE key • Administrator's password	*PACE key*: Same for each QSCD *Password*: Same/distinct for each QSCD
Signatory (for ordinary operations)	• Signatory's PACE key (derived from Signatory's password #1) • Signatory's password #2	*Password #1 (and PACE key)*: Distinct for each QSCD *Password #2*: Distinct for each QSCD
Signatory (for RAD unblock)	• Signatory's PACE key (derived from Signatory's password #1) • Signatory's password #3	*Password #1 (and PACE key)*: Distinct for each QSCD *Password #3*: Distinct for each QSCD

In accordance with Table 2-1, either of the QSCD roles shall perform mutual authentication as follows:

- The Administrator shall perform:
  1. PACE authentication using Administrator's PACE key;
  2. password verification using Administrator's password.
- The Signatory shall perform:
  1. PACE authentication using Signatory's PACE key, which shall be derived from the selected encoding of Signatory's password #1 via the key derivation function defined in ICAO Doc 9303 [R18];
  2. password verification using:
     - Signatory's password #2 for gaining access to ordinary operations;
     - Signatory's password #3 for gaining access to RAD unblock.

Table 2-2 identifies, for each of the Signatory's authentication secrets provided for by the PPs [R9] [R10] [R11], i.e. the RAD, the VAD, and the PUC[1], the Signatory's credentials of which it is comprised.

Table 2-2 Identification of RAD, VAD, and PUC in terms of Signatory's credentials

Signatory's secret	Signatory's credentials
RAD	• Signatory's password #1 (seed to derive Signatory's PACE key) • Signatory's password #2
VAD	Same as for the RAD
PUC	• Signatory's password #1 (seed to derive Signatory's PACE key) • Signatory's password #3

RAD change is implemented as the modification of Signatory's password #2 only, namely Signatory's password #1, and then Signatory's PACE key as well, cannot be changed.
Since Signatory's password #1 is used just as a seed for key derivation, its length is not constrained, whereas Administrator's password, Signatory's password #2, and Signatory's password #3 shall be 8 bytes long.

2.2.2 Generation of SCD/SVD pairs

The QSCD application supports the generation of multiple SCD/SVD pairs in the QSCD preparation phase on the part of the Administrator, as well as in the QSCD operational use

[1] The PPs implicitly provide for the existence of a PUC by allowing the support of RAD unblock.

phase on the part of both the Administrator and the Signatory. SCD keys are activated for signature creation upon their generation just in case they are generated by the Signatory, otherwise they are not active until the Signatory explicitly activates them. The import of certificate info from the CGA is supported as well. If configured as a PKCS #15 application, the QSCD application also supports the distinction between normal and trusted public keys and certificate info defined in PKCS #15 [R28].
SCD/SVD pair generation is only allowed after the authentication of the user in either of the QSCD roles (cf. section 2.2.1), and must be executed over the trusted channel opened via the PACE authentication step. This ensures the protection of SVD integrity upon export of the SVD to the CGA. The import of certificate info from the CGA must be executed over the same trusted channel.
Moreover, the QSCD application supports Client/Server Authentication compliant with IAS ECC specification [R14] as a means of performing an internal authentication of the QSCD to the CGA. This allows the CGA to verify the authenticity of the QSCD and the identity of its legitimate Signatory, as claimed by the certificate of the public key corresponding to the private key which the QSCD proves to know via Client/Server Authentication. The export of the generated SVD over the same trusted channel used for Client/Server Authentication provides the CGA with evidence that the exported SVD be actually linked to the legitimate Signatory, as well as to the SCD stored in the QSCD.
The QSCD application supports the generation of two-prime RSA key pairs compliant with PKCS #1 [R27] of 1024, 1280, 1536, or 2048 bits.
In accordance with IAS ECC specification [R14], the QSCD application supports signature creation algorithm RSASSA-PKCS1-v1_5 compliant with PKCS #1 [R27] for Client/Server Authentication, with keys of 1024, 1280, 1536, or 2048 bits. Following IAS ECC specification [R14], the hash function (i.e. SHA-1 [R25]) is expected to be applied by the terminal before the message to be signed is sent to the QSCD application.

2.2.3 Signature creation with SCD

In accordance with IAS ECC specification [R14], the QSCD application supports digital signature creation with signature creation algorithm RSASSA-PKCS1-v1_5 compliant with PKCS #1 [R27], hash algorithms SHA-1, SHA-256 compliant with FIPS PUB 180-4 [R25], and keys of 1024, 1280, 1536, or 2048 bits.
The signature creation function of the QSCD application can take all of the following types of data as input from the SCA:

- a hash value of the data to be signed;
- an intermediate hash value of a first part of the data to be signed, complemented with the remaining part of such data;
- the data to be signed themselves (provided their length is not larger than 64 bytes).

The export of public keys and certificate info to the SCA is supported as well.

Signature creation is only allowed after the authentication of the user in the Signatory role (cf. section 2.2.1), and must be executed over the trusted channel opened via the PACE authentication step. This guarantees the protection of DTBS/R integrity upon import of the DTBS/R from the SCA. The export of public keys, certificate info, and digital signatures to the SCA must be executed over the same trusted channel.

2.3 TOE life cycle

The TOE life cycle is described in terms of the following four life cycle phases, each divided in one or more steps:

1. Phase 1: Development, composed of
   Step 1) the development of the integrated circuit and of the multi-applications operating system Java Card 3 by the IC Manufacturer and
   Step 2) the development of the QSCD applet by the Embedded Software Developer;

2. Phase 2: Manufacturing, composed of
   Step 3) Loading the applet ,
   Step 4) the embedding of the chip in a substrate with an antenna. The antenna may be omitted if the IC contacts are exposed.
   Step 5) the Initialization and configuration

3. Phase 3: Personalization, comprising
   Step 6) Personalization of the e-Document for the holder

4. Phase 4: Operational Use, comprising
   Step 7) QSCD Preparation
   Step 8) QSCD Operational Use

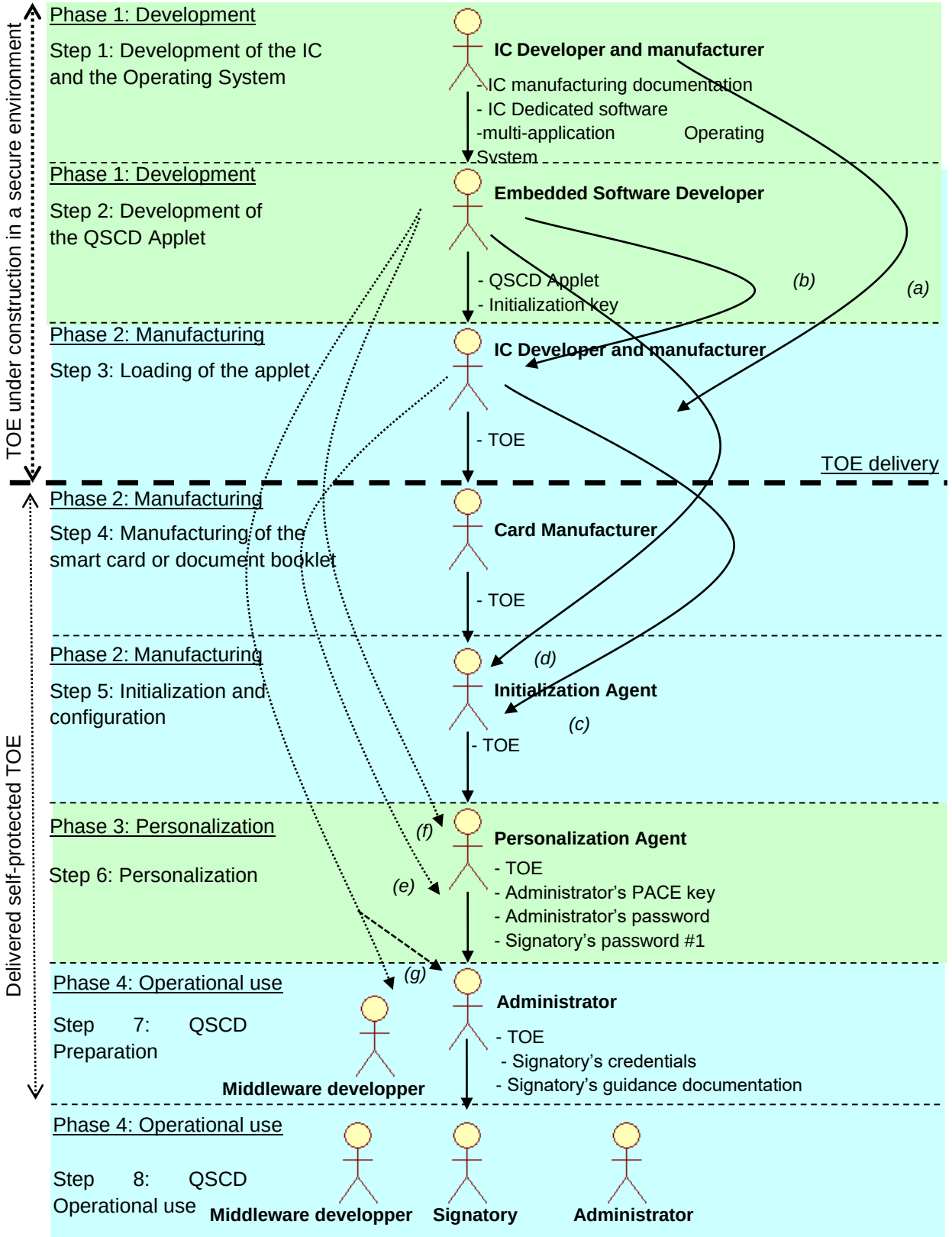
Application Note 1 *The entire Development phase, as well as Step 3 "*Loading the applet *" of the Manufacturing phase are the only phases covered by assurance as during these phases the TOE is under construction in a protected environment.*

Figure 2-2 shows life cycle phases and steps. The picture also identifies the actors involved in each life cycle step. Direct deliveries of items between actors are represented with continuous lines, while deliveries in which intermediate actors may be in charge of receiving the exchanged items and forwarding them to the subsequent actors are represented with dotted lines.
Deliveries of items not occurring between consecutive actors are just marked with letters in order to preserve the clarity of the diagram. A legend for these deliveries, which identifies the exchanged items for each of them, is provided in Table 2-3.

Table 2-3 Legend for deliveries not occurring between consecutive actors

Delivery	Delivered items
(a)	• SCP Key1 - DES - mutual authenticate for loading applet
(b)	QSCD Applet load file.
(c)	• SCP Key4 – DES – mutual authentication for initialization agent (initialization key)
(d)	• Initialization guidance
(e)	• SCP Key5 – DES – mutual authenticate for personalization agent (personalization key)
(f)	• Personalization guidance
(g)	• Operational user guidance

Figure 2-2 TOE life cycle



Detailed information about the operations available in each life cycle phase of the TOE is provided in the guidance documentation.
Table 2-4 identifies the roles in each phase of the TOE life cycle.

Table 2-4 Roles Identification

Phase	Role	Identification
1	IC Developer	NXP
1	IC Manufacturer	NXP
1	Applet Developer	HID Global
2	Card Manufacturer	The agent who is acting on behalf of the Issuing State or Organization to assemble the booklet or plastic card by embedding the TOE and antenna into the substrate.
2	Initialization Agent	The agent who is acting on behalf of the Issuing State or Organization to configure the applet.
3	Personalization Agent	User in charge of performing the personalization of the TOE, particularly of writing personalization data
4	Middleware Developer	Actor that implements the CGA and the SCA
4	Administrator	User in charge of performing QSCD preparation and other administrative operations of a QSCD.
4	Signatory	Legitimate user of a QSCD associated with it in the certificate of the SVD and who is authorized by the QSCD to operate the signature creation function ([R13], article 2.3).

Table 2-5 identifies, for each guidance document, the actors who are the intended recipients of that item.

Table 2-5 Identification of recipient actors for the guidance documentation of the TOE

Guidance document	Recipient actors
Initialization guidance	Initialization Agent
Personalization guidance	Personalization Agent
Operational user guidance	Middleware Developer Administrator

The phases and steps of the TOE life cycle are described in what follows. The names of the involved actors are emphasized using boldface.

2.3.1 Phase 1: Development

(Step 1) The **IC Developer** develops the integrated circuit, the IC Dedicated Software, the multi-application Operating System and the guidance documentation associated with these TOE components. The IC developer generates SCP Key5.

Finally, the following items are securely delivered to the **Embedded Software Developer**:

- the IC manufacturing documentation,
- the multi-application Operating System documentation

Step 2: Development of the Embedded Software

The **Embedded Software Developer** uses the guidance documentation for the integrated circuit and for relevant parts of the IC multi-application operating system and develops the Embedded Software (consisting of the QSCD applet) as well as its associated guidance documentation.

Finally, the following items are securely delivered to the **IC Manufacturer**:

- the QSCD applet

As regards TOE guidance documentation, either all documents are securely delivered to the **Initialization Agent**, or each document is securely delivered to the recipient actors as identified in Table 11-2.

2.3.2 Phase 2: Manufacturing

(Step 3) The **IC Manufacturer** produces the TOE integrated circuit, containing the multi-applications operating system, and creates in the IC persistent memory the high-level objects relevant for the QSCD applet depicted in Table 1-2. This also refers as loading of the applet.
Particularly, the initialization key is stored into the IC persistent memory.

Finally, the TOE is securely delivered to the **Card Manufacturer**.

Application Note 1 *The point of delivery of the TOE coincides with the completion of Step 3, i.e. with the delivery of the TOE from the IC Manufacturer to the Card Manufacturer.*

Figure 2-1 illustrates the high-level objects present in the IC persistent memory at the end of Step 3, i.e. at TOE delivery.

Figure 2-3 High-level persistent objects at the end of IC Manufacturing

(Step 4) The **Card Manufacturer** embeds the programmed IC into a plastic or paper substrate, optionally equipping it with an antenna (for ISO 14443 interface), and optionally exposing IC contacts (for ISO 7816-2 interface).

Finally, the TOE is securely delivered to the **Initialization Agent**.

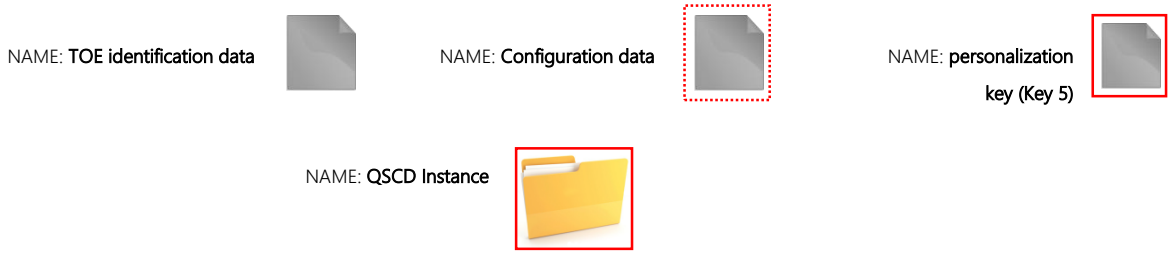
(Step 5) The **Initialization Agent** use the initialization key to mutual authentication with the TOE to instantiate QSCD applet.

Application Note 2 *During TOE initialization, the Initialization Agent establishes a trusted channel with the TOE through the initialization key. After initialization, personalization key will be set to the TOE. For further information, cf. the initialization guidance*[R1].

As regards TOE guidance documentation, if the **Initialization Agent** also received the documents intended for the subsequent actors, then either all of these documents are securely delivered to the **personalization Agent**, or each document is securely delivered to the recipient actors as identified in Table 11-2.

Figure 2-4 illustrates the high-level objects present in the IC persistent memory at the end of the initialization step of Phase 2: Manufacturing.

Figure 2-5 High-level persistent objects at the end of the initialization step



2.3.3 Phase 3: Personalization

Step 6: Personalization

The **Personalization Agent** establishes the identity of the Signatory to whom the TOE is to be assigned, generates the following credentials:

- Administrator's PACE key,
- Administrator's password,
- Signatory's password #1,

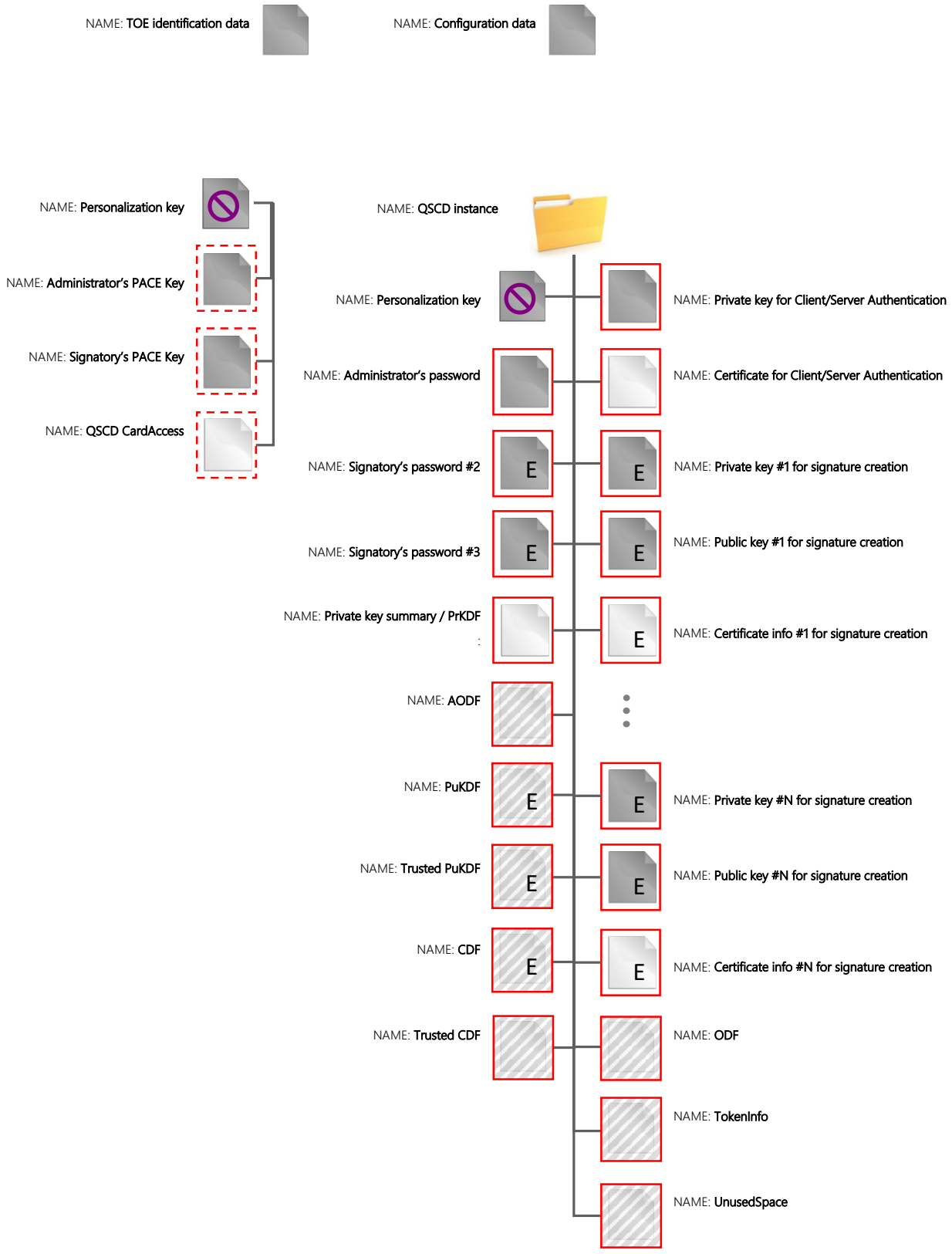
and derives Signatory's PACE key from Signatory's password #1.

Then, the **Personalization Agent** creates/modifies in the IC persistent memory the high-level objects relevant for the QSCD instance depicted accordingly in Figure 2-6.
Particularly:

- Administrator's PACE key object, Signatory's PACE key object, and Administrator's password object are filled with the generated credentials.
- The number of the empty private/public key objects and certificate info files being created, each associated with an unambiguous identifier, is equal to the maximum possible number of key pairs required for signature creation in the operational use phase. Although the key pairs are not generated yet, their lengths are fixed when the key objects are created and cannot be changed afterwards.
- If the QSCD application is configured as a PKCS #15 application [R28], the private key summary consists of a PrKDF file compliant with PKCS #15, and the TokenInfo, UnusedSpace, ODF, AODF, PuKDF, Trusted PuKDF, CDF, and Trusted CDF files are all present and compliant with PKCS #15 as well.

- Both a private key and a certificate attesting the identity of the Signatory are stored for Client/Server Authentication, and logical records are correspondingly added to the private key summary / PrKDF file and to the Trusted CDF file (if present).

Figure 2-6 High-level persistent objects relevant for the QSCD application at the end of TOE personalization



2.3.4 Phase 4: Operational use

(Step 7) QSCD preparation

Finally, the TOE is securely delivered to the **Administrator**, along with the following items:

- Administrator's credentials,
- Signatory's identification information,
- Signatory's password #1.

As regards TOE guidance documentation, if the **Personalization Agent** also received the operational user guidance, then this document is securely delivered to the recipient actors as identified in Table 2-4, i.e. the **Middleware Developer** and the **Administrator**.

(Step 8) QSCD operational use

The **Administrator** and the **Signatory** are allowed to modify in the IC persistent memory the high-level objects relevant for the QSCD application depicted accordingly in Figure 2-7. Particularly:

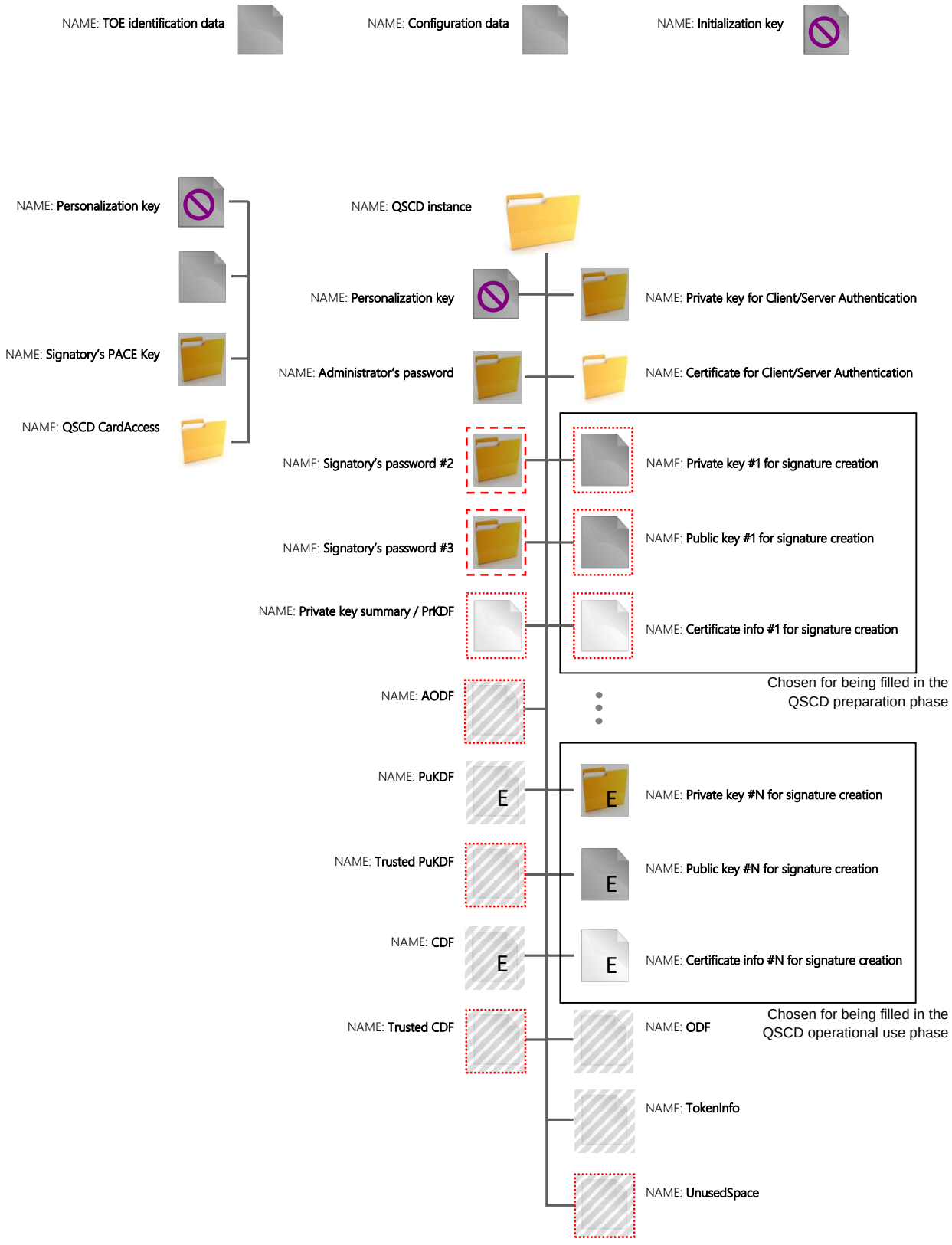
- The **Administrator** can generate one or more key pairs for signature creation using the CGA implemented by the **Middleware Developer**.
  In this case, as many private/public key objects created in the personalization phase are filled with the key pairs being generated. Moreover, as many logical records are added to the private key summary / PrKDF file and to the Trusted PuKDF file (if present), and deleted from the UnusedSpace file (if present).
- The **Administrator** shall fill one or more certificate info files for each generated key pair (if any). Moreover, as many logical records are added to the Trusted CDF file (if present), and deleted from the UnusedSpace file (if present).
- The **Signatory** can generate one or more key pairs for signature creation using the CGA implemented by the **Middleware Developer**.
  In this case, as many private/public key objects created in the personalization phase are filled with the key pairs being generated. Moreover, as many logical records are added to the private key summary / PrKDF file and to the PuKDF file (if present), and deleted from the UnusedSpace file (if present).
- The **Signatory** shall fill one or more certificate info files for each generated key pair (if any). Moreover, as many logical records are added to the CDF file (if present), and deleted from the UnusedSpace file (if present).

Furthermore, the **Signatory** can use the SCA implemented by the **Middleware Developer** to perform the following operations:

- activate signature creation for the private keys generated by the **Administrator**;
- create digital signatures using the available signature creation private keys;
- destroy signature creation private keys;
- change or unblock Signatory's password #2.

Figure 2-7 illustrates the high-level objects relevant for the QSCD application present in the IC persistent memory during QSCD operational use.

Figure 2-7 High-level persistent objects relevant for the QSCD application during QSCD operational use



3. Conformance claims

3.1 Common Criteria conformance claim

This security target claims conformance to Common Criteria (CC) version 3.1, revision 4 [R5] [R6] [R7] [R8].
Particularly:

- Security Functional Requirements (SFRs) are compliant with an extension of those defined in CC Part 2 [R6];
- Security Assurance Requirements (SARs) are compliant with those defined in CC Part 3 [R7].

The software part of the TOE runs on the chip NXP P6022J VB. This IC is certified against Common Criteria at Evaluation Assurance Level EAL5+ (cf. Appendix A).

3.2 Package conformance claim

This security target claims conformance to Evaluation Assurance Level EAL4, augmented with the following security assurance requirements defined in CC Part 3 [R7]:

- ALC_DVS.2 “Sufficiency of security measures”,
- AVA_VAN.5 “Advanced methodical vulnerability analysis”.

3.3 Protection Profile conformance claim

This security target claims strict conformance to the following Protection Profiles (PPs):

- Protection profiles for secure signature creation device – Part 2: Device with key generation, v2.0.1, January 2012, BSI-CC-PP-0059-2009-MA-01 [R9],
- Protection profiles for secure signature creation device – Part 4: Extension for device with key generation and trusted communication with certificate generation application, v1.0.1, November 2012, BSI-CC-PP-0071-2012 [R10],
- Protection profiles for secure signature creation device – Part 5: Extension for device with key generation and trusted communication with signature creation application, v1.0.1, November 2012, BSI-CC-PP-0072-2012 [R11].

3.4 Protection Profile conformance rationale

3.4.1 Security problem definition

Changes, additions, and deletions to the threats, OSPs, and assumptions with respect to the PPs (cf. section 3.3) are listed in Table 3-1, and Table 3-2.

Table 3-1 Changes, additions, and deletions to the OSPs with respect to the PPs

OSP	Difference	Rationale
P.Manufact	Addition	Added to specify the security policy to be enforced by the TOE in the manufacturing phase of its life cycle (cf. section 2.3.2).
P.Personalization	Addition	Added to specify the security policy to be enforced by the TOE in the personalization phase of its life cycle (cf. section 2.3.3).

Table 3-2 Changes, additions, and deletions to the assumptions with respect to the PPs

Assumption	Difference	Rationale
A.Process-Sec-IC	Addition	Added to cover the significant platform assumption A.Process-Sec-IC [R20] [R4] (cf. Appendix A).

3.4.2 Security objectives

Changes, additions, and deletions to the security objectives for the TOE and its operational environment with respect to the PPs (cf. section 3.3) are listed in Table 3-3 and Table 3-4.

Table 3-3 Changes, additions, and deletions to the security objectives for the TOE with respect to the PPs

Security objective	Difference	Rationale
OT.AC_Init	Addition	Added to specify the access control to be enforced by the TOE as regards the storage of TOE initialization data (cf. section 2.3.2).
OT.AC_Pers	Addition	Added to specify the access control to be enforced by the TOE as regards the storage of personalization data (cf. section 2.3.3).

Table 3-4 Changes, additions, and deletions to the security objectives for the operational environment with respect to the PPs

Security objective	Difference	Rationale
OE.Process-Sec-IC	Addition	Added to cover the significant platform security objective for the operational environment OE.Process-Sec-IC (cf. Appendix A).

3.4.3 Security functional requirements

Changes, additions, and deletions to the security functional requirements with respect to the PPs (cf. section 3.3) are listed in Table 3-5.

Table 3-5 Changes, additions, and deletions to the security functional requirements with respect to the PPs

SFR	Difference	Rationale
FIA_UAU.1	Change	Refined to remove user identification from the list of the actions allowed by the TOE before the user is authenticated (cf. Application Note 23).
FIA_AFL.1	Change	Refined to consider all the authentication mechanisms supported by the TOE in addition to those specified in the PPs (cf. Application Note 28).
FMT_SMR.1	Change	Refined to consider all the roles supported by the TOE in addition to those specified in the PPs (cf. Application Note 32).
FMT_MTD.1/Init	Addition	Added to specify the requirements to be enforced by the TOE as regards the management of initialization data (cf. section 2.3.2).
FMT_MTD.1/Pers	Addition	Added to specify the requirements to be enforced by the TOE as regards the management of personalization data (cf. section 2.3.3).
FTP_ITC.1/Pers	Addition	Added to account for the additional trusted channel supported by the TOE for the import of personalization data (cf. section 2.3.3).

3.4.4 Security assurance requirements

The minimum package of security assurance requirements allowed for conformance to the PPs (cf. section 3.3) is Evaluation Assurance Level EAL4 augmented with AVA_VAN.5.

As this security target claims conformance to Evaluation Assurance Level EAL4 augmented with ALC_DVS.2 and AVA_VAN.5 (cf. section 3.2), the aforesaid requirement is met.

4. Security problem definition

4.1 Assets, users, and threat agents

The Common Criteria define assets as entities that the owner of the TOE presumably places value upon. The term “asset” is used to describe the threats in the operational environment of the TOE.
The PPs [R9] [R10] [R11] share the same assets, users, and threat agents, reported here below.

Assets and objects:

1. SCD: private key used to perform an electronic signature operation. The confidentiality, integrity, and Signatory’s sole control over the use of the SCD must be maintained.
2. SVD: public key linked to the SCD and used to perform electronic signature verification. The integrity of the SVD must be maintained when it is exported.
3. DTBS and DTBS/R: set of data, or its representation, which the Signatory intends to sign. Their integrity and the unforgeability of the link to the Signatory provided by the electronic signature must be maintained.

Users and subjects acting for users:

1. User: end user of the TOE who can be identified as Administrator or Signatory. The subject S.User may act as S.Admin in the role R.Admin or as S.Sigy in the role R.Sigy.
2. Administrator: user who is in charge of performing QSCD preparation as well as other administrative functions. The subject S.Admin is acting in the role R.Admin for this user after successful authentication as Administrator.
3. Signatory: user who holds the TOE and uses it on their own behalf or on behalf of the natural or legal person or entity they represent. The subject S.Sigy is acting in the role R.Sigy for this user after successful authentication as Signatory.
4. Initialization Agent: user in charge of performing step 5, initialization, of TOE life cycle (cf. section 2.3.2.3.2), particularly of writing TOE initialization data. The subject S.Init is acting in the role R.Init for this user after successful authentication as Initialization Agent.
5. Personalization Agent: user in charge of performing step 6, personalization, of TOE life cycle (cf. section2.3.2.3.3), particularly of writing personalization data. The subject S.Pers is acting in the role R.Pers for this user after successful authentication as Personalization Agent.

Threat agents:

1. Attacker: Human or process acting on their behalf located outside the TOE. The main goal of the attacker is to access the SCD or to falsify the electronic signature. The attacker has a high attack potential and knows no secret.

4.2 Threats

The PPs [R9] [R10] [R11] share the same threats, reported here below.

4.2.1 T.SCD_Divulg

Storage, copy, and release of Signature Creation Data

An attacker stores or copies the SCD outside the TOE. An attacker can obtain the SCD during generation, storage, and use for signature creation in the TOE.

4.2.2 T.SCD_Derive

Derivation of Signature Creation Data

An attacker derives the SCD from publicly known data, such as SVD corresponding to the SCD or signatures created by means of the SCD or any other data exported outside the TOE, which is a threat against the secrecy of the SCD.

4.2.3 T.Hack_Phys

Physical attacks through TOE interfaces

An attacker interacts physically with the TOE to exploit vulnerabilities, resulting in arbitrary security compromises. This threat is directed against SCD, SVD, and DTBS.

4.2.4 T.SVD_Forgery

Forgery of Signature Verification Data

An attacker forges the SVD presented by the CSP to the CGA. This results in loss of SVD integrity in the certificate of the Signatory.

4.2.5 T.SigF_Misuse

Misuse of the signature creation function of the TOE

An attacker misuses the signature creation function of the TOE to create an SDO for data the Signatory has not decided to sign. The TOE is subject to deliberate attacks by experts possessing a high attack potential with advanced knowledge of security principles and concepts employed by the TOE.

4.2.6 T.DTBS_Forgery

Forgery of the DTBS/R

An attacker modifies the DTBS/R sent by the SCA. Thus the DTBS/R used by the TOE for signing does not match the DTBS that the Signatory intended to sign.

4.2.7 T.Sig_Forgery

Forgery of the electronic signature

An attacker forges an SDO, maybe using an electronic signature which has been created by the TOE, and the violation of the integrity of the SDO is not detectable by the Signatory or by third parties. The signature created by the TOE is subject to deliberate attacks by experts possessing a high attack potential with advanced knowledge of security principles and concepts employed by the TOE.

Here below is a further threat, added in this security target to those defined in the PPs.

4.3 Organizational Security Policies

The PPs [R9] [R10] [R11] share the same OSPs, reported here below.

4.3.1 P.CSP_QCert

Qualified certificates

The CSP uses a trustworthy CGA to generate a qualified certificate or non-qualified certificate ([R13], article 3, clause 14, and Annex I) for the SVD generated by the QSCD. The certificates contain at least the name of the Signatory and the SVD matching the SCD implemented in the TOE under sole control of the Signatory. The CSP ensures that the use of the TOE as QSCD is evident with signatures through the certificate or other publicly available information.

4.3.2 P.QSign

Qualified electronic signatures

The Signatory uses a Signature Creation System to sign data with an advanced electronic signature ([R13], article 2), which is a qualified electronic signature if it is based on a valid qualified certificate (according to [R13], Annex I). The DTBS are presented to the Signatory and sent by the SCA as DTBS/R to the QSCD. The QSCD creates the electronic signature with an SCD implemented in the QSCD that the Signatory maintains under their sole control, and is linked to the DTBS/R in such a manner that any subsequent change of the data is detectable.

4.3.3 P.Sigy_QSCD

TOE as Qualified Signature Creation Device

The TOE meets the requirements for a QSCD laid down in [R13], Annex III. This implies that the SCD is used for digital signature creation under sole control of the Signatory and the SCD can practically occur only once.

4.3.4 P.Sig_Non-Repud

Non-repudiation of signatures

The life cycle of the QSCD, the SCD, and the SVD shall be implemented in a way that the Signatory is not able to deny having signed data if the signature is successfully verified with the SVD contained in their unrevoked certificate.

Here below are further OSPs, added in this security target to those defined in the PPs.

4.3.5 P.Manufact

Manufacturing of the e-Document

The IC Manufacturer writes IC initialization data in step 3, IC manufacturing, of TOE life cycle, including the Initialization key (cf. section 2.3.2).

The IC Manufacturer writes initialization data in step 3, initialization, of TOE life cycle, including the key for the authentication of the initialization Agent (cf. section 2.3.2).

The initialization Agent writes initialization data in step 5 of TOE life cycle (cf. section 2.3.2), including the key for the authentication of the Personalization Agent.

Both the IC Manufacturer and the Initialization Agent act on behalf of the QSCD provisioning service.

4.3.6 P.Personalization

Personalization of the e-Document

The Personalization Agent writes personalization data in step 6, personalization, of TOE life cycle (cf. section 2.3.3), including the credentials for the authentication of the Administrator and the PACE key for the authentication of the Signatory.

The Personalization Agent acts on behalf of the QSCD provisioning service.

4.4 Assumptions

The PPs [R9] [R10] [R11] share the same assumptions, reported here below.

4.4.1 A.CGA

Trustworthy Certificate Generation Application

The CGA protects the authenticity of the Signatory's name or pseudonym and the SVD in the (qualified) certificate by an advanced electronic signature of the CSP.

4.4.2 A.SCA

Trustworthy Signature Creation Application

The Signatory uses only a trustworthy SCA. The SCA generates and sends the DTBS/R of the data that the Signatory wishes to sign in a form appropriate for signing by the TOE.

Here below is a further assumption, added in this security target to those defined in the PPs.

4.4.3 A.Process-Sec-IC

Protection during packaging, finishing, and personalization

It is assumed that security procedures are applied after the delivery of the TOE by the IC Manufacturer, up to delivery to the Signatory, to maintain the confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft, or unauthorized use).

This means that the life cycle steps after TOE delivery up to the completion of step 7, QSCD preparation (cf. section 2.3), are assumed to be protected appropriately.

Application Note 3 *The items to be protected are the following ones:*

- *the Embedded Software, including specifications, implementation, and related documentation;*
- *initialization data and personalization data, including specifications of formats and memory areas, as well as test-related data;*
- *user data, including user authentication data, and related documentation;*
- *material for software development support.*

5. Security objectives

5.1 Security objectives for the TOE

Here below are the security objectives for the TOE defined in PP Part 2 [R9].

5.1.1 OT.Lifecycle_Security

Life cycle security

The TOE shall detect flaws during the initialization, personalization, and operational usage. The TOE shall securely destroy the SCD on demand of the Signatory.

Application Note 4 *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 QSCD, e.g. after the (qualified) certificate for the corresponding SVD has expired.*

5.1.2 OT.SCD/SVD_Auth_Gen

Authorized SCD/SVD generation

The TOE shall provide security features to ensure that authorized users only may invoke the generation of the SCD and the SVD.

5.1.3 OT.SCD_Unique

Uniqueness of Signature Creation Data

The TOE shall ensure the cryptographic quality of an SCD/SVD pair that it creates as suitable for the advanced or qualified electronic signature. The SCD used for signature creation shall practically occur only once and shall not be reconstructible from the SVD. In that context “practically occur once” means that the probability of equal SCDs is negligible.

5.1.4 OT.SCD_SVD_Corresp

Correspondence between SVD and SCD

The TOE shall ensure the correspondence between the SVD and the SCD generated by the TOE. This includes unambiguous reference of a created SVD/SCD pair for export of the SVD and in creating an electronic signature with the SCD.

5.1.5 OT.SCD_Secrecy

Secrecy of Signature Creation Data

The secrecy of the SCD (used for signature creation) shall be reasonably assured against attacks with a high attack potential.

Application Note 5 *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.*

5.1.6 OT.Sig_Secure

Cryptographic security of the electronic signature

The TOE shall create digital signatures that cannot be forged without knowledge of the SCD, through robust encryption techniques. The SCD shall not be reconstructible using the digital signatures or any other data exportable from the TOE. The digital signatures shall be resistant against these attacks, even when executed with a high attack potential.

5.1.7 OT.Sigy_SigF

Signature creation function for the legitimate Signatory only

The TOE shall provide the digital signature creation function for the legitimate Signatory only and protects the SCD against the use of others. The TOE shall resist attacks with high attack potential.

5.1.8 OT.DTBS_Integrity_TOE

DTBS/R integrity inside the TOE

The TOE must not alter the DTBS/R. As by definition of the DTBS/R this may consist of the DTBS themselves, this objective does not conflict with a signature creation process where the TOE hashes the provided DTBS (in part or entirely) for signature creation.

5.1.9 OT.EMSEC_Design

Provision of physical emanations security

The TOE shall be designed and built in such a way as to control the production of intelligible emanations within specified limits.

5.1.10 OT.Tamper_ID

Tamper detection

The TOE shall provide system features that detect physical tampering of its components, and uses those features to limit security breaches.

5.1.11 OT.Tamper_Resistance

Tamper resistance

The TOE shall prevent or resist physical tampering with specified system devices and components.

Here below are the security objectives for the TOE defined in PP Part 4 [R10].

5.1.12 OT.TOE_QSCD_Auth

Authentication proof as QSCD

The TOE shall hold unique identity and authentication data as QSCD and provide security mechanisms to identify and to authenticate itself as QSCD.

5.1.13 OT.TOE_TC_SVD_Exp

TOE trusted channel for SVD export

The TOE shall provide a trusted channel to the CGA to protect the integrity of the SVD exported to the CGA. The TOE shall enable the CGA to detect alteration of the SVD exported by the TOE.

Here below are the security objectives for the TOE defined in PP Part 5 [R11].

5.1.14 OT.TOE_TC_VAD_Imp

TOE trusted channel for VAD import

The TOE shall provide a trusted channel for the protection of the confidentiality and integrity of the VAD received from the HID as needed by the authentication method employed.

Application Note 6 *This security objective for the TOE is partly covering OE.HID_VAD from PP Part 2 [R9]. While OE.HID_VAD in PP Part 2 requires only the operational environment to protect VAD, PP Part 5 [R11] 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, PP Part 5 partly re-assigns 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.

5.1.15 OT.TOE_TC_DTBS_Imp

TOE trusted channel for DTBS import

The TOE shall provide a trusted channel to the SCA to detect alteration of the DTBS/R received from the SCA. The TOE must not generate electronic signatures with the SCD for altered DTBS.

Application Note 7 *This security objective for the TOE is partly covering OE.DTBS_Protect from PP Part 2 [R9]. While OE.DTBS_Protect in PP Part 2 requires only the operational environment to protect DTBS, PP Part 5 [R11] 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, PP Part 5 partly re-assigns 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.*

Here below are further security objectives for the TOE, added in this security target to those defined in the PPs.

5.1.16 OT.AC_Init

Access control for the initialization of the e-Document

The TOE must ensure that initialization data, including the personalization key, can be written in step 5, initialization, of TOE life cycle (cf. section 2.3.2) by the authorized initialization Agent only.

5.1.17 OT.AC_Pers

Access control for the personalization of the e-Document

The TOE must ensure that personalization data, including Administrator's credentials and Signatory's PACE key, can be written in step 6, personalization, of TOE life cycle (cf. section 2.3.3) by the authorized Personalization Agent only.

5.2 Security objectives for the operational environment

PP Part 4 [R10] substitutes OE.QSCD_Prov_Service from PP Part 2 [R9] with OE.Dev_Prov_Service, and adds the security objectives for the operational environment OE.CGA_QSCD_Auth, OE.CGA_TC_SVD_Imp in order to address the additional method of use of SCD/SVD pair generation after delivery to the Signatory and outside a secure preparation environment.
PP Part 5 [R11] replaces OE.HID_VAD from PP Part 2 [R9] with OE.HID_TC_VAD_Exp, and OE.DTBS_Protect from PP Part 2 with OE.SCA_TC_DTBS_Exp.

Here below are the security objectives for the operational environment defined in PP Part 2 [R9].

5.2.1 OE.SVD_Auth

Authenticity of the SVD

The operational environment shall ensure the integrity of the SVD sent to the CGA of the CSP. The CGA verifies the correspondence between the SCD in the QSCD of the Signatory and the SVD in the qualified certificate.

5.2.2 OE.CGA_QCert

Generation of qualified certificates

The CGA shall generate a qualified certificate that includes (among others):

- the name of the Signatory controlling the TOE,
- the SVD matching the SCD stored in the TOE and being under sole control of the Signatory,
- the advanced signature of the CSP.

The CGA shall confirm with the generated qualified certificate that the SCD corresponding to the SVD is stored in the QSCD.

5.2.3 OE.DTBS_Intend

SCA sends data intended to be signed

The Signatory shall use a trustworthy SCA that:

- generates the DTBS/R of the data that has been presented as DTBS and which the Signatory intends to sign in a form which is appropriate for signing by the TOE,
- sends the DTBS/R to the TOE and enables verification of the integrity of the DTBS/R by the TOE,
- attaches the signature produced by the TOE to the data or provides it separately.

5.2.4 OE.Signatory

Security obligation of the Signatory

The Signatory shall check that the SCD stored in the QSCD received from the QSCD provisioning service is in non-operational state. The Signatory shall keep their VAD confidential.

Here below are the security objectives for the operational environment defined in PP Part 4 [R10].

5.2.5 OE.Dev_Prov_Service

Authentic QSCD provided by the QSCD provisioning service

The QSCD provisioning service handles authentic devices that implement the TOE, prepares the TOE for proof as QSCD to external entities, personalizes the TOE for the legitimate user as Signatory, links the identity of the TOE as QSCD with the identity of the legitimate user, and delivers the TOE to the Signatory.

Application Note 8 *This objective replaces OE.QSCD_Prov_Service from PP Part 2 [R9], which is possible as it does not imply any additional requirement for the operational environment when compared with OE.QSCD_Prov_Service (OE.Dev_Prov_Service is a subset of OE.QSCD_Prov_Service).*

5.2.6 OE.CGA_QSCD_Auth

Preparation of the TOE for QSCD authentication

The CSP shall check by means of the CGA whether the device presented for application of a (qualified) certificate holds unique identification as QSCD, successfully proved this identity as QSCD to the CGA, and whether this identity is linked to the legitimate holder of the device as applicant for the certificate.

5.2.7 OE.CGA_TC_SVD_Imp

CGA trusted channel for SVD import

The CGA shall detect alteration of the SVD imported from the TOE with the claimed identity of the QSCD.

Application Note 9 *The developer prepares the TOE for the delivery to the customer (i.e. the QSCD provisioning service) in the development phase, not addressed by security objectives for the operational environment. The QSCD provisioning service performs initialization and personalization as TOE for the legitimate user (i.e. the device holder). If the TOE is delivered to the device holder with SCD, the TOE is a QSCD. This situation is addressed by OE.QSCD_Prov_Service except for the additional initialization of the TOE for proof as QSCD and trusted channel to the CGA. If the TOE is delivered to the device holder without SCD, the TOE will be a QSCD only after generation of the first SCD/SVD pair. Because this SCD/SVD pair generation is performed by the Signatory in the operational use stage, the TOE provides additional security functionality addressed by OT.TOE_QSCD_Auth and OT.TOE_TC_SVD_Exp. But this security functionality must be initialized by the QSCD provisioning service as described in OE.Dev_Prov_Service. Therefore, PP Part 4 [R10] substitutes OE.QSCD_Prov_Service by OE.Dev_Prov_Service, allowing generation of the first SCD/SVD pair after delivery of the TOE to the device holder and requiring initialization of security functionality of the TOE. Nevertheless, the additional security functionality must be used by the operational environment as described in OE.CGA_QSCD_Auth and OE.CGA_TC_SVD_Imp. This approach does not weaken the security objectives and requirements for the TOE, but enforces more security functionalities of the TOE for additional methods of use. Therefore, it does not conflict with the CC conformance claim to PP Part 2 [R9].*

Here below are the security objectives for the operational environment defined in PP Part 5 [R11].

5.2.8 OE.HID_TC_VAD_Exp

HID trusted channel for VAD export

The HID provides the human interface for user authentication. The HID will ensure confidentiality and integrity of the VAD as needed by the authentication method employed, including export to the TOE by means of a trusted channel.

Application Note 10 *This security objective for the TOE is partly covering OE.HID_VAD from PP Part 2 [R9]. While OE.HID_VAD in PP Part 2 requires only the operational environment to protect VAD, this PP 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, PP Part 5 [R11] partly re-assigns 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.

5.2.9 OE.SCA_TC_DTBS_Exp

SCA trusted channel for DTBS export

The SCA provides a trusted channel to the TOE for the protection of the integrity of the DTBS, to ensure that the DTBS/R cannot be altered undetected in transit between the SCA and the TOE.

Application Note 11 *This security objective for the TOE is partly covering OE.DTBS_Protect from PP Part 2 [R9]. While OE.DTBS_Protect in PP Part 2 requires only the operational environment to protect DTBS, this PP 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, PP Part 5 [R11] partly re-assigns 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.*

Here below is a further security objective for the operational environment, added in this security target to those defined in the PPs.

5.2.10 OE.Process-Sec-IC

Protection during product manufacturing

Security procedures shall be applied after TOE delivery, up to delivery to the Signatory, to maintain the confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft, or unauthorized use).

This means that the life cycle steps after TOE delivery up to the completion of step 7, QSCD preparation (cf. section 2.3), must be protected appropriately.

Application Note 12 *The items to be protected are identified in Application Note 3.*

6. Security objectives rationale

6.1 Coverage of security objectives

Table 6-1 and Table 6-2 map the elements of the security problem definition to the security objectives for the TOE and for the operational environment, respectively. The rows are split according to the kind of element (threats, OSPs, assumptions), while the columns are split according to the source of the security objectives (PP Part 2 [R9], PP Part 4 [R10], PP Part 5 [R11], or this security target).

Table 6-1 Mapping of the security problem definition to the security objectives for the TOE

	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_QSCD_Auth	OT.TOE_TC_SVD_Exp	OT.TOE_TC_VAD_Imp	OT.TOE_TC_DTBS_Imp	OT.AC_Init	OT.AC_Pers
T.SCD_Divulg					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		
T.DTBS_Forgery								X							X		
T.Sig_Forgery			X			X											
P.CSP_QCert	X			X								X					
P.QSign						X	X										
P.Sigy_ P.Sigy_QSCD	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		
P.Manufact																X	X
P.Personalization		X					X										X
A.CGA																	
A.SCA																	
A.Process-Sec-IC																	

Table 6-2 Mapping of the security problem definition to the security objectives for the operational environment

	OE.SVD_Auth	OE.CGA_QCert	OE.DTBS_Intend	OE.Signatory	OE.Dev_Prov_Service	OE.CGA_QSCD_Auth	OE.CGA_TC_SVD_Imp	OE.HID_TC_VAD_Exp	OE.SCA_TC_DTBS_Exp	OE.Process-Sec-IC
T.SCD_Divulg										
T.SCD_Derive										
T.Hack_Phys										
T.SVD_Forgery	X						X			
T.SigF_Misuse			X	X				X	X	
T.DTBS_Forgery			X						X	
T.Sig_Forgery		X								
P.CSP_QCert		X				X				
P.QSign		X	X							
P.Sigy_QSCD					X	X	X			
P.Sig_Non-Repud	X	X	X	X	X	X	X	X	X	
P.Manufact					X					
P.Personalization					X					
A.CGA	X	X								
A.SCA			X							
A.Process-Sec-IC										X

6.2 Sufficiency of security objectives

In PP Part 4 [R10], the rationale for T.SCD_Divulg, T.SCD_Derive, T.Hack_Phys, T.SigF_Misuse, T.DTBS_Forgery, T.Sig_Forgery, P.QSign, A.CGA, and A.SCA remains unchanged as given in PP Part 2 [R9], section 7.3.2. The rationale how security objectives address threat T.SVD_Forgery and policies P.CSP_QCert, P.Sigy_QSCD, and P.Sig_Non-Repud changes as reported below.

In PP Part 5 [R11], the rationale for T.Hack_Phys, T.SCD_Divulg, T.SCD_Derive, T.Sig_Forgery, T.SVD_Forgery, P.CSP_QCert, P.QSign, P.Sigy_QSCD, A.CGA, and A.SCA remains unchanged as given in PP Part 2 [R9], section 7.3.2. The rationale how security objectives address threats T.DTBS_Forgery, T.SigF_Misuse and policy P.Sig_Non-Repud changes as reported below.

Here below is the rationale borrowed from PP Part 2 [R9].

T.SCD_Divulg *(Storage, copy, and release of Signature Creation Data)* addresses the threat against the legal validity of electronic signature due to storage and copying of SCD outside the TOE, as expressed in recital (18) of [R13]. This threat is countered by ***OT.SCD_Secrecy***, which assures the secrecy of the SCD used for signature creation.

T.SCD_Derive *(Derivation of Signature Creation Data)* deals with attacks on the SCD via publicly known data produced by the TOE, which are the SVD and the signatures created with the SCD. ***OT.SCD/SVD_Auth_Gen*** counters this threat by implementing cryptographically secure generation of the SCD/SVD pair. ***OT.Sig_Secure*** ensures cryptographically secure electronic signatures.

T.Hack_Phys *(Physical attacks through TOE interfaces)* deals with physical attacks exploiting physical vulnerabilities of the TOE. ***OT.SCD_Secrecy*** preserves the secrecy of the SCD. ***OT.EMSEC_Design*** counters physical attacks through the TOE interfaces and observation of TOE emanations. ***OT.Tamper_ID*** and ***OT.Tamper_Resistance*** counter the threat by detecting and by resisting tampering attacks.

T.Sig_Forgery *(Forgery of the electronic signature)* deals with non-detectable forgery of the electronic signature. ***OT.Sig_Secure***, ***OT.SCD_Unique***, and ***OE.CGA_QCert*** address this threat in general. ***OT.Sig_Secure*** ensures by means of robust cryptographic techniques that the signed data and the electronic signature are securely linked together. ***OT.SCD_Unique*** ensures that the same SCD cannot be generated more than once and the corresponding SVD cannot be included in another certificate by chance. ***OE.CGA_QCert*** prevents forgery of the certificate for the corresponding SVD, which would result in false verification decision concerning a forged signature.

P.QSign *(Qualified electronic signatures)* states that the TOE and the SCA may be employed to sign data with an advanced electronic signature, which is a qualified electronic signature if based on a valid qualified certificate. ***OT.Sigy_SigF*** ensures Signatory's sole control of the SCD by requiring the TOE to provide the signature creation function for the legitimate Signatory only and to protect the SCD against the use of others. ***OT.Sig_Secure*** ensures that the TOE creates electronic signatures which cannot be forged without knowledge of the SCD, through robust encryption techniques. ***OE.CGA_QCert*** addresses the requirement of qualified or non-qualified electronic certificates building a base for the electronic signature. ***OE.DTBS_Intend*** ensures that the SCA provides only those DTBS to the TOE, which the Signatory intends to sign.

A.CGA (*Trustworthy Certificate Generation Application)* establishes the protection of the authenticity of the Signatory's name and the SVD in the qualified certificate by the advanced signature of the CSP by means of the CGA. This is addressed by ***OE.CGA_QCert***, which

ensures the generation of qualified certificates, and by ***OE.SVD_Auth***, which ensures the protection of the integrity of the received SVD and the verification of the correspondence between the SVD and the SCD that is implemented by the QSCD of the Signatory.

A.SCA *(Trustworthy Signature Creation Application)* establishes the trustworthiness of the SCA with respect to generation of DTBS/R. This is addressed by ***OE.DTBS_Intend***, which ensures that the SCA generates the DTBS/R of the data that have been presented to the Signatory as DTBS and which the Signatory intends to sign in a form which is appropriate for being signed by the TOE.

Here below is the rationale borrowed from PP Part 4 [R10].

T.SVD_Forgery *(Forgery of Signature Verification Data)* deals with the forgery of the SVD exported by the TOE to the CGA for the generation of the certificate. The threat is addressed by ***OT.SCD_SVD_Corresp***, which ensures correspondence between SVD and SCD and unambiguous reference of the SVD/SCD pair for the SVD export and signature creation with the SCD, and by ***OE.SVD_Auth***, which ensures the integrity of the SVD exported by the TOE to the CGA and verification of the correspondence between the SCD in the QSCD of the Signatory and the SVD in the input provided to the certificate generation function of the CSP. Additionally, the threat is addressed by ***OT.TOE_TC_SVD_Exp***, which ensures that the TOE sends the SVD in a verifiable form through a trusted channel to the CGA, as well as by ***OE.CGA_TC_SVD_Imp***, which provides verification of SVD authenticity by the CGA.

P.CSP_QCert *(Qualified certificates)* states that the TOE and the SCA may be employed to sign data with (qualified) electronic signatures, as defined by [R13], article 5, paragraph 1. [R13], recital (15) refers to QSCDs to ensure the functionality of advanced signatures. ***OE.CGA_QCert*** addresses the requirement of qualified (or advanced) electronic signatures as being based on qualified (or non-qualified) certificates. According to ***OT.TOE_QSCD_Auth***, the copies of the TOE will hold unique identity and authentication data as QSCD and provide security mechanisms enabling the CGA to identify and to authenticate the TOE as QSCD. ***OE.CGA_QSCD_Auth*** ensures that the CSP checks the proof that the device is a QSCD presented by the applicant. ***OT.SCD_SVD_Corresp*** ensures that the SVD exported by the TOE to the CGA corresponds to the SCD stored in the TOE and used by the Signatory. ***OT.Lifecycle_Security*** ensures that the TOE detects flaws during initialization, personalization, and operational usage.

P.Sigy_QSCD *(TOE as Qualified Signature Creation Device)* requires the TOE to meet [R13], Annex III. Paragraph 1(a) of Annex III is ensured by ***OT.SCD_Unique***, requiring that the SCD used for signature creation can practically occur only once. ***OT.SCD_Secrecy***, ***OT.Sig_Secure***, ***OT.EMSEC_Design***, and ***OT.Tamper_Resistance*** address the secrecy of the SCD (cf. paragraph 1(a) of Annex III). ***OT.SCD_Secrecy*** and ***OT.Sig_Secure*** meet the requirement in paragraph 1(b) of Annex III by the requirement to ensure that the SCD cannot be derived from SVD, the electronic signatures, or any other data exported outside

the TOE. ***OT.Sigy_SigF*** meets the requirement in paragraph 1(c) of Annex III by the requirement to ensure that the TOE provides the signature creation function for the legitimate Signatory only and protects the SCD against the use of others. ***OT.DTBS_Integrity_TOE*** meets the requirement in paragraph 2 of Annex III as the TOE must not alter the DTBS/R. The usage of SCD under sole control of the Signatory is ensured by ***OT.Lifecycle_Security***, ***OT.SCD/SVD_Auth_Gen***, and ***OT.Sigy_SigF***.
OE.Dev_Prov_Service ensures that the legitimate user obtains a TOE sample as an authentic, initialized, and personalized TOE from a QSCD provisioning service through the TOE delivery procedure. If the TOE implements SCD generated under control of the QSCD provisioning service, the legitimate user receives the TOE as QSCD. If the TOE is delivered to the legitimate user without SCD, in the operational phase the user applies for the (qualified) certificate as the device holder and legitimate user of the TOE. The CSP will use the TOE security feature (addressed by security objectives ***OT.TOE_QSCD_Auth*** and ***OT.TOE_TC_SVD_Exp***) to check whether the device presented is a QSCD linked to the applicant, as required by ***OE.CGA_QSCD_Auth***, and whether the received SVD is sent by this QSCD, as required by ***OE.CGA_TC_SVD_Imp***. Thus, the obligation of the QSCD provisioning service for the first SCD/SVD pair is complemented in an appropriate way by the CSP for the SCD/SVD pair generated outside a secure preparation environment.

Here below is the rationale borrowed from PP Part 5 [R11].

T.SigF_Misuse *(Misuse of the signature creation function of the TOE)* addresses the threat of misuse of the TOE signature creation function to create an SDO by others than the Signatory, or to create an electronic signature on data for which the Signatory has not expressed the intent to sign, as required by paragraph 1(c) of [R13], Annex III. ***OT.Lifecycle_Security*** requires the TOE to detect flaws during initialization, personalization, and operational usage, including secure destruction of the SCD, which may be initiated by the Signatory. ***OT.Sigy_SigF*** ensures that the TOE provides the signature creation function for the legitimate Signatory only. ***OE.DTBS_Intend*** ensures that the SCA sends the DTBS/R only for data that the Signatory intends to sign. The combination of ***OT.TOE_TC_DTBS_Imp*** and ***OE.SCA_TC_DTBS_Exp*** counters the undetected manipulation of the DTBS during the transmission from the SCA to the TOE. ***OT.DTBS_Integrity_TOE*** prevents the DTBS/R from alteration inside the TOE. If the SCA provides a human interface for user authentication, ***OE.HID_TC_VAD_Exp*** requires the HID to protect the confidentiality and the integrity of the VAD as needed by the authentication method employed. The HID and the TOE will protect the VAD by a trusted channel between them according to ***OE.HID_TC_VAD_Exp*** and ***OT.TOE_TC_VAD_Imp***. ***OE.Signatory*** ensures that the Signatory checks that an SCD stored in the QSCD, when received from a QSCD provisioning service provider, is in non-operational state, i.e. the SCD cannot be used before the Signatory obtains control over the QSCD. ***OE.Signatory*** also ensures that the Signatory keeps their VAD confidential.

T.DTBS_Forgery *(Forgery of the DTBS/R)* addresses the threat arising from modifications of the DTBS/R sent to the TOE for signing, which then does not match the DTBS/R corresponding to the DTBS that the Signatory intends to sign. The threat is addressed by security objectives ***OT.TOE_TC_DTBS_Imp*** and ***OE.SCA_TC_DTBS_Exp***, which ensure that the DTBS/R is sent through a trusted channel and cannot be altered undetected in transit between the SCA and the TOE. The TOE counters internally this threat by means of ***OT.DTBS_Integrity_TOE***, ensuring the integrity of the DTBS/R inside the TOE. The TOE IT environment also addresses the threat by means of ***OE.DTBS_Intend***, which ensures that the trustworthy SCA generates the DTBS/R of the data that has been presented as DTBS and which the Signatory intends to sign in a form appropriate for signing by the TOE.

Here below is the rationale for policy P.Sig_Non-Repud, resulting from the combination of the rationales provided in PP Part 4 [R10] and PP Part 5 [R11].

P.Sig_Non-Repud *(Non-repudiation of signatures)* deals with the repudiation of signed data by the Signatory, although the electronic signature is successfully verified with the SVD contained in their certificate valid at the time of signature creation. This policy is implemented by the combination of the security objectives for the TOE and its operational environment, that ensure the aspects of Signatory's sole control over and responsibility for the electronic signatures generated with the TOE. ***OE.Dev_Prov_Service*** ensures that the Signatory uses an authentic TOE, initialized and personalized for the Signatory. ***OE.CGA_QCert*** ensures that the certificate allows to identify the Signatory and thus to link the SVD to the Signatory. ***OE.SVD_Auth*** and ***OE.CGA_QCert*** require the environment to ensure authenticity of the SVD as being exported by the TOE and used under sole control of the Signatory. ***OT.SCD_SVD_Corresp*** ensures that the SVD exported by the TOE corresponds to the SCD that is implemented in the TOE. ***OT.SCD_Unique*** ensures that the Signatory's SCD can practically occur just once.
OE.Signatory ensures that the Signatory checks that the SCD stored in the QSCD received from a QSCD provisioning service is in non-operational state (i.e. the SCD cannot be used before the Signatory obtains sole control over the QSCD). The TOE security feature addressed by security objectives ***OT.TOE_QSCD_Auth*** and ***OT.TOE_TC_SVD_Exp***, supported by ***OE.Dev_Prov_Service***, enables the verification whether the device presented by the applicant is a QSCD, as required by ***OE.CGA_QSCD_Auth***, and whether the received SVD is sent by the device holding the corresponding SCD, as required by ***OE.CGA_TC_SVD_Imp***. ***OT.Sigy_SigF*** ensures that only the Signatory may use the TOE for signature creation. As prerequisite, ***OE.Signatory*** ensures that the Signatory keeps their VAD confidential. The confidentiality of VAD is protected during the transmission between the HID and the TOE according to ***OE.HID_TC_VAD_Exp*** and ***OT.TOE_TC_VAD_Imp***. ***OE.DTBS_Intend***, ***OT.DTBS_Integrity_TOE***, ***OE.SCA_TC_DTBS_Exp***, and ***OT.TOE_TC_DTBS_Imp*** ensure that the TOE generates electronic signatures only for a DTBS/R that the Signatory has decided to sign as DTBS. The robust cryptographic techniques required by ***OT.Sig_Secure*** ensure that only this SCD may generate a valid electronic signature that can be successfully verified with the corresponding SVD used for

signature verification. Security objectives for the TOE ***OT.Lifecycle_Security***, ***OT.SCD_Secrecy***, ***OT.EMSEC_Design***, ***OT.Tamper_ID***, and ***OT.Tamper_Resistance*** protect the SCD against any compromise.

Here below is the rationale for the elements of the security problem definition added in this security target to those defined in the PPs.

P.Manufact *(Manufacturing of the e-Document)* requires the storage of the initialization data to be restricted to the initialization Agent, respectively, which is ensured by ***OT.AC_Init***. Furthermore, since access control requires user authentication, the secure storage of the initialization key, initialization and the personalization key prescribed by the policy is implied by ***OT.AC_Init*** and ***OT.AC_Pers***, respectively. Finally, the fact that the initialization Agent act on behalf of the QSCD provisioning service, as stated by the policy, is implied by ***OE.Dev_Prov_Service***, which puts the whole preparation of the TOE for its use as QSCD on the part of the Signatory under the responsibility of the QSCD provisioning service.

P.Personalization *(Personalization of the e-Document)* requires the storage of personalization data to be restricted to the Personalization Agent, which is ensured by ***OT.AC_Pers***. Furthermore, since access control requires user authentication, the secure storage of Administrator's credentials and Signatory's PACE key prescribed by the policy is implied by ***OT.SCD/SVD_Auth_Gen*** and ***OT.Sigy_SigF***. Finally, the fact that the Personalization Agent acts on behalf of the QSCD provisioning service, as stated by the policy, is implied by ***OE.Dev_Prov_Service***, which puts the whole preparation of the TOE for its use as QSCD on the part of the Signatory under the responsibility of the QSCD provisioning service.

A.Process-Sec_IC *(Protection during packaging, finishing, and personalization)* is covered by ***OE.Process-Sec-IC***, as this objective ensures the enforcement of the measures stated in the assumption.

7. Extended components definition

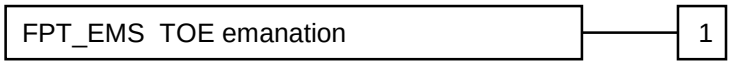
7.1 Definition of family FPT_EMS

The additional family FPT_EMS (TOE emanation) of class FPT (Protection of the TSF) is defined in PP Part 2 [R9] to describe the IT security functional requirements of the TOE.
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. Examples of such attacks are evaluation of TOE electromagnetic radiation, Simple Power Analysis (SPA), Differential Power Analysis (DPA), timing attacks, radio emanation, etc.
Family FPT_EMS describes the functional requirements for the limitation of intelligible emanations. This family belongs to class FPT because it is the class for TSF protection. Other families within class FPT do not cover TOE emanations.

FPT_EMS TOE emanation

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

Component levelling:



FPT_EMS.1 (TOE emanation) has two constituents:

- FPT_EMS.1.1 (Limit of emissions) requires not to emit intelligible emissions enabling access to TSF data or user data.
- FPT_EMS.1.2 (Interface emanation) requires not to emit interface emanation enabling access to TSF data or user data.

Management: FPT_EMS.1

There are no management activities foreseen.

Audit: FPT_EMS.1

There are no actions defined to be auditable.

FPT_EMS.1 TOE emanation

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_EMS.1.1:

The TOE shall not emit [assignment: *types of emissions*] in excess of [assignment: *specified limits*] enabling access to [assignment: *list of types of TSF data*] and [assignment: *list of types of user data*].

FPT_EMS.1.2:

The TSF shall ensure [assignment: *type of users*] are unable to use the following interface [assignment: *type of connection*] to gain access to [assignment: *list of types of TSF data*] and [assignment: *list of types of user data*].

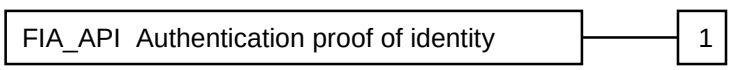
7.2 Definition of family FIA_API

The additional family FIA_API (Authentication proof of identity) of class FIA (Identification and authentication) is defined in PP Part 4 [R10] to describe the IT security functional requirements of the TOE.
This family describes the functional requirements for the proof of the claimed identity of the TOE by an external entity, whereas the other families of class FIA address the verification of the identity of an external entity.

FIA_API Authentication proof of identity

Family behaviour: This family defines functions provided by the TOE to prove its identity and to be verified by an external entity in the TOE IT environment.

Component levelling:



Management: FIA_API.1

The following actions could be considered for the management functions in FMT:

- Management of authentication information used to prove the claimed identity.

Audit: FIA_API.1

There are no actions defined to be auditable.

FIA_API.1 Authentication proof of identity

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_API.1.1:

The TSF shall provide [assignment: *authentication mechanism*] to prove the identity of the [assignment: *authorized user or role*].

8. Security functional requirements

Common Criteria allow several operations to be performed on functional requirements: *refinement*, *selection*, *assignment*, and *iteration* (cf. [R5], section 8.1). Each of these operations is used in this security target.

A (non-editorial) **refinement** operation is used to add details to a requirement, and thus further restricts a requirement (as regards the distinction between editorial and non-editorial refinements, cf. [R5], section 8.1.4). Non-editorial refinements of security requirements are written in **bold** text for additions or changes, in ~~strikethrough~~ text for deletions, and those made by the authors of this security target on the requirements borrowed from the PPs are signalled by an application note.

A **selection** operation is used to select one or more options provided by the CC in stating a requirement. A selection that has been made in the PPs is indicated as underlined text, and the original text of the component is given by a footnote. Selections filled in by the authors of this security target are written in **underlined bold** text, and the original text of the component is given by a footnote.

An **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. An assignment that that has been made in the PPs is indicated as underlined text, and the original text of the component is given by a footnote. Assignments filled in by the authors of this security target are written in **underlined bold** text, and the original text of the component is given by a footnote.

An **iteration** operation is used when a component is repeated with varying operations. Iteration is denoted by showing a slash “/” and the iteration indicator after the component identifier.

Table 8-1 maps each SFR stated in this security target to the PPs in which it is defined, if any. Particularly, SFR FIA_UAU.1 is mapped to both PP Part 4 [R10] and PP Part 5 [R11] since both PPs extend the formulation of the SFR given in PP Part 2 [R9]. Therefore, the formulation of the SFR given in this security target results from the combination of those given in PP Part 4 and PP Part 5.

Table 8-1 Mapping of the security functional requirements to the PPs

Security functional requirement	PP Part 2	PP Part 4	PP Part 5
FCS_CKM.1	X		
FCS_CKM.4	X		

Security functional requirement	PP Part 2	PP Part 4	PP Part 5
FCS_COP.1	X		
FDP_ACC.1/SCD/SVD_Generation	X		
FDP_ACF.1/SCD/SVD_Generation	X		
FDP_ACC.1/SVD_Transfer	X		
FDP_ACF.1/SVD_Transfer	X		
FDP_ACC.1/Signature creation	X		
FDP_ACF.1/Signature creation	X		
FDP_RIP.1	X		
FDP_SDI.2/Persistent	X		
FDP_SDI.2/DTBS	X		
FDP_DAU.2/SVD		X	
FDP_UIT.1/DTBS			X
FIA_UID.1	X		
FIA_UAU.1		X	X
FIA_AFL.1	X		
FIA_API.1		X	
FMT_SMR.1	X		
FMT_SMF.1	X		
FMT_MOF.1	X		
FMT_MSA.1/Admin	X		
FMT_MSA.1/Signatory	X		
FMT_MSA.2	X		
FMT_MSA.3	X		
FMT_MSA.4	X		
FMT_MTD.1/Admin	X		
FMT_MTD.1/Signatory	X		
FMT_MTD.1/Init			
FMT_MTD.1/Pers			
FPT_EMS.1	X		
FPT_FLS.1	X		
FPT_PHP.1	X		
FPT_PHP.3	X		

Security functional requirement	PP Part 2	PP Part 4	PP Part 5
FPT_TST.1	X		
FTP_ITC.1/SVD		X	
FTP_ITC.1/VAD			X
FTP_ITC.1/DTBS			X
FTP_ITC.1/Pers			

8.1 Class FCS: Cryptographic support

8.1.1 FCS_CKM.1

Cryptographic key generation

Hierarchical to: No other components.

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

FCS_CKM.1.1:

Statement applying to platform P6022J VB*:*

The TSF shall generate **SCD/SVD pairs** in accordance with a specified cryptographic key generation algorithm **two-prime RSA**[2] and specified cryptographic key sizes **1024, 1280, 1536, 2048 bits**[3] that meet the following: **PKCS #1 [R27]**[4].

Application Note 13 *The refinement in the element FCS_CKM.1.1 substitutes "cryptographic keys" with "SCD/SVD pairs" because it clearly addresses the SCD/SVD key generation.*

Application Note 14 *The TOE uses the RSA library provided by the underlying IC for key generation (cf. Appendix A). The reason why distinct statements of SFR FCS_CKM.1 apply to platforms* P6022J VB *is that the certification report of the former [R4] covers RSA key*

[2] [assignment: *cryptographic key generation algorithm*]
[3] [assignment: *cryptographic key sizes*]
[4] [assignment: *list of standards*]

lengths in the range 1024-4096 bits, whereas the certification report of the latter[R4] covers RSA key lengths in the range 1976-4096 bits only..

8.1.2 FCS_CKM.4

Cryptographic key destruction

Hierarchical to: No other components.

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

FCS_CKM.4.1:

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method **overwriting with zeros**[5] that meets the following: **none**[6].

8.1.3 FCS_COP.1

Cryptographic operation

Hierarchical to: No other components.

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

FCS_COP.1.1:

Statement applying to platform P6022J VB*:*

The TSF shall perform digital signature creation[7] in accordance with a specified cryptographic algorithm **RSASSA-PKCS1-v1_5 with SHA-1, SHA-256**[8] and cryptographic key sizes **1024,**

[5] [assignment: *cryptographic key destruction method*]
[6] [assignment: *list of standards*]
[7] [assignment: *list of cryptographic operations*]
[8] [assignment: *cryptographic algorithm*]

1280, 1536, 2048 bits[9] that meet the following: **PKCS #1 [R27], FIPS PUB 180-4 [R25]**[10].

Application Note 15 *The TOE uses the RSA library provided by the underlying IC for signature creation (cf. Appendix A). The reason why distinct statements of SFR FCS_COP.1 apply to platform* P6022J VB *is that the certification report of the former [R4] covers RSA key lengths in the range 1024-4096 bits, whereas the certification report of the latter[R4] covers RSA key lengths in the range 1976-4096 bits only..*

8.2 Class FDP: User data protection

The security attributes of subjects and objects relevant for access control and the related values are reported in Table 8-2.

Table 8-2 Security attributes of subjects and objects for access control

Subject or object	Security attribute	Security attribute values
S.User	Role	R.Admin, R.Sigy
S.User	SCD/SVD management	authorized, not authorized
SCD	SCD operational	yes, no
SCD	SCD identifier	arbitrary value
SVD	-	-
DTBS/R	-	-

Application Note 16 *DTBS/R has been added to the list of subjects and objects provided in Table 8-2 because it is mentioned in SFRs FDP_ACC.1/Signature creation and FDP_ACF.1/Signature creation.*

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

- SCD,
- SVD.

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

[9] [assignment: *cryptographic key sizes*]
[10] [assignment: *list of standards*]

8.2.1 FDP_ACC.1/SCD/SVD_Generation

Subset access control – SCD/SVD generation

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/SCD/SVD_Generation:

The TSF shall enforce the SCD/SVD Generation SFP[11] on

- subjects: S.User,
- objects: SCD, SVD,
- operations: generation of SCD/SVD pairs[12].

8.2.2 FDP_ACF.1/SCD/SVD_Generation

Security attribute based access control – SCD/SVD generation

Hierarchical to: No other components.

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

FDP_ACF.1.1/SCD/SVD_Generation:

The TSF shall enforce the SCD/SVD Generation SFP[13] to objects based on the following: the user S.User is associated with the security attribute "SCD/SVD management"[14].

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:

[11] [assignment: *access control SFP*]
[12] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]
[13] [assignment: *access control SFP*]
[14] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

S.User with the security attribute “SCD/SVD management” set to “authorized” is allowed to generate SCD/SVD pairs[15].

FDP_ACF.1.3/SCD/SVD_Generation:

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none[16].

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 pairs[17].

Application Note 17 *Both the Administrator and the Signatory are allowed to generate SCD/SVD pairs (cf. section 2.2.2).*

8.2.3 FDP_ACC.1/SVD_Transfer

Subset access control – SVD transfer

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/SVD_Transfer:

The TSF shall enforce the SVD Transfer SFP[18] on

- subjects: S.User,
- objects: SVD,
- operations: export[19].

8.2.4 FDP_ACF.1/SVD_Transfer

Security attribute based access control – SVD transfer

[15] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[16] [assignment: *rules, based on security attributes, that explicitly authorize access of subjects to objects*]
[17] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]
[18] [assignment: *access control SFP*]
[19] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

Hierarchical to: No other components.

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

FDP_ACF.1.1/SVD_Transfer:

The TSF shall enforce the SVD Transfer SFP[20] to objects based on the following:

- the S.User is associated with the security attribute "Role",
- the SVD[21].

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, R.Sigy**[22] are allowed to export SVD[23].

FDP_ACF.1.3/SVD_Transfer:

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none[24].

FDP_ACF.1.4/SVD_Transfer:

The TSF shall explicitly deny access of subjects to objects based on the following additional rules: none[25].

Application Note 18 *Both the Administrator and the Signatory are allowed to export SVD to the CGA in order to apply for certificates (cf. section 2.2.2).*

8.2.5 FDP_ACC.1/Signature creation

Subset access control – Signature creation

[20] [assignment: *access control SFP*]
[21] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]
[22] [selection: *R.Admin, R.Sigy*]
[23] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[24] [assignment: *rules, based on security attributes, that explicitly authorize access of subjects to objects*]
[25] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/Signature_Creation:

The TSF shall enforce the Signature Creation SFP[26] on

- subjects: S.User,
- objects: DTBS/R, SCD,
- operations: signature creation[27].

8.2.6 FDP_ACF.1/Signature creation

Security attribute based access control – Signature creation

Hierarchical to: No other components.

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

FDP_ACF.1.1/Signature_Creation:

The TSF shall enforce the Signature Creation SFP[28] to objects based on the following:

- the user S.User is associated with the security attribute "Role", and
- the SCD with the security attribute "SCD operational"[29].

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:

[26] [assignment: *access control SFP*]
[27] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]
[28] [assignment: *access control SFP*]
[29] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

R.Sigy is allowed to create electronic signatures for DTBS/R with SCD whose security attribute “SCD operational” is set to “yes”[30].

FDP_ACF.1.3/Signature_Creation:

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none[31].

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 whose security attribute “SCD operational” is set to “no”[32].

8.2.7 FDP_RIP.1

Subset residual information protection

Hierarchical to: No other components.

Dependencies: No dependencies.

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[33] the following objects: SCD[34].

8.2.8 FDP_SDI.2/Persistent

Stored data integrity monitoring and action – Persistent data

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring

Dependencies: No dependencies.

[30] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[31] [assignment: *rules, based on security attributes, that explicitly authorize access of subjects to objects*]
[32] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]
[33] [selection: *allocation of the resource to, deallocation of the resource from*]
[34] [assignment: *list of objects*]

FDP_SDI.2.1/Persistent:

The TSF shall monitor user data stored in containers controlled by the TSF for integrity errors[35] on all objects, based on the following attributes: integrity checked stored data[36].

FDP_SDI.2.2/Persistent:

Upon detection of a data integrity error, the TSF shall

- prohibit the use of the altered data,
- inform the S.Sigy about the integrity error[37].

8.2.9 FDP_SDI.2/DTBS

Stored data integrity monitoring and action – DTBS

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring

Dependencies: No dependencies.

FDP_SDI.2.1/DTBS:

The TSF shall monitor user data stored in containers controlled by the TSF for integrity errors[38] on all objects, based on the following attributes: integrity checked stored DTBS[39].

FDP_SDI.2.2/DTBS:

Upon detection of a data integrity error, the TSF shall

- prohibit the use of the altered data,
- inform the S.Sigy about the integrity error[40].

Application Note 19 *The integrity of TSF data like RAD is also protected to ensure the effectiveness of the user authentication.*

[35] [assignment: *integrity errors*]
[36] [assignment: *user data attributes*]
[37] [assignment: *action to be taken*]
[38] [assignment: *integrity errors*]
[39] [assignment: *user data attributes*]
[40] [assignment: *action to be taken*]

8.2.10 FDP_DAU.2/SVD

Data authentication with identity of guarantor

Hierarchical to: FDP_DAU.1 Basic data authentication

Dependencies: FIA_UID.1 Timing of identification

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[41].

FDP_DAU.2.2/SVD:

The TSF shall provide the CGA[42] with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

Application Note 20 *As a means to generate evidence that can be used by the CGA as a guarantee of the validity of SVD, as well as of the identity of the corresponding legitimate Signatory, the TOE QSCD application supports Client/Server Authentication compliant with IAS ECC specification [R14]. For more details, cf. section 2.2.2.*

Application Note 21 *The TOE uses the RSA library provided by the underlying IC for Client/Server Authentication (cf. Appendix A). However, the certification report of platform* P6022J VB *[R4] covers RSA key lengths in the range 1024-4096 bits, while the certification report of platform JCOP3 [R4] covers RSA key lengths in the range 1976-4096 bits only.*

8.2.11 FDP_UIT.1/DTBS

Data exchange integrity

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FTP_ITC.1 Inter-TSF trusted channel, or
FTP_TRP.1 Trusted path]

FDP_UIT.1.1/DTBS:

[41] [assignment: *list of objects or information types*]
[42] [assignment: *list of subjects*]

The TSF shall enforce the Signature Creation SFP[43] to receive[44] user data in a manner protected from modification and insertion[45] errors.

FDP_UIT.1.2/DTBS:

The TSF shall be able to determine on receipt of user data, whether modification or insertion[46] has occurred.

8.3 Class FIA: Identification and authentication

8.3.1 FIA_UID.1

Timing of identification

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UID.1.1:

The TSF shall allow

- self-test according to FPT_TST.1,
- **establishing a trusted channel between the CGA and the TOE by means of TSF required by FTP_ITC.1/SVD,**
- **establishing a trusted channel between the HID and the TOE by means of TSF required by FTP_ITC.1/VAD,**
- **establishing a trusted channel between the Personalization Agent's terminal and the TOE by means of TSF required by FTP_ITC.1/Pers,**
- **returning product/chip information to the initialization Agent**[47] [48]

[43] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]
[44] [selection: *transmit, receive*]
[45] [selection: *modification, deletion, insertion, replay*]
[46] [selection: *modification, deletion, insertion, replay*]
[47] [assignment: *list of additional TSF-mediated actions*]
[48] [assignment: *list of TSF-mediated actions*]

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 22 *The TOE does not maintain any user identification information prior to user authentication; namely, the user is regarded as an unidentified terminal until user authentication is accomplished. Hence, this security target performs the assignment of the bullet (2) in the element FIA_UID.1.1 of PP Part 2 [R9] by listing the same actions specified in the statement of SFR FIA_UAU.1.*

8.3.2 FIA_UAU.1

Timing of authentication

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.1.1:

The TSF shall allow

- self-test according to FPT_TST.1,
- ~~identification of the user by means of TSF required by FIA_UID.1,~~
- establishing a trusted channel between the CGA and the TOE by means of TSF required by FTP_ITC.1/SVD,
- establishing a trusted channel between the HID and the TOE by means of TSF required by FTP_ITC.1/VAD,
- **establishing a trusted channel between the Personalization Agent's terminal and the TOE by means of TSF required by FTP_ITC.1/Pers,**
- **returning product/chip information to the initialization Agent**[49] [50]

[49] [assignment: *list of additional TSF-mediated actions*]
[50] [assignment: *list of TSF-mediated actions*]

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

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.

Application Note 23 *The TOE does not maintain any user identification information prior to user authentication; namely, the user is regarded as an unidentified terminal until user authentication is accomplished. Hence, this security target refines the element FIA_UAU.1.1 by deleting the bullet (2).*

Application Note 24 *PP Part 4 [R10] performs the assignment of the bullet (3) in the element FIA_UAU.1.1 of PP Part 2 [R9] by adding the establishment of a trusted channel to the CGA.*

Application Note 25 *PP Part 5 [R11] performs the assignment of the bullet (3) in the element FIA_UAU.1.1 of PP Part 2 [R9] by adding the establishment of a trusted channel to the HID.*

Application Note 26 *During TOE initialization (cf. section 2.3), the initialization Agent can retrieve product/chip information before authentication and then establish a trusted channel with the TOE.*

Application Note 27 *During TOE personalization (cf. section 2.3), the Personalization Agent can establish a trusted channel with the TOE.*

8.3.3 FIA_AFL.1

Authentication failure handling

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1:

The TSF shall detect when **a defined number within the ranges specified in Table 8-3, column 1, of** unsuccessful authentication attempts occur related to consecutive failed

authentication attempts[51] **with respect to the authentication procedures specified in Table 8-3, column 2**.

FIA_AFL.1.2:

When the defined number of unsuccessful authentication attempts has been met[52], the TSF shall **perform the actions specified in Table 8-3, column 3**.

Table 8-3 FIA_AFL.1 refinement

Range	Authentication procedure	Action
From 1 to 15	Authentication with respect to the initialization key	Block of the initialization key
From 1 to 255	Authentication with respect to the initialization key	Block of the initialization key
From 1 to 255	Authentication with respect to the personalization key	Block of the personalization key
From 1 to 255	Authentication with respect to Administrator's/Signatory's PACE keys	Return of the authentication outcome with a delay of a few seconds until a successful authentication is performed
From 1 to 255	Authentication with respect to Administrator's/Signatory's passwords	Block of the password

Application Note 28 *This security target refines SFR FIA_AFL.1 by extending it so as to consider all the authentication mechanisms supported by the TOE (cf. SFR FIA_UAU.1).*

Application Note 29 *Distinct thresholds within the specified ranges apply to both steps of Signatory's authentication with respect to the RAD, namely PACE authentication and password verification (cf. section 2.2.1). If the threshold for PACE authentication attempts is reached, the outcome of subsequent attempts is returned with a delay of a few seconds until a successful authentication is performed. If the threshold for password verification attempts is reached, the password is blocked, which enforces the block of the RAD as a whole.*

8.3.4 FIA_API.1

Authentication proof of identity

Hierarchical to: No other components.

Dependencies: No dependencies.

[51] [assignment: *list of authentication events*]
[52] [selection: *met, surpassed*]

FIA_API.1.1:

The TSF shall provide **Client/Server Authentication compliant with IAS ECC specification [R14]**[53] to prove the identity of the QSCD[54].

Application Note 30 *Via Client/Server Authentication, the TOE is able to authenticate itself as QSCD to the CGA (cf. section 2.2.2), using authentication data implemented in the TOE before the QSCD preparation phase (cf. section 2.3).*

Application Note 31 *The TOE uses the RSA library provided by the underlying IC for Client/Server Authentication (cf. Appendix A). However, the certification report of platform* P6022J VB *[R4] covers RSA key lengths in the range 1024-4096 bits, while the certification report of platform* P6022J VB *[R4]covers RSA key lengths in the range 1976-4096 bits only..*

8.4 Class FMT: Security management

8.4.1 FMT_SMR.1

Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1:

The TSF shall maintain the roles R.Admin and R.Sigy[55]**, R.Init, and R.Pers**.

FMT_SMR.1.2:

The TSF shall be able to associate users with roles.

Application Note 32 *This security target refines SFR FMT_SMR.1 by extending it so as to consider all the roles supported by the TOE (cf. section 4.1).*

[53] [assignment: *authentication mechanism*]
[54] [assignment: *authorized user or role*]
[55] [assignment: *the authorized identified roles*]

8.4.2 FMT_SMF.1

Specification of management functions

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1:

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

- creation and modification of RAD,
- enabling the signature creation function,
- modification of the security attributes “SCD/SVD management”, “SCD operational”,
- change the default value of the security attribute “SCD identifier”,
- **unblock of RAD,**
- **writing initialization data,**
- **writing personalization data**[56] [57].

8.4.3 FMT_MOF.1

Management of security functions behaviour

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of management functions

FMT_MOF.1.1:

The TSF shall restrict the ability to enable[58] the functions signature creation function[59] to R.Sigy[60].

[56] [assignment: *list of other security management functions to be provided by the TSF*]
[57] [assignment: *list of security management functions to be provided by the TSF*]
[58] [selection: *determine the behaviour of, disable, enable, modify the behaviour of*]
[59] [assignment: *list of functions*]
[60] [assignment: *the authorized identified roles*]

8.4.4 FMT_MSA.1/Admin

Management of security attributes – Administrator

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of management functions

FMT_MSA.1.1/Admin:

The TSF shall enforce the SCD/SVD Generation SFP[61] to restrict the ability to modify, **none**[62] [63] the security attributes "SCD/SVD management"[64] to R.Admin[65].

8.4.5 FMT_MSA.1/Signatory

Management of security attributes – Signatory

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of management functions

FMT_MSA.1.1/Signatory:

The TSF shall enforce the Signature Creation SFP[66] to restrict the ability to modify[67] the security attributes "SCD operational"[68] to R.Sigy[69].

[61] [assignment: *access control SFP(s), information flow control SFP(s)*]
[62] [assignment: *other operations*]
[63] [selection: *change_default, query, modify, delete, [assignment: other operations]*]
[64] [assignment: *list of security attributes*]
[65] [assignment: *the authorized identified roles*]
[66] [assignment: *access control SFP(s), information flow control SFP(s)*]
[67] [selection: *change_default, query, modify, delete, [assignment: other operations]*]
[68] [assignment: *list of security attributes*]
[69] [assignment: *the authorized identified roles*]

8.4.6 FMT_MSA.2

Secure security attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MSA.2.1:

The TSF shall ensure that only secure values are accepted for "SCD/SVD management" and "SCD operational"[70].

Application Note 33 *Since the TOE supports generation of SCD/SVD pairs on the part of both the Administrator and the Signatory and a trusted channel for export of the SVD to the CGA, the security attribute "SCD/SVD management" is set to "yes" for both of subjects S.Admin and S.Sigy (cf. sections 2.2.2, 2.3).*

8.4.7 FMT_MSA.3

Static attribute initialization

Hierarchical to: No other components.

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

FMT_MSA.3.1:

The TSF shall enforce the SCD/SVD Generation SFP, SVD Transfer SFP, and Signature Creation SFP[71] to provide restrictive[72] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2:

[70] [assignment: *list of security attributes*]
[71] [assignment: *access control SFP, information flow control SFP*]
[72] [selection, choose one of: *restrictive, permissive, [assignment: other property]*]

The TSF shall allow the R.Admin[73] to specify alternative initial values to override the default values when an object or information is created.

8.4.8 FMT_MSA.4

Security attribute value inheritance

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]

FMT_MSA.4.1:

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

- 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.
- 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[74].

8.4.9 FMT_MTD.1/Admin

Management of TSF data – Administrator

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of management functions

FMT_MTD.1.1/Admin:

[73] [assignment: *the authorized identified roles*]
[74] [assignment: *rules for setting the values of security attributes*]

The TSF shall restrict the ability to create[75] the RAD[76] to R.Admin[77].

8.4.10 FMT_MTD.1/Signatory

Management of TSF data – Signatory

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of management functions

FMT_MTD.1.1/Signatory:

The TSF shall restrict the ability to modify, **unblock**[78] [79] the RAD[80] to R.Sigy[81].

8.4.11 FMT_MTD.1/Init

Management of TSF data – initialization Agent

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of management functions

FMT_MTD.1.1/Init:

The TSF shall restrict the ability to **write**[82] the **initialization data**[83] to **R.Init**[84].

[75] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]
[76] [assignment: *list of TSF data*]
[77] [assignment: *the authorized identified roles*]
[78] [assignment: *other operations*]
[79] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]
[80] [assignment: *list of TSF data*]
[81] [assignment: *the authorized identified roles*]
[82] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]
[83] [assignment: *list of TSF data*]
[84] [assignment: *the authorized identified roles*]

8.4.12 FMT_MTD.1/Pers

Management of TSF data – Personalization Agent

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of management functions

FMT_MTD.1.1/Pers:

The TSF shall restrict the ability to **write**[85] the **personalization data**[86] to **R.Pers**[87].

8.5 Class FPT: Protection of the TSF

8.5.1 FPT_EMS.1

TOE emanation

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_EMS.1.1:

The TOE shall not emit **any measurable emissions**[88] in excess of **intelligible thresholds**[89] enabling access to RAD[90] and SCD[91].

FPT_EMS.1.2:

[85] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]
[86] [assignment: *list of TSF data*]
[87] [assignment: *the authorized identified roles*]
[88] [assignment: *types of emissions*]
[89] [assignment: *specified limits*]
[90] [assignment: *list of types of TSF data*]
[91] [assignment: *list of types of user data*]

The TSF shall ensure **any users**[92] are unable to use the following interface **contact-based/contactless interface and circuit contacts**[93] to gain access to RAD[94] and SCD[95].

Application Note 34 *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.*

8.5.2 FPT_FLS.1

Failure with preservation of secure state

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_FLS.1.1:

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

- self-test according to FPT_TST fails,
- **a physical attack is detected**[96] [97].

Application Note 35 *The assignments address failures detected by a failed self-test or revealing the occurrence of a physical attack, and requiring appropriate action to prevent*

[92] [assignment: *type of users*]
[93] [assignment: *type of connection*]
[94] [assignment: *list of types of TSF data*]
[95] [assignment: *list of types of user data*]
[96] [assignment: *list of other types of failures in the TSF*]
[97] [assignment: *list of types of failures in the TSF*]

security violations. 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.

8.5.3 FPT_PHP.1

Passive detection of physical attack

Hierarchical to: No other components.

Dependencies: No dependencies.

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.

8.5.4 FPT_PHP.3

Resistance to physical attack

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_PHP.3.1:

The TSF shall resist **physical manipulation and physical probing**[98] to the **TSF**[99] by responding automatically such that the SFRs are always enforced.

Application Note 36 *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 (i) assuming that there might be an attack at any time, and (ii) 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*

[98] [assignment: *physical tampering scenarios*]
[99] [assignment: *list of TSF devices/elements*]

physical tampering must not reveal information of the SCD. E.g. the TOE may be physically tampered in the power-off state of the TOE, which does not allow the 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 TSF 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 to physical tampering in such a way that the Signatory is able to determine whether the TOE was physically tampered or not. The guidance documentation identifies the failure of TOE start-up as an indication of physical tampering [R1] [R2] [R3].

8.5.5 FPT_TST.1

TSF testing

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_TST.1.1:

The TSF shall run a suite of self-tests **during initial start-up**[100] to demonstrate the correct operation of the TSF[101].

FPT_TST.1.2:

The TSF shall provide authorized users with the capability to verify the integrity of TSF data[102].

FPT_TST.1.3:

The TSF shall provide authorized users with the capability to verify the integrity of TSF[103].

8.6 Class FTP: Trusted path/channels

8.6.1 FTP_ITC.1/SVD

Inter-TSF trusted channel – SVD

[100] [selection: *during initial start-up, periodically during normal operation, at the request of the authorized user, at the conditions [assignment: conditions under which self-test should occur]*]
[101] [selection: *[assignment: parts of TSF], the TSF*]
[102] [selection: *[assignment: parts of TSF data], TSF data*]
[103] [selection: *[assignment: parts of TSF], TSF*]

Hierarchical to: No other components.

Dependencies: No dependencies.

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 assured 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[104] 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

- data authentication with identity of guarantor according to FIA_API.1 and FDP_DAU.2/SVD,
- **import of certificate info from the CGA**[105] [106].

Application Note 37 *The component FTP_ITC.1/SVD requires the TSF to enforce a trusted channel established by the CGA to export the SVD to the CGA. Moreover, the TSF requires the use of the same trusted channel for the import of certificate info from the CGA (cf. section 2.2.2).*

8.6.2 FTP_ITC.1/VAD

Inter-TSF trusted channel – VAD

Hierarchical to: No other components.

Dependencies: No dependencies.

[104] [selection: *the TSF, another trusted IT product*]
[105] [assignment: *list of other functions for which a trusted channel is required*]
[106] [assignment: *list of functions for which a trusted channel is required*]

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 another trusted IT product[107] 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

- user authentication according to FIA_UAU.1,
- **none**[108] [109].

Application Note 38 *The component FTP_ITC.1/VAD requires the TSF to enforce a trusted channel established by the HID to import the VAD from the HID. In more detail, the trusted channel is opened by means of PACE authentication using a key derived from the first VAD component, i.e. Signatory's password #1, and then the second VAD component, i.e. Signatory's password #2, must be sent to the TSF over this trusted channel (cf. section 2.2.1).*

8.6.3 FTP_ITC.1/DTBS

Inter-TSF trusted channel – DTBS

Hierarchical to: No other components.

Dependencies: No dependencies.

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

[107] [selection: *the TSF, another trusted IT product*]
[108] [assignment: *list of other functions for which a trusted channel is required*]
[109] [assignment: *list of functions for which a trusted channel is required*]

identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2/DTBS:

The TSF shall permit another trusted IT product[110] 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

- signature creation,
- **export of public keys, certificate info, and digital signatures to the SCA**[111] [112].

Application Note 39 *The component FTP_ITC.1/DTBS requires the TSF to enforce a trusted channel established by the SCA to import the DTBS from the SCA. Moreover, the TSF requires the use of the same trusted channel for the export of public keys, certificate info, and digital signatures to the SCA (cf. section 2.2.3).*

8.6.4 FTP_ITC.1/Pers

Inter-TSF trusted channel – Personalization data

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1/Pers:

The TSF shall provide a communication channel between itself and another trusted IT product**, the Personalization Agent's terminal,** 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/Pers:

[110] [selection: *the TSF, another trusted IT product*]
[111] [assignment: *list of other functions for which a trusted channel is required*]
[112] [assignment: *list of functions for which a trusted channel is required*]

The TSF shall permit **another trusted IT product**[113] to initiate communication via the trusted channel.

FTP_ITC.1.3/Pers:

The TSF **or the Personalization Agent's terminal** shall initiate communication via the trusted channel for **import of personalization data from the terminal**[114].

Application Note 40 *The component FTP_ITC.1/Pers requires the TSF to enforce a trusted channel established by the Personalization Agent's terminal to import personalization data from the terminal. This trusted channel is established authentication and uses cryptographic algorithm TDES [R24]..*

[113] [selection: *the TSF, another trusted IT product*]
[114] [assignment: *list of functions for which a trusted channel is required*]

9. Security requirements rationale

9.1 Coverage of security functional requirements

Table 9-1 maps the security functional requirements to the security objectives for the TOE. The rows are split according to SFR classes, while the columns are split according to the source of the security objectives (PP Part 2 [R9], PP Part 4 [R10], PP Part 5 [R11], or this security target).

Table 9-1 Mapping of the security functional requirements to the security objectives for the TOE

	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_QSCD_Auth	OT.TOE_TC_SVD_Exp	OT.TOE_TC_VAD_Imp	OT.TOE_TC_DTBS_Imp	OT.AC_	OT.AC_Pers
FCS_CKM.1	X		X	X	X												
FCS_CKM.4	X				X												
FCS_COP.1	X					X											
FDP_ACC.1/ SCD/SVD_Generation	X	X															
FDP_ACF.1/ SCD/SVD_Generation	X	X															
FDP_ACC.1/ SVD_Transfer	X												X				
FDP_ACF.1/ SVD_Transfer	X												X				
FDP_ACC.1/ Signature creation	X						X										
FDP_ACF.1/ Signature creation	X						X										
FDP_RIP.1					X		X										
FDP_SDI.2/ Persistent				X	X	X											
FDP_SDI.2/ DTBS							X	X									

	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_QSCD_Auth	OT.TOE_TC_SVD_Exp	OT.TOE_TC_VAD_Imp	OT.TOE_TC_DTBS_Imp	OT.AC_	OT.AC_Pers
FDP_DAU.2/ SVD													X				
FDP_UIT.1/ DTBS															X		
FIA_UID.1		X					X									X	X
FIA_UAU.1		X					X					X				X	X
FIA_AFL.1							X									X	X
FIA_API.1												X					
FMT_SMR.1	X						X									X	X
FMT_SMF.1	X			X			X									X	X
FMT_MOF.1	X						X										
FMT_MSA.1/ Admin	X	X															
FMT_MSA.1/ Signatory	X						X										
FMT_MSA.2	X	X					X										
FMT_MSA.3	X	X		X			X										
FMT_MSA.4	X	X					X										
FMT_MTD.1/ Admin	X						X										
FMT_MTD.1/ Signatory	X						X										
FMT_MTD.1/ Init																X	
FMT_MTD.1/ Pers																	X
FPT_EMS.1					X				X								
FPT_FLS.1					X												

	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_QSCD_Auth	OT.TOE_TC_SVD_Exp	OT.TOE_TC_VAD_Imp	OT.TOE_TC_DTBS_Imp	OT.AC_	OT.AC_Pers
FPT_PHP.1										X							
FPT_PHP.3					X						X						
FPT_TST.1	X				X	X											
FTP_ITC.1/ SVD													X				
FTP_ITC.1/ VAD														X			
FTP_ITC.1/ DTBS															X		
FTP_ITC.1/ Pers																	X

9.2 Sufficiency of security functional requirements

Here below is the rationale borrowed from PP Part 2 [R9].

OT.Lifecycle_Security *(Life cycle security)* is provided by the SFRs for SCD/SVD generation ***FCS_CKM.1***, SCD usage ***FCS_COP.1***, and SCD destruction ***FCS_CKM.4***, which ensure a cryptographically secure life cycle of the SCD. The SCD/SVD generation is controlled by TSF according to ***FDP_ACC.1/SCD/SVD_Generation*** and ***FDP_ACF.1/SCD/SVD_Generation***. The SVD transfer for certificate generation is controlled by TSF according to ***FDP_ACC.1/SVD_Transfer*** and ***FDP_ACF.1/SVD_Transfer***. The SCD usage is ensured by access control ***FDP_ACC.1/Signature_Creation***, ***FDP_AFC.1/Signature_Creation***, which is based on secure TSF management according to ***FMT_MOF.1***, ***FMT_MSA.1/Admin***, ***FMT_MSA.1/Signatory***, ***FMT_MSA.2***, ***FMT_MSA.3***, ***FMT_MSA.4***, ***FMT_MTD.1/Admin***, ***FMT_MTD.1/Signatory***, ***FMT_SMF.1***, and ***FMT_SMR.1***. The test functions ***FPT_TST.1*** provide failure detection throughout the life cycle.

OT.SCD/SVD_Auth_Gen *(Authorized SCD/SVD generation)* addresses that generation of an SCD/SVD pair requires proper user authentication. The TSF specified by ***FIA_UID.1*** and ***FIA_UAU.1*** provide user identification and user authentication prior to enabling access to authorized functions. The SFRs ***FDP_ACC.1/SCD/SVD_Generation*** and

FDP_ACF.1/SCD/SVD_Generation provide access control for the SCD/SVD generation. The security attributes of the authenticated user are provided by ***FMT_MSA.1/Admin***, ***FMT_MSA.2***, and ***FMT_MSA.3*** for static attribute initialization. The SFR ***FMT_MSA.4*** defines rules for inheritance of the security attribute "SCD operational" of the SCD.

OT.SCD_Unique *(Uniqueness of Signature Creation Data)* implements the requirement of practically unique SCD as laid down in [R13], Annex III, paragraph 1(a), which is provided by the cryptographic algorithms specified by ***FCS_CKM.1***.

OT.SCD_SVD_Corresp *(Correspondence between SVD and SCD)* addresses that the SVD corresponds to the SCD implemented by the TOE. This is provided by the algorithms specified by ***FCS_CKM.1*** to generate corresponding SVD/SCD pairs. The security functions specified by ***FDP_SDI.2/Persistent*** ensure that the keys are not modified, so to retain the correspondence. Moreover, the SCD identifier allows the environment to identify the SCD and to link it with the appropriate SVD. The management functions identified by ***FMT_SMF.1*** and by ***FMT_MSA.3*** allow R.Admin to modify the default value of the security attribute "SCD identifier".

OT.SCD_Secrecy *(Secrecy of Signature Creation Data)* is provided by the security functions specified by the following SFRs. ***FCS_CKM.1*** ensures the use of secure cryptographic algorithms for SCD/SVD generation. Cryptographic quality of SCD/SVD pairs shall prevent disclosure of SCD by cryptographic attacks using the publicly known SVD. The security functions specified by ***FDP_RIP.1*** and ***FCS_CKM.4*** ensure that residual information on SCD is destroyed after the SCD has been used for signature creation and that destruction of SCD leaves no residual information.

The security functions specified by ***FDP_SDI.2/Persistent*** ensure that no critical data are modified which could alter the efficiency of the security functions or leak information on the SCD. ***FPT_TST.1*** tests the working conditions of the TOE, and ***FPT_FLS.1*** guarantees a secure state when integrity is violated and thus assures that the specified security functions are operational. An example where compromising error conditions are countered by ***FPT_FLS.1*** is fault injection for Differential Fault Analysis (DFA).

SFRs ***FPT_EMS.1*** and ***FPT_PHP.3*** require additional security features of the TOE to ensure the confidentiality of the SCD.

OT.Sig_Secure *(Cryptographic security of the electronic signature)* is provided by the cryptographic algorithms specified by ***FCS_COP.1***, which ensures the cryptographic robustness of the signature algorithms. ***FDP_SDI.2/Persistent*** corresponds to the integrity of the SCD implemented by the TOE, and ***FPT_TST.1*** ensures self-tests ensuring correct signature creation.

OT.Sigy_SigF *(Signature creation function for the legitimate Signatory only)* is provided by SFRs for identification, authentication, and access control.

FIA_UAU.1 and ***FIA_UID.1*** ensure that no signature creation function can be invoked before the Signatory is identified and authenticated. The security functions specified by ***FMT_MTD.1/Admin*** and ***FMT_MTD.1/Signatory*** manage the authentication function. SFR ***FIA_AFL.1*** provides protection against a number of attacks, such as cryptographic extraction of residual information, or brute force attacks against authentication. The security function specified by ***FDP_SDI.2/DTBS*** ensures the integrity of stored DTBS, and ***FDP_RIP.1*** prevents misuse of any resources containing the SCD after de-allocation (e.g. after the signature creation process).

The security functions specified by ***FDP_ACC.1/Signature_Creation*** and ***FDP_ACF.1/Signature_Creation*** provide access control based on the security attributes managed according to the SFRs ***FMT_MTD.1/Signatory***, ***FMT_MSA.2***, ***FMT_MSA.3***, and ***FMT_MSA.4***. The SFRs ***FMT_SMF.1*** and ***FMT_SMR.1*** list these management functions and the roles. These ensure that the signature process is restricted to the Signatory. ***FMT_MOF.1*** restricts the ability to enable the signature creation function to the Signatory. ***FMT_MSA.1/Signatory*** restricts the ability to modify the security attribute “SCD operational” to the Signatory.

OT.DTBS_Integrity_TOE *(DTBS/R integrity inside the TOE)* ensures that the DTBS/R is not altered by the TOE. The integrity functions specified by ***FDP_SDI.2/DTBS*** require that the DTBS/R has not been altered by the TOE.

OT.EMSEC_Design *(Provision of physical emanations security)* requires that no intelligible information is emanated. This is provided by ***FPT_EMS.1***.

OT.Tamper_ID *(Tamper detection)* is provided by ***FPT_PHP.1*** by means of passive detection of physical attacks.

OT.Tamper_Resistance *(Tamper resistance)* is provided by ***FPT_PHP.3*** to resist physical attacks.

Here below is the rationale borrowed from PP Part 4 [R10].

OT.TOE_QSCD_Auth *(Authentication proof as QSCD)* requires the TOE to provide security mechanisms to identify and to authenticate itself as QSCD, which is directly provided by ***FIA_API.1***. The SFR ***FIA_UAU.1*** allows establishment of the trusted channel before the (human) user is authenticated.

OT.TOE_TC_SVD_Exp *(TOE trusted channel for SVD export)* requires the TOE to provide a trusted channel to the CGA to protect the integrity of the SVD exported to the CGA, which is directly provided by:

- The SVD transfer for certificate generation is controlled by TSF according to ***FDP_ACC.1/SVD_Transfer*** and ***FDP_ACF.1/SVD_Transfer***.

- ***FDP_DAU.2/SVD***, which requires the TOE to provide the CGA with the ability to verify evidence of the validity of the SVD and the identity of the user that generated the evidence.
- ***FTP_ITC.1/SVD***, which requires the TOE to provide a trusted channel to the CGA.

Here below is the rationale borrowed from PP Part 5 [R11].

OT.TOE_TC_VAD_Imp *(TOE trusted channel for VAD import)* is met by ***FTP_ITC.1/VAD***, which requires the TSF to enforce a trusted channel to protect the VAD provided by the HID to the TOE.

OT.TOE_TC_DTBS_Imp *(TOE trusted channel for DTBS import)* is covered by ***FTP_ITC.1/DTBS***, which requires the TSF to enforce a trusted channel to protect the DTBS provided by the SCA to the TOE, and by ***FDP_UIT.1/DTBS***, which requires the TSF to verify the integrity of the received DTBS.

Here below is the rationale for the security objectives added in this security target to those defined in the PPs.

OT.AC_Init *(Access control for the initialization of the e-Document)* is covered by:

- ***FIA_UID.1*** and ***FIA_UAU.1***, which state that writing TOE initialization data requires a previous authentication on the part of the Initialization Agent;
- ***FIA_AFL.1***, which specifies how unsuccessful authentication attempts are managed for the authentication as Initialization Agent;
- ***FMT_MTD.1/Init*** (based on ***FMT_SMR.1*** and ***FMT_SMF.1***), which restricts the capability to write TOE initialization data to the Initialization Agent;

OT.AC_Pers *(Access control for the personalization of the e-Document)* is covered by:

- ***FIA_UID.1*** and ***FIA_UAU.1***, which state that writing personalization data requires a previous authentication on the part of the Personalization Agent;
- ***FIA_AFL.1***, which specifies how unsuccessful authentication attempts are managed for the authentication as Personalization Agent;
- ***FMT_MTD.1/Pers*** (based on ***FMT_SMR.1*** and ***FMT_SMF.1***), which restricts the capability to write personalization data to the Personalization Agent;
- ***FTP_ITC.1/Pers***, which requires the TSF to enforce a trusted channel for the import of personalization data, so as to ensure that the data actually written match those sent by the Personalization Agent.

9.3 Satisfaction of dependencies of security requirements

Table 9-2 Satisfaction of dependencies of security functional requirements

Requirement	Dependencies	Satisfied by
FCS_CKM.1	FCS_CKM.2 or FCS_COP.1	FCS_COP.1
	FCS_CKM.4	FCS_CKM.4
FCS_CKM.4	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	FCS_CKM.1
FCS_COP.1	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	FCS_CKM.1
	FCS_CKM.4	FCS_CKM.4
FDP_ACC.1/ SCD/SVD_Generation	FDP_ACF.1	FDP_ACF.1/ SCD/SVD_Generation
FDP_ACF.1/ SCD/SVD_Generation	FDP_ACC.1	FDP_ACC.1/ SCD/SVD_Generation
	FMT_MSA.3	FMT_MSA.3
FDP_ACC.1/ SVD_Transfer	FDP_ACF.1	FDP_ACF.1/ SVD_Transfer
FDP_ACF.1/ SVD_Transfer	FDP_ACC.1	FDP_ACC.1/ SVD_Transfer
	FMT_MSA.3	FMT_MSA.3
FDP_ACC.1/ Signature_Creation	FDP_ACF.1	FDP_ACF.1/ Signature_Creation
FDP_ACF.1/ Signature_Creation	FDP_ACC.1	FDP_ACC.1/ Signature_Creation
	FMT_MSA.3	FMT_MSA.3
FDR_RIP.1	No dependencies	-
FDP_SDI.2/Persistent	No dependencies	-
FDP_SDI.2/DTBS	No dependencies	-
FDP_DAU.2/SVD	FIA_UID.1	FIA_UID.1
FDP_UIT.1/DTBS	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/ Signature_Creation
	FTP_ITC.1 or FTP_TRP.1	FTP_ITC.1/DTBS

Requirement	Dependencies	Satisfied by
FIA_UID.1	No dependencies	-
FIA_UAU.1	FIA_UID.1	FIA_UID.1
FIA_AFL.1	FIA_UAU.1	FIA_UAU.1
FIA_API.1	No dependencies	-
FMT_SMR.1	FIA_UID.1	FIA_UID.1
FMT_SMF.1	No dependencies	-
FMT_MOF.1	FMT_SMR.1	FMT_SMR.1
	FMT_SMF.1	FMT_SMF.1
FMT_MSA.1/Admin	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/ SCD/SVD_Generation
	FMT_SMR.1	FMT_SMR.1
	FMT_SMF.1	FMT_SMF.1
FMT_MSA.1/Signatory	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/ Signature_Creation
	FMT_SMR.1	FMT_SMR.1
	FMT_SMF.1	FMT_SMF.1
FMT_MSA.2	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/ SCD/SVD_Generation, FDP_ACC.1/ Signature_Creation
	FMT_MSA.1	FMT_MSA.1/Admin, FMT_MSA.1/Signatory
	FMT_SMR.1	FMT_SMR.1
FMT_MSA.3	FMT_MSA.1	FMT_MSA.1/Admin, FMT_MSA.1/Signatory
	FMT_SMR.1	FMT_SMR.1
FMT_MSA.4	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/ SCD/SVD_Generation, FDP_ACC.1/ Signature_Creation
FMT_MTD.1/Admin	FMT_SMR.1	FMT_SMR.1

Requirement	Dependencies	Satisfied by
	FMT_SMF.1	FMT_SMF.1
FMT_MTD.1/Signatory	FMT_SMR.1	FMT_SMR.1
	FMT_SMF.1	FMT_SMF.1
	FMT_SMF.1	FMT_SMF.1
FMT_MTD.1/Init	FMT_SMR.1	FMT_SMR.1
	FMT_SMF.1	FMT_SMF.1
FMT_MTD.1/Pers	FMT_SMR.1	FMT_SMR.1
	FMT_SMF.1	FMT_SMF.1
FPT_EMS.1	No dependencies	-
FPT_FLS.1	No dependencies	-
FPT_PHP.1	No dependencies	-
FPT_PHP.3	No dependencies	-
FPT_TST.1	No dependencies	-
FTP_ITC.1/SVD	No dependencies	-
FTP_ITC.1/VAD	No dependencies	-
FTP_ITC.1/DTBS	No dependencies	-
FTP_ITC.1/Pers	No dependencies	-

Table 9-3 Satisfaction of dependencies of security assurance requirements

Requirement	Dependencies	Satisfied by
EAL4 package	Dependencies of the EAL4 package are not reproduced here (cf. [R7])	By construction, all dependencies are satisfied in a CC EAL package
ALC_DVS.2	No dependencies	-

Requirement	Dependencies	Satisfied by
AVA_VAN.5	ADV_ARC.1	ADV_ARC.1[115]
	ADV_FSP.4	ADV_FSP.5
	ADV_TDS.3	ADV_TDS.4
	ADV_IMP.1	ADV_IMP.1
	AGD_OPE.1	AGD_OPE.1
	AGD_PRE.1	AGD_PRE.1
	ATE_DPT.1	ATE_DPT.3

9.4 Rationale for security assurance requirements

The assurance level for this security target is EAL4 augmented. EAL4 permits a developer to gain maximum assurance from security engineering based upon rigorous commercial development practises, supported by moderate application of specialist security engineering techniques. EAL4 is therefore applicable in those circumstances where developers or users require a high level of independently assured security in a planned development and require a rigorous development approach, without incurring unreasonable costs attributable to specialist security engineering techniques (cf. [R7]).

The TOE described in this security target is just such a product. Augmentation results from the selection of:

- ALC_DVS.2 “Sufficiency of security measures”,
- AVA_VAN.5 “Advanced methodical vulnerability analysis”.

The selection of component ALC_DVS.2 provides a higher assurance on the security of the development and manufacturing of the TOE.

The selection of component AVA_VAN.5 ensures that the TOE be resistant to penetration attacks performed by an attacker possessing a high attack potential, which is necessary to meet security objectives OT.SCD_Secrecy, OT.Sigy_SigF, and OT.Sig_Secure (cf. section 5.1).

[115] This assurance component and the subsequent ones are all included in the EAL4 package.

10. TOE summary specification

Table 10-1 describes how each security functional requirement claimed in this security target is satisfied by the TOE.

Table 10-1 Implementation of the security functional requirements in the TOE

Security functional requirement	Implementation
FCS_CKM.1	The private key objects storing the private keys meant for signature creation (cf. section 2.3) contain an algorithm identifier, as well as the length of the key to be generated. Both fields refer to allowed values as specified in the statement of the SFR.
FCS_CKM.4	The private key objects storing the private keys meant for signature creation (cf. section 2.3) are overwritten with zeros in case the keys are destroyed by the Signatory (cf. section 2.3.4).
FCS_COP.1	As specified for SFR FCS_CKM.1.
FDP_ACC.1/SCD/SVD_Generation	As specified for SFR FDP_ACF.1/SCD/SVD_Generation.
FDP_ACF.1/SCD/SVD_Generation	The private key objects storing the private keys meant for signature creation (cf. section 2.3) contain an access condition for key generation, which refers to the logical *OR* of Administrator's and Signatory's credentials (cf. section 2.2.1).
FDP_ACC.1/SVD_Transfer	As specified for SFR FDP_ACF.1/SVD_Transfer.
FDP_ACF.1/SVD_Transfer	The public key objects storing the public keys meant for signature creation (cf. section 2.3) contain an access condition for public key export, which refers to the logical *OR* of Administrator's and Signatory's credentials (cf. section 2.2.1).
FDP_ACC.1/Signature creation	As specified for SFR FDP_ACF.1/Signature creation.
FDP_ACF.1/Signature creation	The private key objects storing the private keys meant for signature creation (cf. section 2.3) contain an access condition for signature creation, as well as a life cycle state compliant with ISO/IEC 7816-9 [R22]. The access condition refers to Signatory's credentials (cf. section 2.2.1); moreover, signature creation is forbidden unless the life cycle state matches "operational activated".

Security functional requirement	Implementation
FDP_RIP.1	Any volatile copy of, or pointer to, a private key meant for signature creation is overwritten with zeros upon the completion of either the generation of the key, or the creation of a signature with the key.
FDP_SDI.2/Persistent	The private/public key objects storing the key pairs meant for signature creation (cf. section 2.3) contain a CRC, which is checked whenever the keys are used for signature creation or public key export. In case such a check fails, the OS enters an endless loop, so that the resulting fall of communication informs the user about the integrity error.
FDP_SDI.2/DTBS	The volatile data structure storing the DTBS/R contains a CRC, which is checked upon signature creation. In case such a check fails, the OS enters an endless loop, so that the resulting fall of communication informs the user about the integrity error.
FDP_DAU.2/SVD	Cf. Application Note 20.
FDP_UIT.1/DTBS	DTBS/R import must be executed over the trusted channel opened by means of Signatory's PACE authentication step (cf. sections 2.2.1, 2.2.3).
FIA_UID.1	Cf. Application Note 22.
FIA_UAU.1	As regards the authentication operations available in pre-operational phases (described in sections 2.3.2 and 2.3.3), cf. Application Note 26, Application Note 27. As regards the authentication operations available in the operational use phase (described in section 2.3.4), cf. section 2.2.1.
FIA_AFL.1	The initialization and personalization keys, as well as PACE key objects and password objects (cf. section 2.3), are associated with distinct thresholds, stored by the actors that write the objects. The behaviour taking place in case of unsuccessful authentication attempts is as specified in the statement of the SFR.
FIA_API.1	Cf. section 2.2.2.

Security functional requirement	Implementation
FMT_SMR.1	The initialization Agent, and Personalization Agent roles are implicitly distinguished via the distinct values of the respective authentication keys. The Administrator and Signatory roles are distinguished by storing the respective credentials into distinct file system objects, viz. distinct PACE key objects and password objects (cf. sections 2.2.1, 2.3), with distinct identifiers. Then, upon user authentication, the OS keeps track of the identifier of the employed credentials.
FMT_SMF.1	Cf. section 2.3.
FMT_MOF.1	The Signatory alone can activate the signature creation function for each single private key, as specified for SFR FMT_MSA.1/Signatory.
FMT_MSA.1/Admin	The private key objects storing the private keys meant for signature creation (cf. section 2.3) contain an access condition for key generation, which is assigned by the Personalization Agent, on behalf of the Administrator, upon creation of the objects (cf. section 2.3.3).
FMT_MSA.1/Signatory	The private key objects storing the private keys meant for signature creation (cf. section 2.3) contain an access condition for the shift of the object life cycle state, which refers to Signatory's credentials, as well as the identifier of Administrator's credentials (cf. section 2.2.1). Upon key generation, the private key object is shifted to the "operational activated" state, unless the identifier of the credentials employed for user authentication matches the Administrator's one stored in the object. In this case, the object is shifted to the "operational deactivated" state, and then the access condition allows the Signatory alone to bring the state to "operational activated".
FMT_MSA.2	As specified for SFRs FMT_MSA.1/Admin, FMT_MSA.1/Signatory.

Security functional requirement	Implementation
FMT_MSA.3	The security attributes applying to key generation and signature creation (as specified for SFRs FDP_ACF.1/SCD/SVD_Generation, FDP_ACF.1/Signature creation), as well as those applying to public key export (as specified for SFR FDP_ACF.1/SVD_Transfer), respectively stored in private and public key objects (cf. section 2.3), are assigned by the Personalization Agent, on behalf of the Administrator, upon creation of the objects (cf. section 2.3.3).
FMT_MSA.4	As specified for SFR FMT_MSA.1/Signatory.
FMT_MTD.1/Admin	The RAD is comprised of Signatory's password #1 and Signatory's password #2 (cf. section 2.2.1). The PACE key object storing the key derived from Signatory's password #1 (cf. section 2.3) is created and filled by the Personalization Agent, on behalf of the Administrator, upon creation of the object (cf. section 2.3.3). The password object storing Signatory's password #2 (cf. section 2.3) contains an access condition for password initialization, which refers to Administrator's credentials (cf. section 2.2.1).
FMT_MTD.1/Signatory	The RAD is comprised of Signatory's password #1 and Signatory's password #2 (cf. section 2.2.1). The PACE key object storing the key derived from Signatory's password #1 (cf. section 2.3) cannot be modified (cf. section 2.2.1) and is never blocked (cf. Application Note 28). The password object storing Signatory's password #2 (cf. section 2.3) contains an access condition for password modification, which refers to Signatory's credentials (cf. section 2.2.1).
FMT_MTD.1/Init	The command APDU available for the writing of initialization data (cf. section 2.3.2) is protected by an implicit access condition, which during initialization requires a previous authentication with respect to the initialization key.

Security functional requirement	Implementation
FMT_MTD.1/Pers	The command APDU available for the writing of personalization data (cf. section 2.3.3) is protected by an implicit access condition, which during personalization requires a previous authentication with respect to the personalization key.
FPT_EMS.1	Leakage of confidential data via side channels is ensured by the security features of both the IC and the OS, in accordance with the security recommendations contained in the IC guidance documentation[R1], [R2], [R3].
FPT_FLS.1	In case self-test fails or a physical attack is detected, the OS enters an endless loop, so that all cryptographic operations and data output interfaces are inhibited.
FPT_PHP.1	Detection of physical attacks is ensured by the security features of both the IC and the OS, in accordance with the security recommendations contained in the IC guidance documentation[R1], [R2], [R3]..
FPT_PHP.3	In case a physical attack is detected, the OS increments an attack counter, stored in the IC persistent memory, and then enters an endless loop. During initial start-up, the OS checks whether the attack counter has reached its threshold value, and enters an endless loop if this is the case. Being executed at any start-up, this mechanism ensures that all cryptographic operations and data output interfaces are permanently inhibited.
FPT_TST.1	During initial start-up, the IC performs a self-test procedure that tests alarm lines and environmental sensor mechanisms (UmSLC test, cf.[R1], [R2], [R3]), and the OS checks the integrity of the TSF by computing a hash value of the code and comparing it with a reference hash value stored internally. Moreover, the integrity of TSF data is checked whenever they are used (as specified for SFR FDP_SDI.2/Persistent as regards private and public keys). In case any one of such checks fails, the OS enters an endless loop, so that the resulting fall of communication informs the user about the integrity error.
FTP_ITC.1/SVD	Cf. Application Note 37.

Security functional requirement	Implementation
FTP_ITC.1/VAD	Cf. Application Note 38.
FTP_ITC.1/DTBS	Cf. Application Note 39.
FTP_ITC.1/Pers	Cf. Application Note 40.

11. Glossary, Abbreviations and References

11.1 Glossary

Table 11-1 defines technical terms pertaining to the TOE on the whole and used throughout this security target. Wherever applicable, terms are associated with the related acronyms.

Table 11-1 Technical terms pertaining to the TOE on the whole

Term	Acronym	Description
Card Manufacturer		Actor that equips the IC with contact-based and/or contactless interfaces, and embeds the IC into a smart card or a document booklet (cf. section 2.3.2).
Configuration data		Data defined by the Embedded Software Developer (cf. section 2.3.1), stored into the IC persistent memory by the IC Manufacturer and possibly updated by the Initialization Agent (cf. section 2.3.2), used to configure global features of the OS (e.g. enabled command APDUs and communication protocols).
Electronic document (e-Document)		The contact-based or contactless smart card integrated into plastic or paper, possibly with an optical readable cover, and providing an ICAO application and/or a QSCD application.
Embedded Software		Software developed by the Embedded Software Developer (cf. section 2.3.1) and stored into the IC persistent memory by the IC Manufacturer (cf. section 2.3.2). Such software consists of the OS, the ICAO application, and the QSCD application.
Embedded Software Developer		HID Global Actor that develops the Embedded Software and the guidance documentation associated with this TOE component (cf. section 2.3.1).
IC Dedicated Software		Software developed by the IC Developer (cf. section 2.3.1) and stored into the IC persistent memory by the IC Manufacturer (cf. section 2.3.2). Such software might support special functionality of the IC hardware and be used, amongst other, for

Term	Acronym	Description
		implementing delivery procedures between different players. The usage of parts of the IC Dedicated Software might be restricted to certain life cycle phases.
IC Developer and Manucfacturer		NXP Actor who: 1. develops the integrated circuit, the IC multi-applications operating system, and the guidance documentation associated with these TOE components (cf. section 2.3.1). 2. produces the TOE integrated circuit, containing the IC Dedicated Software and the Embedded Software, and stores the IC initialization data into the IC persistent memory (cf. section 2.3.2).
Initialization Agent		NXP User in charge of performing the initialization of the TOE, particularly of writing TOE initialization data (cf. section 2.3.2).
Initialization key		Cryptographic key used to encrypt initialization Key.
Integrated Circuit	IC	Electronic component designed to perform processing and/or memory functions. The e-Document's chip is an integrated circuit.
Password Authenticated Connection Establishment	PACE	A communication establishment protocol defined in [R18]. The PACE protocol is a password-authenticated Diffie-Hellman key agreement protocol, providing implicit password-based authentication of the communication partners (e.g. the smart card and the terminal connected); i.e., PACE provides a verification whether the communication partners share the same value of a password. Based on this authentication, PACE also provides a secure communication, whereby confidentiality and authenticity of data transferred within this communication channel are maintained.
Personalization Agent		User in charge of performing the personalization of the TOE, particularly of writing personalization data (cf. section 2.3.3).

Term	Acronym	Description
Personalization data		Data defined and stored into the IC persistent memory by the Personalization Agent. Particularly, they include Administrator's credentials and part of Signatory's credentials (cf. Table 11-2 and section 2.3.3).
Personalization key		Cryptographic key used by the Personalization Agent for mutual authentication with the TOE.
initialization Agent		User in charge of performing the initialization of the TOE, particularly of writing initialization data (cf. section 2.3.2).
initialization data		Data defined and stored into the IC persistent memory by the initialization Agent. Particularly, they include the personalization key (cf. section 2.3.2).
initialization key		Cryptographic key used by the initialization Agent for mutual authentication with the TOE.
Qualified Signature Creation Device	QSCD	Configured software or hardware which is used to implement signature creation and which meets the requirements laid down in [R13], Annex III ([R13], articles 2.5 and 2.6).
QSCD application		A part of the TOE containing non-executable, related user data as well as data needed for authentication, intended to be used, amongst other, as a Qualified Signature Creation Device (QSCD).
Terminal		Any technical system communicating with the TOE through either the contact-based or the contactless interface.
TOE identification data		Data defined by the Embedded Software Developer and stored into the IC persistent memory by the IC Manufacturer (cf. section 2.3.2), used to unambiguously identify the TOE subject to Common Criteria evaluation (cf. section 1.3).

Table 11-2 defines technical terms specifically pertaining to TOE Qualified Signature Creation Device (QSCD) applet and used throughout this security target. Wherever applicable, terms are associated with the related acronyms.

Table 11-2 Technical terms pertaining to the TOE QSCD applet

Term	Acronym	Description
Administrator		User in charge of performing QSCD preparation (cf. section 2.3.4) and other administrative operations of a QSCD.
Advanced electronic signature		Digital signature which meets specific requirements in [R13], article 2.2. *Note: According to [R13], a digital signature qualifies as an advanced electronic signature if it:* • *is uniquely linked to the Signatory;* • *is capable of identifying the Signatory;* • *is created using means that the Signatory can maintain under his sole control, and* • *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		Digital signature used as electronic attestation binding an SVD to a person and confirming the identity of that person as legitimate signer ([R13], article 2.9).
Certificate info		Information associated with an SCD/SVD pair that may be stored in a QSCD. *Note: Certificate info is either:* • *a signer's public key certificate, or* • *one or more hash values of a signer's public key certificate, together with an identifier of the hash function used to compute the hash values.* *Certificate info may be combined with information to allow the user to distinguish between several certificates.*
Certificate Generation Application	CGA	Collection of application components that receive the SVD from a QSCD in order to generate a certificate and create a digital signature of the certificate.
Certification Service Provider	CSP	Entity that issues certificates or provides other services related to electronic signatures ([R13], article 2.11).

Term	Acronym	Description
Data To Be Signed	DTBS	All electronic data to be signed, including a user message and signature attributes.
Data To Be Signed or its unique Representation	DTBS/R	Data received by a QSCD as input in a single signature creation operation. *Note: DTBS/R is either:* • *a hash value of the data to be signed (DTBS), or* • *an intermediate hash value of a first part of the DTBS complemented with a remaining part of the DTBS, or* • *the DTBS.*
Human Interface Device	HID	Human interface provided by the SCA for user authentication.
Legitimate User		User of a QSCD who gains possession of it from a QSCD provisioning service provider and who can be authenticated by the QSCD as its Signatory.
Middleware Developer		Actor that implements the CGA and the SCA.
Qualified certificate		Public key certificate that meets the requirements laid down in [R13], Annex I and that is provided by a CSP that fulfils the requirements laid down in [R13], Annex II ([R13], article 2.10).
Qualified electronic signature		Advanced electronic signature that has been created with a QSCD with a key certified with a qualified certificate ([R13], article 5.1).
Reference Authentication Data	RAD	Data persistently stored by the TOE for authentication of a user as authorized for a particular role.
Signatory		Legitimate user of a QSCD associated with it in the certificate of the SVD and who is authorized by the QSCD to operate the signature creation function ([R13], article 2.3).
Signature attributes		Additional information that is signed together with a user message.
Signature Creation Application	SCA	Application complementing a QSCD with a user interface with the purpose to create an electronic signature. *Note: A signature creation application is software consisting of a collection of application components configured to:*

Term	Acronym	Description
		• *present the data to be signed (DTBS) for review by the Signatory,* • *obtain prior to the signature process a decision by the Signatory,* • *if the Signatory indicates by specific unambiguous input or action its intent to sign, send a DTBS/R to the TOE,* • *process the electronic signature generated by the QSCD as appropriate, e.g. as attachment to the DTBS.*
Signature Creation Data	SCD	Private cryptographic key stored in a QSCD under exclusive control by the Signatory to create an electronic signature ([R13], article 2.4).
Signature Creation System	SCS	Complete system that creates an electronic signature, consisting of an SCA and a QSCD.
Signature Verification Data	SVD	Public cryptographic key that can be used to verify an electronic signature ([R13], article 2.7).
Signed Data Object	SDO	Electronic data to which an electronic signature has been attached to or logically associated with as a method of authentication.
QSCD provisioning service		Service that prepares and provides a QSCD to a subscriber, and supports the Signatory with certification of generated keys and administrative functions of the QSCD.
User		Entity (human user or external IT entity) outside the TOE that interacts with the TOE.
User Message		Data determined by the Signatory as the correct input for signing.
Verification Authentication Data	VAD	Data provided as input to a QSCD for authentication by knowledge.

11.2 Abbreviations

AES	Advanced Encryption Standard
APDU	Application Protocol Data Unit
ASCII	American Standard Code for Information Interchange
BAC	Basic Access Control

BSI	Bundesamt für Sicherheit in der Informationstechnik
CC	Common Criteria
CCMB	Common Criteria Maintenance Board
CEN	Comité Européen de Normalisation
CGA	Certificate Generation Application
COS	Chip Operating System
CPS	Card Personalization Specification
CRC	Cyclic Redundancy Check
CSP	Certification Service Provider
DES	Data Encryption Standard
DF	Dedicated/Directory File
DFA	Differential Power Analysis
DTBS	Data To Be Signed
DTBS/R	Data To Be Signed Representation
EAL	Evaluation Assurance Level
ECC	European Citizen Card
EF	Elementary File
FID	File Identifier
FIPS	Federal Information Processing Standards
HID	Human Interface Device
IAS	Identification Authentication Signature
IC	Integrated Circuit
ICAO	International Civil Aviation Organization
IEC	International Electrotechnical Commission
IETF	Internet Engineering Task Force
ISO	International Organization for Standardization
IT	Information Technology
JIWG	Joint Interpretation Working Group
MAC	Message Authentication Code
MF	Master File
NIST	National Institute of Standards and Technology
OS	Operating System
OSP	Organizational Security Policy
PACE	Password Authenticated Connection Establishment

PKCS	Public-Key Cryptography Standards
PP	Protection Profile
PUC	Personal Unblocking Code
QSCD	Qualified Signature Creation Device
RAD	Reference Authentication Data
RFC	Request For Comments
RSA	Rivest-Shamir-Adleman
SAR	Security Assurance Requirement
SCA	Signature Creation Application
SCD	Signature Creation Data
SCS	Signature Creation System
SDO	Signed Data Object
SFP	Security Function Policy
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SPA	Simple Power Analysis
QSCD	Qualified Signature Creation Device
ST	Security Target
SVD	Signature Verification Data
TDES	Triple DES
TOE	Target Of Evaluation
TR	Technical Report
TSF	TOE Security Functionality
VAD	Verification Authentication Data

11.3 Technical references

[R1] **HID Global :** *initialization Guidance for ASapp-QSCDApplet, ref. TCAE160087*

[R2] **HID Global:** *Personalization Guidance for ASapp-QSCD Applet, ref. TCAE160084*

[R3] **HID Global:** *Operational User Guidance for ASapp-QSCD Applet ref. TCAE160075*

[R4] **TUV Rheinland Nederland B.V.: JCOP 3 SECID P60 CS (OSB), certification report, 2018-01-15, NSCIB-CC-98209-CR2**

[R5] **CCMB:** *Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, version 3.1, revision 4, September 2012, ref. CCMB-2012-09-001*

[R6] **CCMB:** *Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components, version 3.1, revision 4, September 2012, ref. CCMB-2012-09-002*

[R7] **CCMB:** *Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components, version 3.1, revision 4, September 2012, ref. CCMB-2012-09-003*

[R8] **CCMB:** *Common Methodology for Information Technology Security Evaluation, Evaluation methodology, version 3.1, revision 4, September 2012, ref. CCMB-2012-09-004*

[R9] **CEN:** *Protection Profiles for Secure Signature Creation Device, Part 2: Device with Key Generation, version 2.0.1, ref. BSI-CC-PP-0059-2009-MA-01, January 2012*

[R10] **CEN:** *Protection profiles for Secure Signature Creation Device, Part 4: Extension for device with key generation and trusted communication with certificate generation application, version 1.0.1, ref. BSI-CC-PP-0071-2012, November 2012*

[R11] **CEN:** *Protection profiles for Secure Signature Creation Device, Part 5: Extension for device with key generation and trusted communication with signature creation application, version 1.0.1, ref. BSI-CC-PP-0072-2012, November 2012*

[R12] **GlobalPlatform:** *Card Specification, version 2.2.1, January 2011*

[R13] **European Parliament:** ***Regulation No. 910/2014*** *on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC 23 July 2014*

[R14] **GIXEL:** *European Card for e-Services and National e-ID Applications, IAS ECC, Identification Authentication Signature European Citizen Card, Technical Specifications, version 1.0.1, March 2008*

[R15] **Oracle:** *Java Card 3 Platform, Virtual Machine Specification, Classic Edition, Version 3.0.4, September 2011*

[R16] **Oracle:** *Java Card 3 Platform, Runtime Environment Specification, Classic Edition, Version 3.0.4, September 2011*

[R17] **Oracle:** *Java Card 3 Platform, Application Programming Interface Specification, Classic Edition, Version 3.0.4, September 2011*

[R18] **ICAO:** *Doc 9303, Machine Readable Travel Documents, Part 11: Security Mechanisms for MRTDs, Seventh Edition, 2015*

[R19] **IETF Network Working Group:** *Request for Comments 2119, Key words for use in RFCs to Indicate Requirement Levels, March 1997*

[R20] ***NXP*: JCOP 3 SECID P60 CS (OSB) Security Target lite,** *Rev. 2.6 – 2017-10-19*, **NSCIB-CC-16-98209**

[R21] **ISO/IEC:** *International Standard 7816-4, Identification cards – Integrated circuit cards – Part 4: Organization, security and commands for interchange*

[R22] **ISO/IEC:** *International Standard 7816-9, Identification cards – Integrated circuit cards – Part 9: Commands for card management*

[R23] **JIWG:** *Joint Interpretation Library, Composite product evaluation for Smart Cards and similar devices, version 1.5, August 2015*

[R24] **NIST:** *FIPS PUB 46-3, Federal Information Processing Standards Publication, Data Encryption Standard (DES), October 1999*

[R25] **NIST:** *FIPS PUB 180-4, Federal Information Processing Standards Publication, Secure Hash Standard (SHS), March 2012*

[R26] **NIST:** *FIPS PUB 197, Federal Information Processing Standards Publication, Advanced Encryption Standard (AES), November 2001*

[R27] **RSA Laboratories:** *PKCS #1: RSA Cryptography Standard, version 2.2, October 2012*

[R28] **RSA Laboratories:** *PKCS #15: Cryptographic Token Information Syntax Standard, version 1.1, June 2000*

Appendix A Platform JCOP3

This section provides a statement of compatibility between this security target (Composite-ST) and the platform JCOP3 security target (Platform-ST). In some detail, they identify the elements of the Platform-ST being relevant for the composite TOE, map such elements to the corresponding ones of the Composite-ST, and provide a rationale for this mapping.

A.1 Platform Identification

The platform on which the TOE is based is a secure microcontroller P6022J-VB equipped with a multi-applications operating system Java Card 3.0.4, Software for implementing cryptographic operations on the Micro Controller (called Crypto Lib) and Software for implementing content management according to *GlobalPlatform* [R12].

This platform received a Common Criteria certification at the EAL5 assurance level augmented by AVA_VAN.5, ALC_DVS.2, ASE_TSS.2 and ALC_FLR.1 [R20][R4] with certification ID:

CC-16-99111-CR2

The platform's certificate is valid and up-to-date.

A.2 IC Developer Identification

The developer of the P6022J VB IC is NXP.

A.3 IC Manufacturer Identification

The manufacturer of the P6022J VB chip is NXP.

A.4 Operating System Developer Identification

The developer of the multi-applications operating system JCOP 3 implementing Java Card 3.0.4 is NXP.

END OF DOCUMENT