HID Global
viale Remo De Feo, 1
80022 Arzano (NA), ITALY

www.hidglobal.com

SOMA-c016 Machine Readable Electronic Document

Security Target
eIDAS QSCD Application

Common Criteria version 3.1 revision 5
Assurance Level EAL5+

Version	2.0
Date	2023-06-19
Reference	TCLE180024
Classification	PUBLIC

Table of Contents

Abbreviations and notations 12

Foreword 13

1. Introduction 14

1.1 ST overview 14

1.2 ST reference 14

1.3 TOE reference 15

1.4 Glossary 15

1.5 TOE overview 22

1.5.1 TOE type, usage, and major security features 22

1.5.2 Required non-TOE hardware/software/firmware 23

2. TOE description 25

2.1 TOE physical scope 25

2.2 TOE logical scope 26

2.2.1 Mutual authentication 28

2.2.2 Generation of SCD/SVD pairs 30

2.2.3 Signature creation with SCD 31

2.2.4 Decipherment of encrypted data 31

2.3 TOE life cycle 32

2.3.1 Phase 1: Development 34

2.3.2 Phase 2: Manufacturing 35

2.3.3 Phase 3: Personalization 37

2.3.4 Phase 4: Operational use 38

3. Conformance claims 40

3.1 Common Criteria conformance claim 40
3.2 Package conformance claim 40
3.3 Protection Profile conformance claim 40
3.4 Protection Profile conformance rationale 41
3.4.1 Terminology 41
3.4.2 Security problem definition 41
3.4.3 Security objectives for the TOE 43
3.4.4 Security objectives for the operational environment 44
3.4.5 Security functional requirements 45
3.4.6 Security assurance requirements 49
4. Security problem definition 50
4.1 Assets, users, and threat agents 50
4.2 Threats 51
4.2.1 Threats defined in the PPs 51
4.2.2 Threats added to those defined in the PPs 53
4.3 Organizational Security Policies 53
4.3.1 OSPs defined in the PPs 54
4.3.2 OSPs added to those defined in the PPs 55
4.4 Assumptions 55
4.4.1 Assumptions defined in the PPs 55
4.4.2 OSPs added to those defined in the PPs 56
5. Security objectives 57
5.1 Security objectives for the TOE 57
5.1.1 OT.Lifecycle_Security 57
5.1.2 OT.SCD/SVD_Auth_Gen 57

5.1.3 OT.SCD_Unique 57
5.1.4 OT.SCD_SVD_Corresp 58
5.1.5 OT.SCD_Secrecy 58
5.1.6 OT.Sig_Secure 58
5.1.7 OT.Sigy_SigF 58
5.1.8 OT.DTBS_Integrity_TOE 58
5.1.9 OT.EMSEC_Design 59
5.1.10 OT.Tamper_ID 59
5.1.11 OT.Tamper_Resistance 59
5.1.12 OT.TOE_QSCD_Auth 59
5.1.13 OT.TOE_TC_SVD_Exp 59
5.1.14 OT.TOE_TC_VAD_Imp 60
5.1.15 OT.TOE_TC_DTBS_Imp 60
5.1.16 OT.AC_Init 61
5.1.17 OT.AC_Pre-pers 61
5.1.18 OT.AC_Pers 61
5.1.19 OT.Abuse-Func 61
5.1.20 OT.Sigy_DecF 61
5.1.21 OT.DTBD_Integrity_TOE 62
5.1.22 OT.TOE_TC_DTBD_Imp 62
5.2 Security objectives for the operational environment 62
5.2.1 OE.SVD_Auth 62
5.2.2 OE.CGA_QCert 62
5.2.3 OE.DTBS_Intend 63
5.2.4 OE.Signatory 63

5.2.5 OE.Dev_Prov_Service....63
5.2.6 OE.CGA_QSCD_Auth....63
5.2.7 OE.CGA_TC_SVD_Imp....64
5.2.8 OE.HID_TC_VAD_Exp....64
5.2.9 OE.SCA_TC_DTBS_Exp....65
5.2.10 OE.DTBD_Intend....65
5.2.11 OE.DDA_TC_DTBD_Exp....66
6. Security objectives rationale....67
6.1 Coverage of security objectives....67
6.2 Sufficiency of security objectives....69
7. Extended components definition....76
7.1 Definition of family FPT_EMS....76
7.2 Definition of family FIA_API....77
7.3 Definition of family FMT_LIM....78
8. Security functional requirements....81
8.1 Class FCS: Cryptographic support....83
8.1.1 FCS_CKM.1....83
8.1.2 FCS_CKM.4....84
8.1.3 FCS_COP.1/Signature_Creation....85
8.1.4 FCS_COP.1/Data_Decipherment....85
8.2 Class FDP: User data protection....86
8.2.1 FDP_ACC.1/SCD/SVD_Generation....87
8.2.2 FDP_ACF.1/SCD/SVD_Generation....87
8.2.3 FDP_ACC.1/SVD_Transfer....88
8.2.4 FDP_ACF.1/SVD_Transfer....89

8.2.5 FDP_ACC.1/Signature_Creation 90
8.2.6 FDP_ACC.1/Data_Decipherment 90
8.2.7 FDP_ACF.1/Signature_Creation 90
8.2.8 FDP_ACF.1/Data_Decipherment 92
8.2.9 FDP_RIP.1 93
8.2.10 FDP_SDI.2/Persistent 93
8.2.11 FDP_SDI.2/DTBS 94
8.2.12 FDP_SDI.2/DTBD 94
8.2.13 FDP_DAU.2/SVD 95
8.2.14 FDP_UIT.1/DTBS 96
8.2.15 FDP_UIT.1/DTBD 96
8.3 Class FIA: Identification and authentication 97
8.3.1 FIA_UID.1 97
8.3.2 FIA_UAU.1 98
8.3.3 FIA_AFL.1/Signatory 100
8.3.4 FIA_AFL.1/Admin 101
8.3.5 FIA_AFL.1/Init 102
8.3.6 FIA_AFL.1/Pre-pers 103
8.3.7 FIA_AFL.1/Pers 104
8.3.8 FIA_API.1 104
8.4 Class FMT: Security management 105
8.4.1 FMT_SMR.1/QSCD 105
8.4.2 FMT_SMR.1/Init 105
8.4.3 FMT_SMR.1/Pre-pers 106
8.4.4 FMT_SMR.1/Pers 106

8.4.5 FMT_SMF.1 106
8.4.6 FMT_MOF.1 107
8.4.7 FMT_MSA.1/Admin 108
8.4.8 FMT_MSA.1/Signatory 108
8.4.9 FMT_MSA.2 109
8.4.10 FMT_MSA.3 109
8.4.11 FMT_MSA.4 110
8.4.12 FMT_MTD.1/Admin 110
8.4.13 FMT_MTD.1/Signatory 111
8.4.14 FMT_MTD.1/Init 111
8.4.15 FMT_MTD.1/Pre-pers 112
8.4.16 FMT_MTD.1/Pers 112
8.4.17 FMT_LIM.1 112
8.4.18 FMT_LIM.2 113
8.5 Class FPT: Protection of the TSF 113
8.5.1 FPT_EMS.1 113
8.5.2 FPT_FLS.1 114
8.5.3 FPT_PHP.1 115
8.5.4 FPT_PHP.3 115
8.5.5 FPT_TST.1 116
8.6 Class FTP: Trusted path/channels 117
8.6.1 FTP_ITC.1/SVD 117
8.6.2 FTP_ITC.1/VAD 118
8.6.3 FTP_ITC.1/DTBS 119
8.6.4 FTP_ITC.1/DTBD 119

8.6.5 FTP_ITC.1/Init 120
8.6.6 FTP_ITC.1/Pre-pers 121
8.6.7 FTP_ITC.1/Pers 122
9. Security assurance requirements 124
10. Security requirements rationale 126
10.1 Coverage of security functional requirements 126
10.2 Sufficiency of security functional requirements 129
10.3 Satisfaction of dependencies of security requirements 134
10.4 Rationale for security assurance requirements 138
11. TOE summary specification 140
12. References 146
12.1 Acronyms 146
12.2 Technical references 148
Appendix A Platform identification 151

List of Tables

Table 1-1 ST reference....14
Table 1-2 TOE reference....15
Table 1-3 Technical terms pertaining to the TOE on the whole....15
Table 1-4 Technical terms pertaining to the TOE eIDAS QSCD application....19
Table 2-1 TOE component delivery....25
Table 2-2 Mapping between QSCD roles and their credentials....28
Table 2-3 Identification of RAD, VAD, and PUC in terms of Signatory's credentials....29
Table 2-4 Legend for deliveries occurring between non-consecutive actors....32
Table 2-5 Identification of recipient actors for the guidance documentation of the TOE QSCD application....34
Table 3-1 Source of assumptions, threats, and OSPs....41
Table 3-2 Changes, additions, and deletions to the asset with respect to the PPs....42
Table 3-3 Changes, additions, and deletions to the threat agents with respect to the PPs....42
Table 3-4 Changes, additions, and deletions to the threats with respect to the PPs....42
Table 3-5 Changes, additions, and deletions to the OSPs with respect to the PPs....42
Table 3-6 Changes, additions, and deletions to the Assumptions with respect to the PPs....43
Table 3-7 Source of security objectives for the TOE....43
Table 3-8 Changes, additions, and deletions to the security objectives for the TOE with respect to the PPs....43

Table 3-9 Source of security objectives for the operational environement.....45
Table 3-10 Changes, additions, and deletions to the security objectives for the operational environment with respect to the PPs.....45
Table 3-11 Source of security functional requirements 46
Table 3-12 Changes, additions, and deletions to the security functional requirements with respect to the PPs.....47
Table 6-1 Mapping of the security problem definition to the security objectives for the TOE.....67
Table 6-2 Mapping of the security problem definition to the security objectives for the operational environment.....68
Table 8-1 Mapping of the security functional requirements to the PPs.....81
Table 8-2 Security attributes of subjects and objects for access control 86
Table 9-1 Security assurance requirements: EAL5 augmented with ALC_DVS.2 and AVA_VAN.5.....124
Table 10-1 Mapping of the security functional requirements to the security objectives for the TOE.....126
Table 10-2 Satisfaction of dependencies of security functional requirements 134
Table 10-3 Satisfaction of dependencies of security assurance requirements 138
Table 11-1 Implementation of the security functional requirements in the TOE....140

List of Figures

Figure 2-1 eIDAS QSCD application operations split by QSCD life cycle phase and role 27

Figure 2-2 Life cycle of the TOE QSCD application 33

Abbreviations and 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.

Acronyms

The term HID is an acronym for Human Interface Device, as described in section 12.1, used in the protection profiles for secure signature creation device [R8] [R9] [R10], and should not be confused with the name of the company HID Global.

Keywords

The words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” are to be interpreted as described in RFC 2119 [R27].

Foreword

This security target refers to the European Parliament Directive 1999/93/EC [R14] in accordance with the protection profiles EN 419211-2:2013 [R8], EN 419211-4:2013 [R9], EN 419211-5:2013 [R10] it declares conformance with (cf. Section 3.3). However, it also incorporates the requirements of the eIDAS Regulation (EU) No 910/2014 [R12] and the according Commission Implementing Decision (EU) 2016/650 [R13], repealing the Directive 1999/93/EC.

1. Introduction

1.1 ST overview

This document is the sanitized version of the document Security Target for SOMA-c016 Machine Readable Electronic Document [R18].

This security target defines the security requirements, as well as the scope of the Common Criteria evaluation, for the signature creation and data decipherment functionalities of SOMA-c016 Machine Readable Electronic Document.

The Target Of Evaluation (TOE) is the integrated circuit chip NXP N7121 with IC Dedicated Software and Crypto Library, the operating system SOMA-c016 and with an eIDAS Qualified Signature Creation Device (QSCD) application providing signature features and encrypted data decipherment feature. The signature features are compliant with the eIDAS Regulation (EU) No 910/2014 [R12] and the according Commission Implementing Decision (EU) 2016/650 [R13], repealing the European Parliament Directive 1999/93/EC [R14]. The eIDAS QSCD application can optionally be configured as a PKCS #15 application [R40].

This security target specifies the security requirements for the eIDAS QSCD application of the TOE.
Furthermore, the e-Document also supports:

- Basic Access Control (BAC) compliant with ICAO Doc 9303 [R25],
- Password Authenticated Connection Establishment (PACE) compliant with ICAO Doc 9303 [R25];
- Active Authentication (AA) compliant with ICAO Doc 9303 [R25];
- Extended Access Control (EAC) v1 compliant with BSI TR-03110 [R2] [R3].

which is addressed by another STs [R16] [R17].

1.2 ST reference

Table 1-1 ST reference

Title	Security Target for SOMA-c016 Machine Readable Electronic Document - QSCD Application - Public Version
Version	2.0
Authors	Giovanni LICCARDO, Roberta SODANO
Date	2023-06-19
Reference	TCLE180024

1.3 TOE reference

Table 1-2 TOE reference

TOE name	SOMA-c016 Machine Readable Electronic Document eIDAS QSCD Application
TOE version	4
TOE developer	HID Global
TOE identifier	SOMA-c016_4
TOE identification data	53h 4Fh 4Dh 41h 2Dh 63h 30h 31h 36h 5Fh 34h
IC security target	NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library (R1/R2/R3/R4), Security Target Lite Rev. 2.6 – 13 June 2022 [R35]
IC certification report	BSI-DSZ-CC-1136-V3-2022 [R1]

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

SOMA-c016_4

(ASCII encoding: 53h 4Fh 4Dh 41h 2Dh 63h 30h 31h 36h 5Fh 34h)

where:

- “SOMA-c016” is the TOE name,
- the underscore character is a separator, and
- “4” is the TOE version number.

The ASCII encoding of the TOE identifier constitutes the TOE identification data, located in the persistent memory of the chip. Instructions for reading these data are provided by the guidance documentation [R19] [R21] [R22].

1.4 Glossary

Table 1-3 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 1-3 Technical terms pertaining to the TOE on the whole

Term	Acronym	Definition
Card Manufacturer		Actor that equips the IC with contact-based and/or contactless interfaces, and embeds the IC into a

Term	Acronym	Definition
		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 IDentification, Authentication and trust Services	eIDAS	EU regulation on / a set of standards for electronic identification and trust services for electronic transactions in the European Single Market [R12].
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 an eIDAS 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 eIDAS QSCD application.
Embedded Software Developer		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 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		Actor that develops the integrated circuit, the IC Dedicated Software, and the guidance documentation associated with these TOE components (cf. section 2.3.1).
IC initialization data		Data defined by the Embedded Software Developer and stored into the IC persistent memory by the IC Manufacturer. Particularly, they include the initialization key (cf. section 2.3.2).
IC Manufacturer		Actor that produces the TOE integrated circuit, containing the IC Dedicated Software and the Embedded Software, and stores the IC

Term	Acronym	Definition
		initialization data into the IC persistent memory (cf. section 2.3.2).
ICAO 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 Machine Readable Travel Document (MRTD).
Initialization Agent		User in charge of performing the initialization of the TOE, particularly of writing TOE initialization data (cf. section 2.3.2).
Initialization data		Data defined by the Embedded Software Developer and stored into the IC persistent memory by either the IC Manufacturer or the Initialization Agent (cf. section 2.3.2). These data consist of IC initialization data and TOE initialization data.
Initialization key		Cryptographic key used by the Initialization Agent for mutual authentication with the TOE.
Integrated Circuit	IC	Electronic component designed to perform processing and/or memory functions. The e-Document's chip is an integrated circuit.
Machine Readable Travel Document	MRTD	Official document issued by a state or organization which is used by the holder for international travel (e.g. passport, visa, official document of identity) and which contains mandatory visual (eye-readable) data and a separate mandatory data summary, intended for global use, reflecting essential data elements capable of being machine-read [R24].
Password Authenticated Connection Establishment	PACE	A communication establishment protocol defined in [R25]. 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	Definition
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 1-4 and section 2.3.3).
Personalization key		Cryptographic key used by the Personalization Agent for mutual authentication with the TOE.
Pre-personalization Agent		User in charge of performing the pre-personalization of the TOE, particularly of writing pre-personalization data (cf. section 2.3.2).
Pre-personalization data		Data defined and stored into the IC persistent memory by the Pre-personalization Agent. Particularly, they include the personalization key (cf. section 2.3.2).
Pre-personalization key		Cryptographic key used by the Pre-personalization Agent for mutual authentication with the TOE.
Product information		Readable information about the product as a whole, such as TOE identification data and traceability information, stored into the IC persistent memory by the IC Manufacturer and possibly updated by the Initialization Agent (cf. section 2.3.2). However, TOE identification data cannot be modified after TOE delivery.
Qualified Signature Creation Device	QSCD	Configured software or hardware which is used to implement signature creation and encrypted data decipherment, and which meets the requirements laid down in [R12], Annex II.
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).
Secure Signature Creation Device	SSCD	Configured software or hardware which is used to implement signature creation and encrypted data decipherment covered by the PPs [R8] [R9] [R10].
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).

Term	Acronym	Definition
TOE initialization data		Data defined by the Embedded Software Developer and stored into the IC persistent memory by the Initialization Agent. Particularly, they include the pre-personalization key (cf. section 2.3.2).

Table 1-4 defines technical terms specifically pertaining to TOE eIDAS QSCD application and used throughout this security target. Wherever applicable, terms are associated with the related acronyms.

Table 1-4 Technical terms pertaining to the TOE eIDAS QSCD application

Term	Acronym	Definition
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 [R14], article 2.2. *Note: According to [R14], 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 ([R14], 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.*

Term	Acronym	Definition
		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 ([R14], article 2.11).
Data Decipherment Application	DDA	Application complementing a QSCD with a user interface with the purpose to decipher encrypted data.
Deciphered Data Object	DDO	Data derived from decipherment of DTBD
Data To Be Deciphered	DTBD	All electronic data to be deciphered.
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		Set of software applications, particularly comprising the CGA and the SCA, meant for being used by the Administrator and/or the Signatory to interact with a QSCD during the operational use phase (cf. section 2.3.4).
Middleware Developer		Actor that implements the middleware.
Qualified certificate		Public key certificate that meets the requirements laid down in [R14], Annex I and that is provided by a CSP that fulfils the requirements laid down in [R14], Annex II ([R14], article 2.10).
Qualified electronic signature		Advanced electronic signature that has been created with a QSCD with a key certified with a qualified certificate ([R14], article 5.1).

Term	Acronym	Definition
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 ([R14], article 2.3) and encrypted data decipherment functions.
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:* • *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 ([R14], article 2.4) and to decipher encrypted data.
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 ([R14], 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.

Term	Acronym	Definition
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.

1.5 TOE overview

1.5.1 TOE type, usage, and major security features

The TOE is a combination of hardware and software configured to securely create, use, and manage 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 and deciphered data.

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.
7. if in an operational state, to decipher encrypted 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 decipher,
    c. receive data to be deciphered (DTBD) from the DDA over a trusted channel,

d. apply an appropriate cryptographic decipher function to the DTBD 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.

1.5.2 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 decipherment environment, where it interacts with the signer through a Data Decipherment Application (DDA) to decipher data after authenticating the signer as its Signatory. The DDA provides the data to be deciphered as input to the TOE data

decipherment function, and obtains the resulting deciphered data. The TOE and the DDA communicate through a trusted channel to ensure the integrity of the DTBD;

- 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), a Signature Creation Application (SCA) and a Data Decipherment Application (DDA) 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 N7121 equipped with IC Dedicated Software (cf. Appendix A for more details);
- smart card operating system SOMA-c016;
- an eIDAS Qualified Signature Creation Device (QSCD) application compliant with the eIDAS Regulation (EU) No 910/2014 [R12] and the according Commission Implementing Decision (EU) 2016/650 [R13], repealing the European Parliament Directive 1999/93/EC [R14];
- guidance documentation about the initialization of the TOE and the preparation and use of the eIDAS QSCD application, composed by:
  - the Initialization Guidance [R19];
  - the Pre-personalization Guidance [R20];
  - the Personalization Guidance [R21];
  - the Operational User Guidance [R22].

Table 2-5 identifies, for each guidance document, the actors involved in TOE life cycle who are the intended recipients of that document.

Table 2-1 described the format and delivery method of each TOE components:

Table 2-1 TOE component delivery

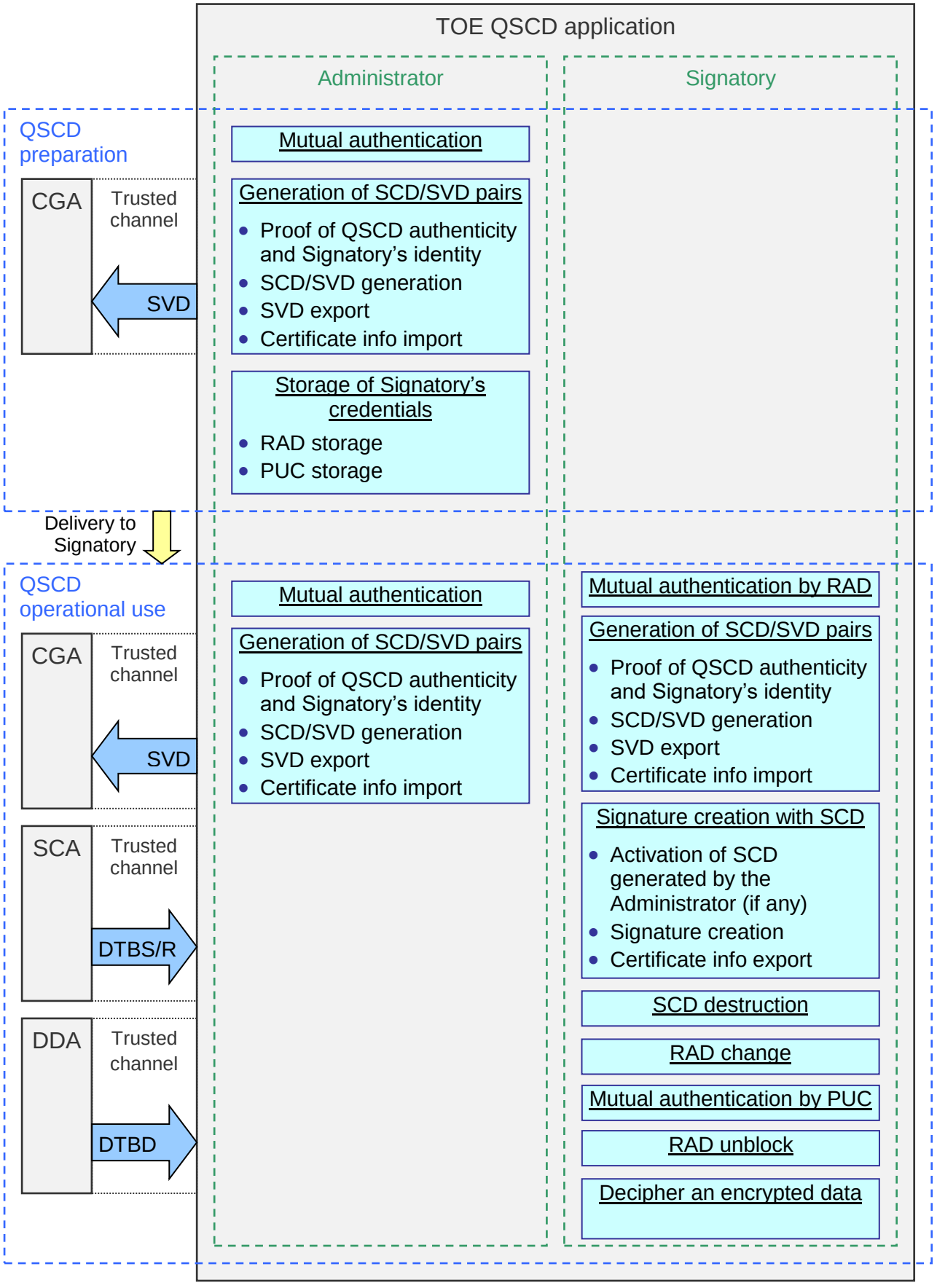
Type	TOE component	Format	Delivery method
IC	NXP N7121	Smart Card	Secure courier
OS and eIDAS QSCD Application	SOMA-c016 Machine Readable Electronic Document	HEX file	Secure IC Manufacturer's Web application
Document	Preparative and operational guidance	pdf/docx	Encrypted email message

The delivery procedure for the TOE is described in detail [R23].

2.2 TOE logical scope

The eIDAS application of the TOE supports the same QSCD life cycle phases, i.e. Q*SCD preparation* and Q*SCD operational use*, as well as the same QSCD roles, i.e. *Administrator* and *Signatory*, as those defined in the PPs [R8] [R9] [R10].
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 eIDAS QSCD application 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 under the MF by means of a PACE authentication compliant with ICAO Doc 9303 [R25];
2. external authentication under the QSCD application 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 [R25] are supported for PACE authentication. All the encoding methods specified in PKCS #15 [R40] 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 or the import of DTBD from the DDA upon data decipherment, shall be executed over the trusted channel compliant with ICAO Doc 9303 [R25] opened by means of PACE authentication.

Table 2-2 identifies the credentials associated to either of the QSCD roles, through which they can perform their respective mutual authentication procedures.

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

- The Administrator shall perform:
  1. PACE authentication under the MF using Administrator's PACE key;
  2. password verification under the QSCD application using Administrator's password.
- The Signatory shall perform:
  1. PACE authentication under the MF using Signatory's PACE key, which shall be derived from the selected encoding of Signatory's password #1 by means of the key derivation function defined in ICAO Doc 9303 [R25];
  2. password verification under the QSCD application using:
     - Signatory's password #2 for gaining access to ordinary operations;
     - Signatory's password #3 for gaining access to RAD unblock.

Table 2-3 identifies, for each of the Signatory's authentication secrets provided for by the PPs [R8] [R9] [R10], i.e. the RAD, the VAD, and the PUC[1], the Signatory's credentials of which it is comprised.

Table 2-3 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.

Signatory's password #1 is used just as a seed for key derivation, viz. Signatory's PACE key is precomputed and directly stored into the IC persistent memory. As a result, the length of Signatory's password #1 is not constrained, whereas the length of Administrator's password, Signatory's password #2, and Signatory's password #3 shall be comprised between 4 and 8 bytes.

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

In addition to the mutual authentication mechanisms associated to the roles defined in the PPs, a further such mechanism, consisting of a mutual authentication with a Middleware's PACE key, is available for the purposes of middleware integration. This key is the same for each QSCD, is shared between the QSCDs and the middleware, and enables read access to product information, certificate info, and PKCS #15 files (if present).

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 phase on the part of both the Administrator and the Signatory. SCD keys are activated for signature creation or data decipherment 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 [R40].

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 [R15] 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 [R39] of 2048, 3072 or 4096 bits.

In accordance with IAS ECC specification [R15], the QSCD application supports signature creation algorithm RSASSA-PKCS1-v1_5 compliant with PKCS #1 [R39] for Client/Server Authentication, with keys of 2048, 3072 or 4096 bits. The signature creation algorithm RSASSA-PSS is supported as well with same key lengths. The hash function (i.e. SHA-256 compliant with [R33]) is expected to be applied by the terminal before the message to be signed is sent to the QSCD application (cf. [R15]).

2.2.3 Signature creation with SCD

In accordance with IAS ECC specification [R15], the QSCD application supports digital signature creation with signature creation algorithm RSASSA-PKCS1-v1_5 compliant with PKCS #1 [R39]. The signature creation algorithm RSASSA-PSS [R39] is supported as well. In both cases the hash algorithm is SHA-256 compliant with FIPS PUB 180-4 [R33], and keys of 2048, 3072 or 4096 bits are supported.

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 digital signatures to the SCA must be executed over the same trusted channel.

2.2.4 Decipherment of encrypted data

In accordance with IAS ECC specification [R15], the QSCD application supports encryption data decipherment with encryption data decipherment method algorithm RSASSA-PKCS1-v1_5 compliant with PKCS #1 [R39] and keys of 2048, 3072 or 4096 bits.

The decipherment function of the QSCD application can take encrypted data (DTBD) as input from the DDA.

Decryption operation 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 DTBD integrity upon import of the DTBD from the DDA.
The decrypted data is returned to the DDA after the correct execution of the operation.

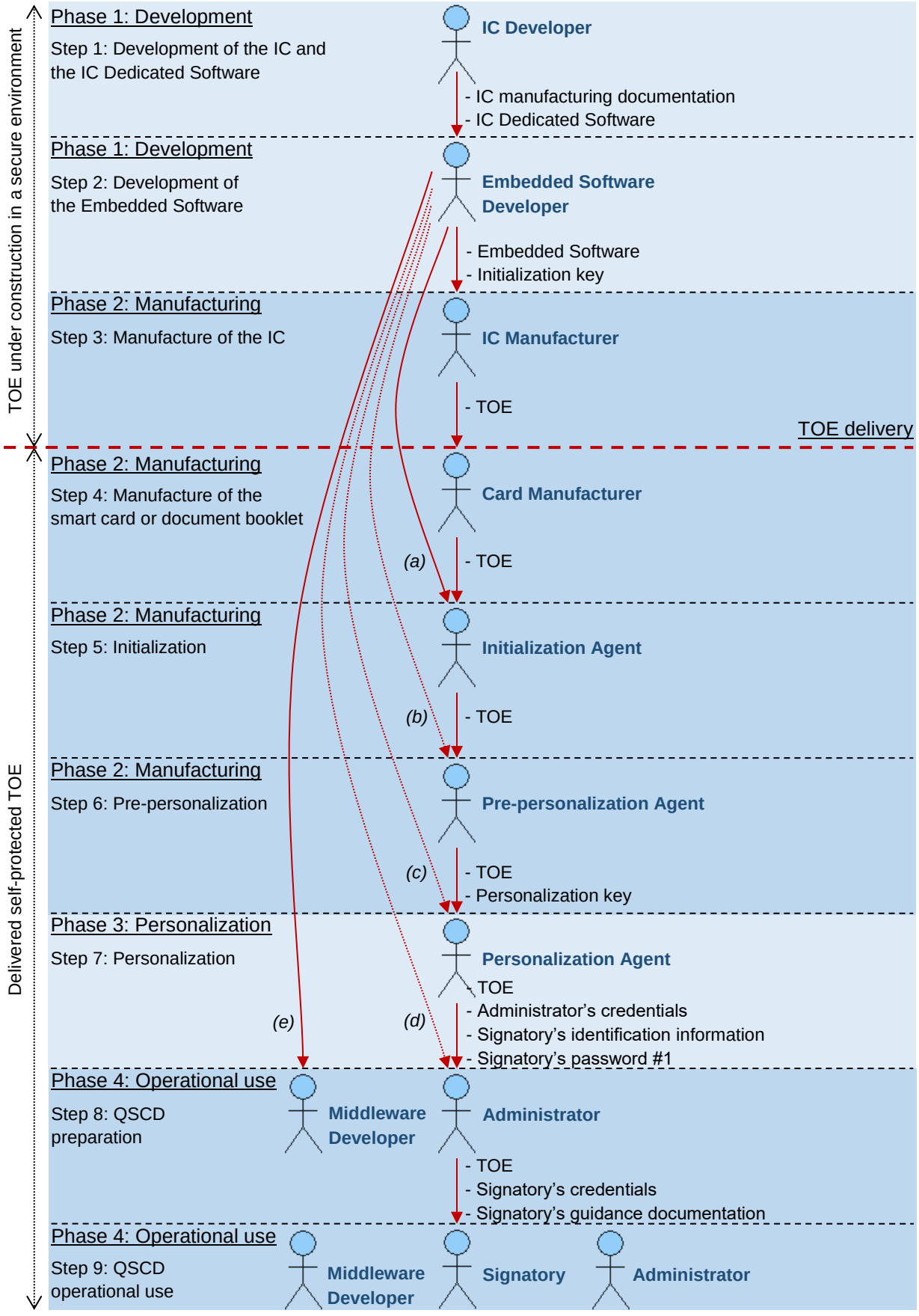
2.3 TOE life cycle

The TOE life cycle is comprised of four life cycle phases, i.e. *development*, *manufacturing*, *personalization*, and *operational use*. With regard to the life cycle of the QSCD application, these phases can be split into nine steps. The last two steps, which take place when the TOE stands in the operational use phase, match the QSCD life cycle phases defined in the PPs [R8] [R9] [R10], i.e. *QSCD preparation* and *QSCD operational use*.
Figure 2-2 represents the life cycle of the TOE QSCD application. Particularly, it illustrates the correspondence between the life cycle phases of the TOE and the life cycle phases of the QSCD application as defined in the PPs, and 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 occurring between non-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 2-4.

Table 2-4 Legend for deliveries occurring between non-consecutive actors

Delivery	Delivered items
(a)	• Initialization cryptograms • Initialization guidance
(b)	• Pre-personalization key • Pre-personalization guidance
(c)	• Personalization guidance • Middleware’s credentials
(d)	• Operational user guidance
(e)	• Operational user guidance • Middleware’s credentials

Figure 2-2 Life cycle of the TOE QSCD application



Detailed information about the operations available in each life cycle phase of the TOE is provided in the guidance documentation of the TOE QSCD application [R19] [R20] [R21] [R22]. Table 2-5 identifies, for each guidance document, the actors who are the intended recipients of that document.

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

Guidance document	Recipient actors
Initialization guidance	Initialization Agent
Pre-personalization guidance	Pre-personalization 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: Development of the IC and the IC Dedicated Software

The **IC Developer** develops the integrated circuit, the IC Dedicated Software, and the guidance documentation associated with these TOE components.

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

- the IC manufacturing documentation;
- the IC Dedicated Software.

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 Dedicated Software and develops the Embedded Software, as well as the guidance documentation.

Furthermore, the **Embedded Software Developer** generates the initialization key and the pre-personalization key, and makes use of the former key to encrypt the latter one, as well as (optionally) some bitmaps encoding product information and/or configuration data.

In addition, the **Embedded Software Developer** generates middleware's credentials. They comprise Middleware's PACE key, granting read access during operational use (cf. section 2.2.1), and possibly also a middleware's secret to be used in addition to users' passwords in the derivation of Administrator's and Signatory's PACE keys.

Finally:

- the Embedded Software and the initialization key are securely delivered to the **IC Manufacturer**;
- the cryptograms enciphered using the initialization key are securely delivered to the **Initialization Agent**;
- the pre-personalization key is securely delivered to either the **Initialization Agent** or the **Pre-personalization Agent**;
- middleware's credentials are securely delivered to both the **Middleware Developer**, and either the **Initialization Agent** or the **Personalization Agent**.

As regards TOE guidance documentation, the operational user guidance is securely shared with the **Middleware Developer**; moreover, either all documents are securely delivered to the **Initialization Agent**, or each document is securely delivered to the recipient actors as identified in Table 2-5.

2.3.2 Phase 2: Manufacturing

Step 3: Manufacture of the IC

The **IC Manufacturer** produces the TOE integrated circuit, containing the IC Dedicated Software and the Embedded Software, and creates in the IC persistent memory the high-level objects relevant for the QSCD application.
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, in the form of an IC not yet embedded, from the IC Manufacturer to the Card Manufacturer. That is to say, this is the event upon which the construction of the TOE in a secure environment ends, and the TOE begins to be self-protected.*

Step 4: Manufacture of the smart card or document booklet

The **Card Manufacturer** equips the IC with contact-based and/or contactless interfaces, and embeds the IC into a smart card or a document booklet.

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

Step 5: Initialization

The **Initialization Agent** sends the encrypted product information and/or configuration data (if any), as well as the encrypted pre-personalization key, to the TOE, which deciphers the cryptograms using the initialization key, verifies the correctness of the resulting plaintexts, and stores the data into persistent memory.

Finally, the TOE is securely delivered to the **Pre-personalization Agent**, along with the pre-personalization key if it was delivered to the **Initialization Agent** rather than directly to the **Pre-personalization Agent**. Likewise, in case middleware's credentials were delivered to the **Initialization Agent** rather than directly to the **Personalization Agent**, they are securely delivered to either the **Pre-personalization Agent** or the **Personalization Agent**.

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 **Pre-personalization Agent**, or each document is securely delivered to the recipient actors as identified in Table 2-5.

Step 6: Pre-personalization

The **Pre-personalization Agent** generates the personalization key, then creates/modifies in the IC persistent memory the high-level objects relevant for the QSCD application. Particularly:

- The pre-personalization key is overwritten with the personalization key;
- The DIR file, if present, is compliant with ISO/IEC 7816-4 [R28].

Finally, the TOE and the personalization key are securely delivered to the **Personalization Agent**. Furthermore, if middleware's credentials were delivered to the **Pre-personalization Agent** rather than directly to the **Personalization Agent**, they are securely delivered to the **Personalization Agent**.

As regards TOE guidance documentation, if the **Pre-personalization 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 2-5.

2.3.3 Phase 3: Personalization

Step 7: 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. Both Administrator's and Signatory's PACE keys are possibly derived employing, in addition to users' passwords, a middleware's secret included in the middleware's credentials received from the **Embedded Software Developer**.

Then, the **Personalization Agent** creates/modifies in the IC persistent memory the high-level objects relevant for the QSCD application.
Particularly:

- Administrator's PACE key object, Signatory's PACE key object, and Administrator's password object are filled with the generated credentials, while Middleware's PACE key object is filled with the key included in the middleware's credentials received from the **Embedded Software Developer**.
- 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 and encrypted data decipherment 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 [R40], 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).

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 **Administrator**.

2.3.4 Phase 4: Operational use

Step 8: QSCD preparation

The **Administrator** generates the remaining Signatory's credentials, i.e.:

- Signatory's password #2;
- Signatory's password #3.

Then, the **Administrator** is required/allowed to modify in the IC persistent memory the high-level objects relevant for the QSCD application.
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 **Administrator** shall fill Signatory's password #2 and password #3 objects with the generated credentials, and shall update the AODF file (if present) to indicate that these passwords have been initialized.

Finally, the TOE is securely delivered to the intended **Signatory**, along with Signatory's credentials and appropriate Signatory's guidance documentation produced by the QSCD provisioning service.

Step 9: 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.
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 and data decipherment 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 and data decipherment for the private keys generated by the **Administrator**;
- create digital signatures using the available signature creation private keys;
- perform decipherment of encrypted data using the available decipherment private keys;
- destroy signature creation private keys;
- change or unblock Signatory's password #2.

In addition to a mutual authentication by means of the **Administrator**'s or **Signatory**'s PACE keys, read access to product information, certificate info, and PKCS #15 files (if present) is also granted by a mutual authentication by means of the Middleware's PACE key (cf. section 2.2.1).

3. Conformance claims

3.1 Common Criteria conformance claim

This security target claims conformance to:

- Common Criteria version 3.1 revision 5 [R5] [R6] [R7], as follows:
  - Part 2 (security functional requirements) extended,
  - Part 3 (security assurance requirements) conformant.

The applet runs on the platform NXP N7121. This platform is certified against Common Criteria at the assurance level EAL6+ (cf. Appendix A).

3.2 Package conformance claim

This security target claims conformance to Evaluation Assurance Level EAL5, 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, EN 419211-2:2013 (certificate BSI-CC-PP-0059-2009-MA-02) [R8];
- 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, EN 419211-4:2013 (certificate BSI-CC-PP-0071-2012-MA-01) [R9];
- 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, EN 419211-5:2013 (certificate BSI-CC-PP-0072-2012-MA-01) [R10].

3.4 Protection Profile conformance rationale

3.4.1 Terminology

In this ST the term QSCD replaces all occurrences of the term SSCD referred to in the PPs;

3.4.2 Security problem definition

The source of threats, organizational security policies and assumptions is specified in Table 3-1.

Table 3-1 Source of assumptions, threats, and OSPs

	Source		
	PP Part 2 [R8]	**PP Part 4 [R9]**	**PP Part 5 [R10]**
Threats	• T.SCD_Divulg • T.SCD_Derive • T.Hack_Phys • T.SVD_Forgery • T.SigF_Misuse • T.DTBS_Forgery • T.Sig_Forgery	All threats of the PP Part 2 [R8] This PP does not define any additional threats.	All threats of the PP Part 2 [R8] This PP does not define any additional threats.
Organizational Security Policies	• P.CSP_QCert • P.QSign • P.Sigy_QSCD • P.Sig_Non-Repud	All OSP of the PP Part 2 [R8] This PP does not define any additional OSP.	All OSP of the PP Part 2 [R8] This PP does not define any additional OSP.
Assumptions	• A.CGA • A.SCA	All Assumptions of the PP Part 2 [R8] This PP does not define any additional assumptions.	All Assumptions of the PP Part 2 [R8] This PP does not define any additional assumptions.

Changes, additions, and deletions to asset, threat agents, threats, OSPs and assumptions with respect to the PPs (cf. section 3.3) are listed in Table 3-2, Table 3-3, Table 3-4, Table 3-5 and Table 3-6.

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

Asset	Difference	Rationale
SCD	Change	Refined to add the capability to decipher encrypted data
SVD	Change	Refined to add the capability to encipher plain data
DTBD	Addition	Added to take in account data to be deciphered. Their integrity must be maintained.

Table 3-3 Changes, additions, and deletions to the threat agents with respect to the PPs

Threat Agent	Difference	Rationale
Attacker	Change	Refined to add the capability to alter data to be deciphered

Table 3-4 Changes, additions, and deletions to the threats with respect to the PPs

Threat	Difference	Rationale
T.SCD_Divulg	Change	Refined to add the use of the SCD for encrypted data decipherment
T.Hack_Phys	Change	Refined to add the asset DTBD
T.DecF_Misuse	Addition	Added to take in account the misuse of decipherment function
T.DTBD_Forgery	Addition	Added to take in account the counterfeiting of DTBD
T.Dec_Forgery	Addition	Added to take in account the counterfeiting of DDO
T.Abuse-Func	Addition	Added to cover the relevant platform threat T.Abuse-Func.

Table 3-5 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).
P.Sigy_QSCD	Change	Refined to add the use of the SCD for encrypted data decipherment
P.Dec_Integrity	Addition	Added to take in account of the integrity of the decrypted data

Table 3-6 Changes, additions, and deletions to the Assumptions with respect to the PPs

Assumption	Difference	Rationale
A.DDA	Addition	Added to specify the assumption that is made on the operational environment in order to guarantee that the DTBD corresponds to the data that the Signatory wishes to decipher. This assumption is in addition to A.SCA included in the PPs, regarding the SCA.

3.4.3 Security objectives for the TOE

The source of the security objectives for the TOE is specified in Table 3-7.

Table 3-7 Source of security objectives for the TOE

	Source		
	PP Part 2 [R8]	**PP Part 4 [R9]**	**PP Part 5 [R10]**
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

Changes, additions, and deletions to the security objectives for the TOE with respect to the PPs (cf. section 3.3) are listed in Table 3-8.

Table 3-8 Changes, additions, and deletions to the security objectives for the TOE with respect to the PPs

Security objective	Difference	Rationale
OT.SCD_Unique	Change	Refined to add the use of the SCD for encrypted data decipherment

Security objective	Difference	Rationale
OT.SCD_SVD_Corresp	Change	Refined to add the use of the SCD for encrypted data decipherment
OT.SCD_Secrecy	Change	Refined to add the use of the SCD for encrypted data decipherment. The corresponding application note was refined too.
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_Pre-pers	Addition	Added to specify the access control to be enforced by the TOE as regards the storage of pre-personalization 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).
OT.Abuse-Func	Addition	Added to cover the relevant platform security objectives for the TOE O.Abuse-Func and O.NVM_INTEGRITY.
OT.Sigy_DecF	Addition	Added to take in account the use of the SCD for encrypted data decipherment
OT.DTBD_Integrity_TOE	Addition	Added to preserve the integrity of the DTBD
OT.TOE_TC_DTBD_Imp	Addition	Added to avoid alteration of the DTBD

3.4.4 Security objectives for the operational environment

The source of the security objectives for the operational environment is specified in Table 3-9.

PP Part 4 [R9] replaces (~~strikethrough~~ text) OE.QSCD_Prov_Service from PP Part 2 [R8] 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 [R10] replaces (~~strikethrough~~ text) OE.HID_VAD from PP Part 2 [R8] with OE.HID_TC_VAD_Exp, and OE.DTBS_Protect from PP Part 2 with OE.SCA_TC_DTBS_Exp.

Table 3-9 Source of security objectives for the operational environement

	Source		
	PP Part 2 [R8]	**PP Part 4 [R9]**	**PP Part 5 [R10]**
Security objectives for the operational environment	• OE.SVD_Auth • OE.CGA_QCert • ~~OE.QSCD_Prov_Service~~ • ~~OE.HID_VAD~~ • OE.DTBS_Intend • ~~OE.DTBS_Protect~~ • 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

Changes, additions, and deletions to the security objectives for the operational environment with respect to the PPs (cf. section 3.3) are listed in Table 3-10.

Table 3-10 Changes, additions, and deletions to the security objectives for the operational environment with respect to the PPs

Security objective	Difference	Rationale
OE.DTBD_Intend	Addition	Added to specify the need that the DTBD corresponds to the data that the Signatory wishes to decipher. This security objective is in addition to OE.DTBS_Intend included in the PPs, regarding the DTBS/R.
OE.DDA_TC_DTBD_Exp	Addition	Added to specify the need that the DTBD cannot be altered undetected in transit between the DDA and the TOE. This security objective is in addition to OE.SCA_TC_DTBS_Exp included in the PPs, regarding the DTBS.

3.4.5 Security functional requirements

The source of the security functional requirements is specified in Table 3-11.

Table 3-11 Source of security functional requirements

	Source		
	PP Part 2 [R8]	**PP Part 4 [R9]**	**PP Part 5 [R10]**
Security objectives for the TOE	• FCS_CKM.1 • FCS_CKM.4 • FCS_COP.1 • FDP_ACC.1/SCD/SVD_Generation • FDP_ACF.1/SCD/SVD_Generation • FDP_ACC.1/SVD_Transfer • FDP_ACF.1/SVD_Transfer • FDP_ACC.1/Signature_Creation • FDP_ACF.1/Signature creation • FDP_RIP.1 • FDP_SDI.2/Persistent • FDP_SDI.2/DTBS • FIA_UID.1 • FIA_UAU.1 • FIA_AFL.1 • FMT_SMR.1 • FMT_SMF.1 • 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 • FPT_EMS.1 • FPT_FLS.1 • FPT_PHP.1	• FIA_UAU.1 (extends [R8]) • FIA_API.1 • FDP_DAU.2/SVD • FTP_ITC.1/SVD	• FIA_UAU.1 (extends [R8]) • FDP_UIT.1/DTBS • FTP_ITC.1/VAD • FTP_ITC.1/DTBS

	• FPT_PHP.3 • FPT_TST.1		

Changes, additions, and deletions to the security functional requirements with respect to the PPs (cf. section 3.3) are listed in Table 3-12.

Table 3-12 Changes, additions, and deletions to the security functional requirements with respect to the PPs

SFR	Difference	Rationale
FCS_COP.1/Signature_Creation	Change	Iteration performed on PP SFR FCS_COP.1 due to the introduction of further iterations, related to the cryptographic operations supported by the TOE in addition to those addressed in the PPs (cf. below).
FCS_COP.1/Data_Decipherment	Addition	Added to cover encrypted data decipherment, supported by the TOE in addition to the cryptographic operations addressed in the PPs.
FDP_ACC.1/Data_Decipherment	Addition	Added to cover encrypted data decipherment, supported by the TOE in addition to the access controls addressed in the PPs.
FDP_ACF.1/Data_Decipherment	Addition	Added to cover encrypted data decipherment, supported by the TOE in addition to the attribute based access controls addressed in the PPs.
FDP_SDI.2/DTBD	Addition	Added to cover encrypted data integrity stored in the TOE
FDP_UIT.1/DTBD	Addition	Added to cover encrypted data exchange integrity.
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 19).
FIA_AFL.1/Signatory	Change	Iteration performed on PP SFR FIA_AFL.1 due to the introduction of further iterations, related to the authentication mechanisms supported by the TOE in addition to those addressed in the PPs (cf. below). Moreover, a refinement has been performed to specify that the SFR refers to consecutive failed authentication attempts with respect to the RAD.
FIA_AFL.1/Admin	Addition	Added to cover authentication with respect to the Administrator's credentials, supported by

SFR	Difference	Rationale
		the TOE in addition to the authentication mechanisms addressed in the PPs.
FIA_AFL.1/Init	Addition	Added to cover authentication with respect to the initialization key, supported by the TOE in addition to the authentication mechanisms addressed in the PPs (cf. section 2.3.2).
FIA_AFL.1/Pre-pers	Addition	Added to cover authentication with respect to the pre-personalization key, supported by the TOE in addition to the authentication mechanisms addressed in the PPs (cf. section 2.3.2).
FIA_AFL.1/Pers	Addition	Added to cover authentication with respect to the personalization key, supported by the TOE in addition to the authentication mechanisms addressed in the PPs (cf. section 2.3.3).
FMT_SMR.1/QSCD	Change	Iteration performed on PP SFR FMT_SMR.1 due to the introduction of further iterations, related to the roles supported by the TOE in addition to those specified in the PPs (cf. below).
FMT_SMR.1/Init	Addition	Added to cover the Initialization Agent role, supported by the TOE in addition to the roles specified in the PPs (cf. section 2.3.2).
FMT_SMR.1/Pre-pers	Addition	Added to cover the Pre-personalization Agent role, supported by the TOE in addition to the roles specified in the PPs (cf. section 2.3.2).
FMT_SMR.1/Pers	Addition	Added to cover the Personalization Agent role, supported by the TOE in addition to the roles specified in the PPs (cf. section 2.3.3).
FMT_MOF.1	Change	Refined to add data decipherment function to R.Sigy.
FMT_MSA.1/Signatory	Change	Refined to add the ability to modify security attributes for data decipherment function to R.Sigy.
FMT_MSA.3	Change	Refined to add the ability to provide default values for security attributes for data decipherment function.
FMT_MTD.1/Init	Addition	Added to specify the requirements to be enforced by the TOE as regards the management of TOE initialization data (cf. section 2.3.2).
FMT_MTD.1/Pre-pers	Addition	Added to specify the requirements to be enforced by the TOE as regards the management of pre-personalization data (cf. section 2.3.2).

SFR	Difference	Rationale
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).
FMT_LIM.1	Addition	Added to cover the relevant platform SFR FMT_LIM.1 (cf. Appendix B).
FMT_LIM.2	Addition	Added to cover the relevant platform SFR FMT_LIM.2 (cf. Appendix B).
FTP_ITC.1/DTBD	Addition	Added to account for the additional trusted channel supported by the TOE to DDA.
FTP_ITC.1/Init	Addition	Added to account for the additional trusted channel supported by the TOE for the import of TOE initialization data (cf. section 2.3.2).
FTP_ITC.1/Pre-pers	Addition	Added to account for the additional trusted channel supported by the TOE for the import of pre-personalization data (cf. section 2.3.2).
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).

3.4.6 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 EAL5 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.

Assets and objects:

The PPs [R8] [R9] [R10] share the same assets, reported here below. The definition of SCD and SVD has been extended.

1. SCD: private key used to perform an electronic signature operation and/or the decipherment of encrypted data. 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 and/or the encipherment of plain data. 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.

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

4. DTBD: encrypted data which the Signatory intends to decipher. Their integrity must be maintained.

Users and subjects acting for users:

The PPs [R8] [R9] [R10] share the same users, reported here below.

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), 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. Pre-personalization Agent: user in charge of performing step 6, pre-personalization, of TOE life cycle (cf. section 2.3.2), particularly of writing pre-personalization data. The subject S.Pre-pers is acting in the role R.Pre-pers for this user after successful authentication as Pre-personalization Agent.
6. Personalization Agent: user in charge of performing step 7, personalization, of TOE life cycle (cf. section 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:

The PPs [R8] [R9] [R10] share the same threat agents, reported here below. The definition of Attacker has been extended.

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

4.2 Threats

4.2.1 Threats defined in the PPs

The PPs [R8] [R9] [R10] share the same threats, reported here below. The threats T.SCD_Divulg and T.SCD_Derive have been extended.

4.2.1.1 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, use for signature creation and encrypted data decipherment in the TOE.

4.2.1.2 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 encrypted data deciphered by means of the SCD or any other data exported outside the TOE, which is a threat against the secrecy of the SCD.

4.2.1.3 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, DTBS, DTBD.

4.2.1.4 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.

4.2.1.5 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.

4.2.1.6 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.

4.2.1.7 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.

4.2.2 Threats added to those defined in the PPs

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

4.2.2.1 T.DecF_Misuse

Misuse of the decipherment function of the TOE

An attacker misuses the decipherment function of the TOE to create a DDO for data the Signatory has not decided to decipher. 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.

4.2.2.2 T.DTBD_Forgery

Forgery of the DTBD

An attacker modifies the DTBD sent by the DDA. Thus the DTBD used by the TOE for decipherment does not match the DTBD that the Signatory intended to decipher.

4.2.2.3 T.Dec_Forgery

Forgery of the deciphered data

An attacker forges an DDO, maybe using deciphered data which has been created by the TOE, and the violation of the integrity of the DDO is not detectable by the Signatory or by third parties. The deciphered data 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.

4.2.2.4 T.Abuse-Func

Abuse of functionality

An attacker may abuse functions of the TOE which may not be used after TOE delivery in order (i) to manipulate or disclose the user data stored in the TOE, (ii) to manipulate or disclose the TSF data stored in the TOE, or (iii) to manipulate (bypass, deactivate, or modify) the TSF.

4.3 Organizational Security Policies

4.3.1 OSPs defined in the PPs

The PPs [R8] [R9] [R10] share the same OSPs, reported here below. The OSP P.Sigy_QSCD has been extended.

4.3.1.1 P.CSP_QCert

Qualified certificates

The CSP uses a trustworthy CGA to generate a qualified certificate or non-qualified certificate ([R14], article 2, clause 9, 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.

4.3.1.2 P.QSign

Qualified electronic signatures

The Signatory uses a Signature Creation System to sign data with an advanced electronic signature ([R14], article 1, clause 2), which is a qualified electronic signature if it is based on a valid qualified certificate (according to [R14], 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.

4.3.1.3 P.Sigy_QSCD

TOE as Secure Signature Creation Device

The TOE meets the requirements for a QSCD laid down in [R14], Annex III. This implies that the SCD is used for digital signature creation and data decipherment under sole control of the Signatory and the SCD can practically occur only once.

4.3.1.4 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.

4.3.2 OSPs added to those defined in the PPs

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

4.3.2.1 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 key for the authentication of the Initialization Agent (cf. section 2.3.2).

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

The Pre-personalization Agent writes pre-personalization data in step 6, pre-personalization, of TOE life cycle (cf. section 2.3.2), including the key for the authentication of the Personalization Agent.

The Initialization Agent and the Pre-personalization Agent act on behalf of the QSCD provisioning service.

4.3.2.2 P.Personalization

Personalization of the e-Document

The Personalization Agent writes personalization data in step 7, 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.

4.3.2.3 P.Dec_Integrity

Integrity of decrypted data

The DDO shall be managed in a way to preserve their integrity.

4.4 Assumptions

4.4.1 Assumptions defined in the PPs

The PPs [R8] [R9] [R10] share the same assumptions, reported here below.

4.4.1.1 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.

4.4.1.2 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.

4.4.2 OSPs added to those defined in the PPs

Here below there is one more Assumption, added in this security target to those defined in the PPs.

4.4.2.1 A.DDA

Trustworthy Data Decipherment Application

The Signatory uses only a trustworthy DDA. The DDA generates and sends the DTBD of the data that the Signatory wishes to decipher in a form appropriate for decipherment by the TOE.

5. Security objectives

5.1 Security objectives for the TOE

Here below are the security objectives for the TOE defined in PP Part 2 [R8]. The Security Objectives OT.SCD_Unique, OT.SCD_SVD_Corresp and OT.SCD_Secrecy have been extended.

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 2 *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 and/or data decipherment. The SCD used for signature creation and/or data decipherment 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, in creating an electronic signature and in data decipherment with the SCD.

5.1.5 OT.SCD_Secrecy

Secrecy of Signature Creation Data

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

Application Note 3 *The TOE shall keep the confidentiality of the SCD at all times, in particular during SCD/SVD generation, signature creation operation, data decipherment 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 reconstructable 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 [R9].

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 [R10].

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 4 *This security objective for the TOE is partly covering OE.HID_VAD from PP Part 2 [R8]. While OE.HID_VAD in PP Part 2 requires only the operational environment to protect VAD, PP Part 5 [R10] 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 5 *This security objective for the TOE is partly covering OE.DTBS_Protect from PP Part 2 [R8]. While OE.DTBS_Protect in PP Part 2 requires only the operational environment to protect DTBS, PP Part 5 [R10] 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 TOE initialization data, including the pre-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_Pre-pers

Access control for the pre-personalization of the e-Document

The TOE must ensure that pre-personalization data, including the personalization key, can be written in step 6, pre-personalization, of TOE life cycle (cf. section 2.3.2) by the authorized Pre-personalization Agent only.

5.1.18 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 7, personalization, of TOE life cycle (cf. section 2.3.3) by the authorized Personalization Agent only.

5.1.19 OT.Abuse-Func

Protection against abuse of functionality

The TOE must prevent that functions of the TOE, which may not be used after TOE delivery, can be abused in order (i) to manipulate or disclose the user data stored in the TOE, (ii) to manipulate or disclose the TSF data stored in the TOE, or (iii) to manipulate (bypass, deactivate, or modify) the TSF.

5.1.20 OT.Sigy_DecF

Data decipherment function for the legitimate Signatory only

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

5.1.21 OT.DTBD_Integrity_TOE

DTBD integrity inside the TOE

The TOE must not alter the DTBD.

5.1.22 OT.TOE_TC_DTBD_Imp

TOE trusted channel for DTBD import

The TOE shall provide a trusted channel to the DDA to detect alteration of the DTBD received from the DDA. The TOE must not decipher encrypted data with the DDA for altered DTBD.

5.2 Security objectives for the operational environment

Here below are the security objectives for the operational environment defined in PP Part 2 [R8].

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 [R9].

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 6 *This objective replaces OE.QSCD_Prov_Service from PP Part 2 [R8], 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 7 *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 [R9] 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 [R8].*

Here below are the security objectives for the operational environment defined in PP Part 5 [R10].

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 8 *This security objective for the TOE is partly covering OE.HID_VAD from PP Part 2 [R8]. 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 [R10] 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 9 *This security objective for the TOE is partly covering OE.DTBS_Protect from PP Part 2 [R8]. 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 [R10] 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 operational environment, added in this security target to those defined in the PPs.

5.2.10 OE.DTBD_Intend

DDA sends data intended to be deciphered

The Signatory shall use a trustworthy DDA that:

- sends the DTBD to the TOE and enables verification of the integrity of the DTBD by the TOE;
- provides the deciphered data produced by the TOE.

5.2.11 OE.DDA_TC_DTBD_Exp

DDA trusted channel for DTBD export

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

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 [R8], PP Part 4 [R9], PP Part 5 [R10], 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_Pre-pers	OT.AC_Pers	OT.Abuse-Func	OT.Sigy_DecF	OT.DTBD_Integrity_TOE	OT.TOE_TC_DTBD_Imp
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																
T.Abuse-Func																			X			
T.DecF_Misuse	X																			X	X	X
T.DTBD_Forgery																					X	X
T.Dec_Forgery			X																			
P.CSP_QCert	X			X								X										
P.QSign						X	X															
P.Sigy_QSCD	X	X	X		X	X	X	X	X		X	X	X								X	X
P.Sig_Non-Repud	X		X	X	X	X	X	X	X	X	X	X	X	X	X							
P.Manufact																X	X					
P.Personalization		X																X				
P.Dec_Integrity	X		X	X	X				X	X	X			X							X	X	X
A.CGA																						

	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_Pre-pers	OT.AC_Pers	OT.Abuse-Func	OT.Sigy_DecF	OT.DTBD_Integrity_TOE	OT.TOE_TC_DTBD_Imp
A.SCA																						
A.DDA																						

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.DTBD_Intend	OE.DDA_TC_DTBD_Exp
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									
T.Abuse-Func											
T.DecF_Misuse				X				X		X	X
T.DTBD_Forgery										X	X
T.Dec_Forgery											
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						
P.Dec_Integrity				X				X		X	X

	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.DTBD_Intend	OE.DDA_TC_DTBD_Exp
A.CGA	X	X									
A.SCA			X								
A.DDA										X	

6.2 Sufficiency of security objectives

In PP Part 4 [R9], 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 [R8], section 7.3.2. The rationale how security objectives address threats T.SCD_Divulg, T.SVD_Forgery and policies P.CSP_QCert, P.Sigy_QSCD, and P.Sig_Non-Repud changes as reported below.
In PP Part 5 [R10], the rationale for T.Hack_Phys, T.SCD_Divulg, T.SCD_Derive, T.Sig_Forgery, T.SVD_Forgery, P.CSP_QCert, P.QSign, A.CGA, and A.SCA remains unchanged as given in PP Part 2 [R8], 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 [R8].

T.SCD_Divulg *(Storage, copy, and release of Signature Creation Data)* addresses the threat against the legal validity of electronic signature, as expressed in recital (18) of [R14], and confidentiality of encrypted data due to storage and copying of SCD outside the TOE. This threat is countered by ***OT.SCD_Secrecy***, which assures the secrecy of the SCD used for signature creation and encrypted data decipherment.

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 [R9].

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, signature creation and encrypted data decipherment 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 [R14], article 5, paragraph 1. [R14], 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 Secure Signature Creation Device)* requires the TOE to meet [R14], 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.Sigy_DecF*** meets the requirement to ensure that the TOE provides the data decipherment 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. ***OT.DTBD_Integrity_TOE*** meets the requirement that TOE must not alter the DTBD. 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 [R10].

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 [R14], 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 [R9] and PP Part 5 [R10].

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.

T.Abuse-Func *(Abuse of functionality)* addresses attacks abusing pre-delivery functionality of the TOE to manipulate or disclose the stored user or TSF data, as well as to disable or

bypass the TSF. ***OT.Abuse-Func*** ensures that the usage of functions having not to be used after the delivery of the TOE is effectively prevented.

T.DecF_Misuse *(Misuse of the decipherment function of the TOE)* addresses the threat of misuse of the TOE decipherment function to create a DDO by others than the Signatory, or to create a DDO for data for which the Signatory has not expressed the intent to decipher. ***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_DecF*** ensures that the TOE provides the data decipherment function for the legitimate Signatory only. ***OE.DTBD_Intend*** ensures that the DDA sends the DTBD only for data that the Signatory intends to decipher. The combination of ***OT.TOE_TC_DTBD_Imp*** and ***OE.DDA_TC_DTBD_Exp*** counters the undetected manipulation of the DTBD during the transmission from the DDA to the TOE. ***OT.DTBD_Integrity_TOE*** prevents the DTBD from alteration inside the TOE. If the DDA 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***. ***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.DTBD_Forgery *(Forgery of the DTBD)* addresses the threat arising from modifications of the DTBD sent to the TOE for decipherment. The threat is addressed by security objectives ***OT.TOE_TC_DTBD_Imp*** and ***OE.DDA_TC_DTBD_Exp***, which ensure that the DTBD is sent through a trusted channel and cannot be altered undetected in transit between the DDA and the TOE. The TOE counters internally this threat by means of ***OT.DTBD_Integrity_TOE***, ensuring the integrity of the DTBD inside the TOE. The TOE IT environment also addresses the threat by means of ***OE.DTBD_Intend***, which ensures that the trustworthy DDA sends the DTBD to the TOE and which the Signatory intends to decipher.

T.Dec_Forgery *(Forgery of the deciphered data)* deals with non-detectable forgery of the data decipherment. ***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.

P.Manufact *(Manufacturing of the e-Document)* requires the storage of TOE initialization data and pre-personalization data to be restricted to the Initialization Agent and to the Pre-personalization Agent, respectively, which is ensured by ***OT.AC_Init*** and ***OT.AC_Pre-pers***. Furthermore, since access control requires user authentication, the secure storage of the initialization key, the pre-personalization key and the personalization key prescribed by the

policy is implied by ***OT.AC_Init***, and ***OT.AC_Pre-pers***, respectively. Finally, the fact that the Initialization Agent and the Pre-personalization 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***. 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.

P.Dec_Integrity *(Integrity of decrypted data)* deals with the preservation of the integrity of DDO. This policy is implemented by the combination of the security objectives for the TOE and its operational environment. ***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.
OT.Sigy_DecF ensures that only the Signatory may use the TOE for data decipherment. 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.DTBD_Intend***, ***OT.DTBD_Integrity_TOE***, ***OE.DDA_TC_DTBD_Exp***, and ***OT.TOE_TC_DTBD_Imp*** ensure that the TOE generates deciphered data only for a DTBD that the Signatory has decided to decipher. 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.

A.DDA *(Trustworthy Data Decipherment Application)* establishes the trustworthiness of the DDA with respect to generation of DTBD. This is addressed by ***OE.DTBD_Intend***, which ensures that the DDA generates the DTBD and which the Signatory intends to decipher.

7. Extended components definition

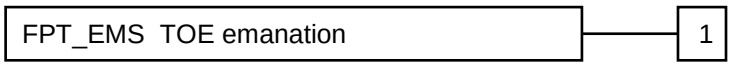
7.1 Definition of family FPT_EMS

The additional family FPT_EMS (TOE emanation) of class FPT (Protection of the TSF) is defined in PP Part 2 [R8] 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*].

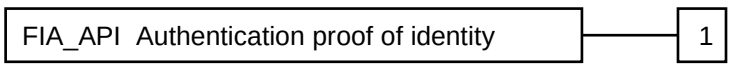
7.2 Definition of family FIA_API

The additional family FIA_API (Authentication proof of identity) of class FIA (Identification and authentication) is defined in PP Part 4 [R9] 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*].

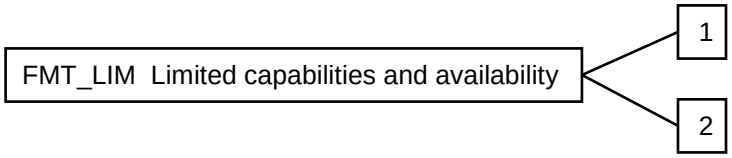
7.3 Definition of family FMT_LIM

The additional family FMT_LIM (Limited capabilities and availability) of class FMT (Security management) is defined in the PACE PP [R4] to describe the functional requirements for the test features of the TOE.
The new functional requirements are defined in the class FMT because this class addresses the management of the functions of the TSF. No other class is appropriate to address the specific issues of preventing abuse of functions by limiting the capabilities of the functions and by limiting their availability.

FMT_LIM Limited capabilities and availability

Family behaviour: This family defines requirements that limit the capabilities and availability of functions in a combined manner. Note that FDP_ACF [R6] restricts access to functions, whereas the component FMT_LIM.1 (Limited capabilities) of this family requires the functions themselves to be designed in a specific manner.

Component levelling:



- FMT_LIM.1 (Limited capabilities) requires that the TSF is built to provide only the capabilities (perform action, gather information) necessary for its genuine purpose.
- FMT_LIM.2 (Limited availability) requires that the TSF restricts the use of functions; refer to FMT_LIM.1 (Limited capabilities). This can be achieved, for instance, by removing or by disabling functions in a specific phase of the TOE life cycle.

Management: FMT_LIM.1, FMT_LIM.2

There are no management activities foreseen.

Audit: FMT_LIM.1, FMT_LIM.2

There are no actions defined to be auditable.

FMT_LIM.1 Limited capabilities

Hierarchical to: No other components.

Dependencies: FMT_LIM.2 Limited availability

FMT_LIM.1.1:

The TSF shall be designed in a manner that limits its capabilities so that in conjunction with “Limited availability (FMT_LIM.2)”, the following policy is enforced: [assignment: *limited capability and availability policy*].

FMT_LIM.2 Limited availability

Hierarchical to: No other components.

Dependencies: FMT_LIM.1 Limited capabilities

FMT_LIM.2.1:

The TSF shall be designed in a manner that limits its availability so that in conjunction with “Limited capabilities (FMT_LIM.1)”,

the following policy is enforced: [assignment: *limited capability and availability policy*].

Application Note 10 *The functional requirements FMT_LIM.1 and FMT_LIM.2 assume the existence of two types of mechanisms (limited capabilities and limited availability) which together shall provide protection in order to enforce the related policy. This also allows that (i) the TSF is provided without restrictions in the product in its user environment, but its capabilities are so limited that the policy is enforced, or conversely (ii) the TSF is designed with high functionality, but it is removed or disabled in the product in its user environment. The combination of both requirements shall enforce the related policy.*

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 [R9] and PP Part 5 [R10] since both PPs extend the formulation of the SFR given in PP Part 2 [R8]. 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/Signature_Creation	X		
FCS_COP.1/Data_Decipherment			
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_ACC.1/Data_Decipherment			
FDP_ACF.1/Signature_Creation	X		
FDP_ACF.1/Data_Decipherment			
FDP_RIP.1	X		
FDP_SDI.2/Persistent	X		
FDP_SDI.2/DTBS	X		
FDP_SDI.2/DTBD			
FDP_DAU.2/SVD		X	
FDP_UIT.1/DTBS			X
FDP_UIT.1/DTBD			
FIA_UID.1	X		
FIA_UAU.1		X	X
FIA_AFL.1/Signatory	X		
FIA_AFL.1/Admin			
FIA_AFL.1/Init			
FIA_AFL.1/Pre-pers			
FIA_AFL.1/Pers			
FIA_API.1		X	
FMT_SMR.1/QSCD	X		
FMT_SMR.1/Init			
FMT_SMR.1/Pre-pers			
FMT_SMR.1/Pers			
FMT_SMF.1	X		
FMT_MOF.1	X		
FMT_MSA.1/Admin	X		

Security functional requirement	PP Part 2	PP Part 4	PP Part 5
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/Pre-pers			
FMT_MTD.1/Pers			
FMT_LIM.1			
FMT_LIM.2			
FPT_EMS.1	X		
FPT_FLS.1	X		
FPT_PHP.1	X		
FPT_PHP.3	X		
FPT_TST.1	X		
FTP_ITC.1/SVD		X	
FTP_ITC.1/VAD			X
FTP_ITC.1/DTBS			X
FTP_ITC.1/DTBD			
FTP_ITC.1/Init			
FTP_ITC.1/Pre-pers			
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:

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 **2048, 3072, 4096 bits**[3] that meet the following: **PKCS #1 [R39]**[4].

Application Note 11 *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.*

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].

Application Note 12 *The TOE shall set as unavailable the Initialization Key at the end of initialization step, setting its counter to zero and locking the initialization mechanism. The TOE shall overwrite the pre-personalization key with the personalization key at the end of pre-personalization step. The TOE shall set as unavailable the personalization key at the end of personalization step, setting its state as destroyed.*

Application Note 13 *The TOE overwrites with zeros the SCD key in the volatile memory after its generation and usage.*

---

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

8.1.3 FCS_COP.1/Signature_Creation

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/Signature_Creation:

The TSF shall perform digital signature creation[7] in accordance with a specified cryptographic algorithm **RSASSA-PKCS1-v1_5 with SHA-256 or RSASSA-PSS with SHA-256**[8] and cryptographic key sizes **2048, 3072, 4096 bits**[9] that meet the following: **PKCS #1 [R39], FIPS PUB 180-4 [R33]**[10].

8.1.4 FCS_COP.1/Data_Decipherment

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/Data_Decipherment:

The TSF shall **perform encrypted data decipherment**[11] in accordance with a specified cryptographic algorithm **RSADP**[12]

[7] [assignment: *list of cryptographic operations*]
[8] [assignment: *cryptographic algorithm*]
[9] [assignment: *cryptographic key sizes*]
[10] [assignment: *list of standards*]
[11] [assignment: *list of cryptographic operations*]
[12] [assignment: *cryptographic algorithm*]

and cryptographic key sizes **2048, 3072, 4096 bits**[13] that meet the following: **PKCS #1 [R39]**[14].

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	-	-
DTBD	-	-

Application Note 14 *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.*

Application Note 15 *DTBD has been added to the list of subjects and objects provided in Table 8-2 because it is mentioned in SFRs* FDP_ACC.1/Data_Decipherment *and* FDP_ACF.1/Data_Decipherment*.*

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

- SCD;
- SVD.

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

- DTBS/R;

[13] [assignment: *cryptographic key sizes*]
[14] [assignment: *list of standards*]

- DTBD.

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[15] on

- subjects: S.User;
- objects: SCD, SVD;
- operations: generation of SCD/SVD pairs[16].

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[17] to objects based on the following: the user S.User is associated with the security attribute "SCD/SVD management"[18].

FDP_ACF.1.2/SCD/SVD_Generation:

[15] [assignment: *access control SFP*]
[16] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]
[17] [assignment: *access control SFP*]
[18] [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*]

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

S.User with the security attribute “SCD/SVD management” set to “authorized” is allowed to generate SCD/SVD pairs[19].

FDP_ACF.1.3/SCD/SVD_Generation:

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none[20].

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[21].

Application Note 16 *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[22] on

- subjects: S.User;
- objects: SVD;
- operations: export[23].

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

8.2.4 FDP_ACF.1/SVD_Transfer

Security attribute based access control – SVD transfer

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[24] to objects based on the following:

- the S.User is associated with the security attribute "Role";
- the SVD[25].

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**[26] are allowed to export SVD[27].

FDP_ACF.1.3/SVD_Transfer:

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none[28].

FDP_ACF.1.4/SVD_Transfer:

The TSF shall explicitly deny access of subjects to objects based on the following additional rules: none[29].

Application Note 17 *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).*

[24] [assignment: *access control SFP*]
[25] [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*]
[26] [selection: *R.Admin, R.Sigy*]
[27] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[28] [assignment: *rules, based on security attributes, that explicitly authorize access of subjects to objects*]
[29] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]

8.2.5 FDP_ACC.1/Signature_Creation

Subset access control – Signature creation

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[30] on

- subjects: S.User;
- objects: DTBS/R, SCD;
- operations: signature creation[31].

8.2.6 FDP_ACC.1/Data_Decipherment

Subset access control – Data decipherment

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/Data_decipherment:

The TSF shall enforce the **Data Decipherment SFP**[32] on

- subjects: S.User;
- objects: **DTBD**, SCD;
- operations: **data decipherment**[33].

8.2.7 FDP_ACF.1/Signature_Creation

Security attribute based access control – Signature creation

Hierarchical to: No other components.

[30] [assignment: *access control SFP*]
[31] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]
[32] [assignment: *access control SFP*]
[33] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

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[34] 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"[35].

FDP_ACF.1.2/Signature_Creation:

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

R.Sigy is allowed to create electronic signatures for DTBS/R with SCD whose security attribute "SCD operational" is set to "yes"[36].

FDP_ACF.1.3/Signature_Creation:

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none[37].

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"[38].

[34] [assignment: *access control SFP*]
[35] [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*]
[36] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[37] [assignment: *rules, based on security attributes, that explicitly authorize access of subjects to objects*]
[38] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]

8.2.8 FDP_ACF.1/Data_Decipherment

Security attribute based access control – Data Decipherment

Hierarchical to: No other components.

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

FDP_ACF.1.1/Data_Decipherment:

The TSF shall enforce the **Data Decipherment SFP**[39] 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"[40].

FDP_ACF.1.2/Data_Decipherment:

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

R.Sigy is allowed to decipher encrypted data for DTBD with SCD whose security attribute "SCD operational" is set to "yes"[41].

FDP_ACF.1.3/Data_Decipherment:

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none[42].

FDP_ACF.1.4/Data_Decipherment:

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

[39] [assignment: *access control SFP*]
[40] [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*]
[41] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[42] [assignment: *rules, based on security attributes, that explicitly authorize access of subjects to objects*]

S.User is not allowed to decipher encrypted data for DTBD with SCD whose security attribute "SCD operational" is set to "no"[43].

8.2.9 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[44] the following objects: SCD[45].

Application Note 18 *The identification of integrity checked data that appears just before the statement of SFR FDP_SDI.2/Persistent within PP Part 2 [R8] has been moved to the beginning of section 8.2 in this security target.*

8.2.10 FDP_SDI.2/Persistent

Stored data integrity monitoring and action – Persistent data

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring

Dependencies: No dependencies.

FDP_SDI.2.1/Persistent:

The TSF shall monitor user data stored in containers controlled by the TSF for integrity errors[46] on all objects, based on the following attributes: integrity checked stored data[47].

[43] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]
[44] [selection: *allocation of the resource to, deallocation of the resource from*]
[45] [assignment: *list of objects*]
[46] [assignment: *integrity errors*]
[47] [assignment: *user data attributes*]

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[48].

8.2.11 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[49] on all objects, based on the following attributes: integrity checked stored DTBS[50].

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[51].

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

8.2.12 FDP_SDI.2/DTBD

Stored data integrity monitoring and action – DTBD

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring

Dependencies: No dependencies.

[48] [assignment: *action to be taken*]
[49] [assignment: *integrity errors*]
[50] [assignment: *user data attributes*]
[51] [assignment: *action to be taken*]

FDP_SDI.2.1/DTBD:

The TSF shall monitor user data stored in containers controlled by the TSF for integrity errors[52] on all objects, based on the following attributes: **integrity checked stored DTBD**[53].

FDP_SDI.2.2/DTBD:

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[54].

Application Note 20 *The DTBD is a cryptogram resulting from the encryption of a raw data encapsulated as specified in section 6.3.1.1 of [R15]. The TOE shall check the integrity of the DTBD verifying the presence of this encapsulation after the decryption. This check is successful just in case the DTBD has not been altered, because even the change of a single bit of the DTBD is enough to have an inconsistent plaintext. If this check fails, the TOE returns an error status word, so as to inform the user of the integrity error.*

8.2.13 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[55].

FDP_DAU.2.2/SVD:

The TSF shall provide the CGA[56] with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

[52] [assignment: *integrity errors*]
[53] [assignment: *user data attributes*]
[54] [assignment: *action to be taken*]
[55] [assignment: *list of objects or information types*]
[56] [assignment: *list of subjects*]

Application Note 21 *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 [R15]. For more details, cf. section 2.2.2.*

8.2.14 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:

The TSF shall enforce the Signature Creation SFP[57] to receive[58] user data in a manner protected from modification and insertion[59] errors.

FDP_UIT.1.2/DTBS:

The TSF shall be able to determine on receipt of user data, whether modification or insertion[60] has occurred.

8.2.15 FDP_UIT.1/DTBD

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]

[57] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]
[58] [selection: *transmit, receive*]
[59] [selection: *modification, deletion, insertion, replay*]
[60] [selection: *modification, deletion, insertion, replay*]

FDP_UIT.1.1/DTBD:

The TSF shall enforce the **Data Decipherment SFP**[61] to receive[62] user data in a manner protected from modification and insertion[63] errors.

FDP_UIT.1.2/DTBD:

The TSF shall be able to determine on receipt of user data, whether modification or insertion[64] 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 Initialization Agent's terminal and the TOE by means of TSF required by FTP_ITC.1/Init;**

[61] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]
[62] [selection: *transmit, receive*]
[63] [selection: *modification, deletion, insertion, replay*]
[64] [selection: *modification, deletion, insertion, replay*]

- **establishing a trusted channel between the Pre-personalization Agent's terminal and the TOE by means of TSF required by FTP_ITC.1/Pre-pers;**
- **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 information to the Initialization Agent,**
- **returning product information to the Pre-personalization Agent**[65] [66]

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 [R8] 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;

[65] [assignment: *list of additional TSF-mediated actions*]
[66] [assignment: *list of TSF-mediated actions*]

- ~~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 Initialization Agent's terminal and the TOE by means of TSF required by FTP_ITC.1/Init;**
- **establishing a trusted channel between the Pre-personalization Agent's terminal and the TOE by means of TSF required by FTP_ITC.1/Pre-pers;**
- **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 information to the Initialization Agent,**
- **returning product information to the Pre-personalization Agent**[67] [68].

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 [R9] performs the assignment of the bullet (3) in the element FIA_UAU.1.1 of PP Part 2 [R8] by adding the establishment of a trusted channel to the CGA.*

[67] [assignment: *list of additional TSF-mediated actions*]
[68] [assignment: *list of TSF-mediated actions*]

Application Note 25 *PP Part 5 [R10] performs the assignment of the bullet (3) in the element FIA_UAU.1.1 of PP Part 2 [R8] by adding the establishment of a trusted channel to the HID.*

Application Note 26
During TOE initialization (cf. section 2.3.2), the Initialization Agent can retrieve product information before authentication and then establish a trusted channel with the TOE by means of a GIM authentication, which consists of sending updated product information and/or configuration data (optionally) and the pre-personalization key, encrypted with the initialization key, to the TOE. For further information, cf. the initialization guidance.

Application Note 27 *During TOE pre-personalization (cf. section 2.3.2), the Pre-personalization Agent can retrieve product information before authentication and then establish a trusted channel with the TOE through a CPS authentication following EMV CPS specification [R11]. For further information, cf. the pre-personalization guidance [R20].*

Application Note 28 *During TOE personalization (cf. section 2.3.3), the Personalization Agent can establish a trusted channel with the TOE through a CPS authentication following EMV CPS specification [R11]. Any other operation, including product information retrieval, requires CPS authentication as a precondition. For more information, cf. the personalization guidance [R21].*

Application Note 29 *During TOE operational use (cf. section 2.3.4), users can establish a trusted channel with the TOE by means of a PACE authentication compliant with ICAO Doc 9303 [R25]. Any other operation, including product information retrieval, requires PACE authentication as a precondition. For more information, cf. section 2.2 and the operational user guidance [R22].*

8.3.3 FIA_AFL.1/Signatory

Authentication failure handling

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1/Signatory:

The TSF shall detect when **an administrator configurable positive integer within the range between 1 and 255**[69]

[69] [selection: *[assignment: positive integer number]*, *an administrator configurable positive integer within [assignment: range of acceptable values]*]

unsuccessful authentication attempts occur related to consecutive failed authentication attempts[70] **with respect to RAD**.

FIA_AFL.1.2/Signatory:

When the defined number of unsuccessful authentication attempts has been met[71], the TSF shall block RAD[72].

Application Note 30 *This security target refines the element FIA_AFL.1.1/Signatory to specify that it refers to consecutive failed authentication attempts with respect to the RAD.*

Application Note 31 *Distinct thresholds within the specified range 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 in a range from 2 to 8 seconds, depending on the clock frequency, 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.*
The threshold for PACE authentication is set by the subject that creates Signatory's PACE key, namely the Pre-personalization Agent (cf. section 2.3.2), on behalf of the QSCD provisioning service.
The threshold for password verification is set by the subject that creates Signatory's passwords, namely the Personalization Agent (cf. section 2.3.3), on behalf of the QSCD provisioning service.

Application Note 32 *The system clock frequency is dependent on the power received from the terminal reader. The value of the system clock frequency is dynamically adjusted according to the preconfigured power settings of the OS.*

8.3.4 FIA_AFL.1/Admin

Authentication failure handling

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1/Admin:

[70] [assignment: *list of authentication events*]
[71] [selection: *met, surpassed*]
[72] [assignment: *list of actions*]

The TSF shall detect when **an administrator configurable positive integer within the range between 1 and 255**[73] unsuccessful authentication attempts occur related to **consecutive failed authentication attempts with respect to the Administrator's credentials**[74].

FIA_AFL.1.2/Admin:

When the defined number of unsuccessful authentication attempts has been **met**[75], the TSF shall **block the Administrator's credentials**[76].

Application Note 33 *Distinct thresholds within the specified range apply to both steps of Administrator's authentication, 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 in a range from 2 to 8 seconds, depending on the clock frequency, 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 Administrator's credentials as a whole.*
The threshold for PACE authentication is set by the subject that creates Administrator's PACE key, namely the Pre-personalization Agent (cf. section 2.3.2), on behalf of the QSCD provisioning service.
The threshold for password verification is set by the subject that creates Administrator's password, namely the Personalization Agent (cf. section 2.3.3), on behalf of the QSCD provisioning service.

Application Note 34 *The system clock frequency is dependent on the power received from the terminal reader. The value of the system clock frequency is dynamically adjusted according to the preconfigured power settings of the OS.*

8.3.5 FIA_AFL.1/Init

Authentication failure handling

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

[73] [selection: *[assignment: positive integer number]*, *an administrator configurable positive integer within [assignment: range of acceptable values]*]
[74] [assignment: *list of authentication events*]
[75] [selection: *met, surpassed*]
[76] [assignment: *list of actions*]

FIA_AFL.1.1/Init:

The TSF shall detect when **31**[77] unsuccessful authentication attempts occur related to **consecutive failed authentication attempts with respect to the initialization key**[78].

FIA_AFL.1.2/Init:

When the defined number of unsuccessful authentication attempts has been **met**[79], the TSF shall **block the initialization key**[80].

8.3.6 FIA_AFL.1/Pre-pers

Authentication failure handling

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1/Pre-pers:

The TSF shall detect when **3**[81] unsuccessful authentication attempts occur related to **consecutive failed authentication attempts with respect to the pre-personalization key**[82].

FIA_AFL.1.2/Pre-pers:

When the defined number of unsuccessful authentication attempts has been **met**[83], the TSF shall **block the pre-personalization key**[84].

---

[77] [selection: *[assignment: positive integer number]*, *an administrator configurable positive integer within [assignment: range of acceptable values]*]
[78] [assignment: *list of authentication events*]
[79] [selection: *met, surpassed*]
[80] [assignment: *list of actions*]
[81] [selection: *[assignment: positive integer number]*, *an administrator configurable positive integer within [assignment: range of acceptable values]*]
[82] [assignment: *list of authentication events*]
[83] [selection: *met, surpassed*]
[84] [assignment: *list of actions*]

8.3.7 FIA_AFL.1/Pers

Authentication failure handling

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1/Pers:

The TSF shall detect when **an administrator configurable positive integer within the range between 1 and 15**[85] unsuccessful authentication attempts occur related to **consecutive failed authentication attempts with respect to the personalization key**[86].

FIA_AFL.1.2/Pers:

When the defined number of unsuccessful authentication attempts has been **met**[87], the TSF shall **block the personalization key**[88].

Application Note 35 *The threshold for authentication with respect to the personalization key is set by the subject writing the key, namely the Pre-personalization Agent (cf. section 2.3.2), on behalf of the QSCD provisioning service.*

8.3.8 FIA_API.1

Authentication proof of identity

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_API.1.1:

[85] [selection: *[assignment: positive integer number]*, *an administrator configurable positive integer within [assignment: range of acceptable values]*]
[86] [assignment: *list of authentication events*]
[87] [selection: *met, surpassed*]
[88] [assignment: *list of actions*]

The TSF shall provide **Client/Server Authentication compliant with IAS ECC specification [R15]**[89] to prove the identity of the QSCD[90].

Application Note 36 *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).*

8.4 Class FMT: Security management

8.4.1 FMT_SMR.1/QSCD

Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1/QSCD:

The TSF shall maintain the roles R.Admin and R.Sigy[91].

FMT_SMR.1.2/QSCD:

The TSF shall be able to associate users with roles.

8.4.2 FMT_SMR.1/Init

Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1/Init:

The TSF shall maintain the roles **R.Init**[92].

[89] [assignment: *authentication mechanism*]
[90] [assignment: *authorized user or role*]
[91] [assignment: *the authorized identified roles*]
[92] [assignment: *the authorized identified roles*]

FMT_SMR.1.2/Init:

The TSF shall be able to associate users with roles.

8.4.3 FMT_SMR.1/Pre-pers

Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1/Pre-pers:

The TSF shall maintain the roles **R.Pre-pers**[93].

FMT_SMR.1.2/Pre-pers:

The TSF shall be able to associate users with roles.

8.4.4 FMT_SMR.1/Pers

Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1/Pers:

The TSF shall maintain the roles **R.Pers**[94].

FMT_SMR.1.2/Pers:

The TSF shall be able to associate users with roles.

8.4.5 FMT_SMF.1

Specification of management functions

Hierarchical to: No other components.

[93] [assignment: *the authorized identified roles*]
[94] [assignment: *the authorized identified roles*]

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”;
- **enabling the data decipherment function;**
- **unblock of RAD;**
- **writing TOE initialization data,**
- **writing pre-personalization data;**
- **writing personalization data**[95] [96].

8.4.6 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[97] the functions signature creation function, **data decipherment function**[98] to R.Sigy[99].

[95] [assignment: *list of other security management functions to be provided by the TSF*]
[96] [assignment: *list of security management functions to be provided by the TSF*]
[97] [selection: *determine the behaviour of, disable, enable, modify the behaviour of*]
[98] [assignment: *list of functions*]
[99] [assignment: *the authorized identified roles*]

8.4.7 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[100] to restrict the ability to modify[101] [102] the security attributes "SCD/SVD management"[103] to R.Admin[104].

8.4.8 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, **Data Decipherment SFP**[105] to restrict the ability to modify[106] the security attributes "SCD operational"[107] to R.Sigy[108].

[100] [assignment: *access control SFP(s), information flow control SFP(s)*]
[101] [assignment: *other operations*]
[102] [selection: *change_default, query, modify, delete, [assignment: other operations]*]
[103] [assignment: *list of security attributes*]
[104] [assignment: *the authorized identified roles*]
[105] [assignment: *access control SFP(s), information flow control SFP(s)*]
[106] [selection: *change_default, query, modify, delete, [assignment: other operations]*]
[107] [assignment: *list of security attributes*]
[108] [assignment: *the authorized identified roles*]

8.4.9 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"[109].

Application Note 37 *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).*

8.4.10 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, Signature Creation SFP, and **Data Decipherment SFP**[110] to provide restrictive[111] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2:

[109] [assignment: *list of security attributes*]
[110] [assignment: *access control SFP, information flow control SFP*]
[111] [selection, choose one of: *restrictive, permissive, [assignment: other property]*]

The TSF shall allow the R.Admin[112] to specify alternative initial values to override the default values when an object or information is created.

8.4.11 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[113].

8.4.12 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:

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

The TSF shall restrict the ability to create[114] the RAD[115] to R.Admin[116].

8.4.13 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**[117] [118] the RAD[119] to R.Sigy[120].

8.4.14 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**[121] the **TOE initialization data**[122] to **R.Init**[123].

[114] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]
[115] [assignment: *list of TSF data*]
[116] [assignment: *the authorized identified roles*]
[117] [assignment: *other operations*]
[118] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]
[119] [assignment: *list of TSF data*]
[120] [assignment: *the authorized identified roles*]
[121] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]
[122] [assignment: *list of TSF data*]
[123] [assignment: *the authorized identified roles*]

8.4.15 FMT_MTD.1/Pre-pers

Management of TSF data – Pre-personalization Agent

Hierarchical to: No other components.

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

FMT_MTD.1.1/Pre-pers:

The TSF shall restrict the ability to **write**[124] the **pre-personalization data**[125] to **R.Pre-pers**[126].

8.4.16 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**[127] the **personalization data**[128] to **R.Pers**[129].

8.4.17 FMT_LIM.1

Limited capabilities

Hierarchical to: No other components.

Dependencies: FMT_LIM.2 Limited availability

FMT_LIM.1.1:

[124] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]
[125] [assignment: *list of TSF data*]
[126] [assignment: *the authorized identified roles*]
[127] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]
[128] [assignment: *list of TSF data*]
[129] [assignment: *the authorized identified roles*]

The TSF shall be designed in a manner that limits its capabilities so that in conjunction with “Limited availability (FMT_LIM.2)”, the following policy is enforced: **deploying test features after TOE delivery does not allow user data of the TOE to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed, or any substantial information about the construction of the TSF to be gathered which may enable other attacks**[130].

8.4.18 FMT_LIM.2

Limited availability

Hierarchical to: No other components.

Dependencies: FMT_LIM.1 Limited capabilities

FMT_LIM.2.1:

The TSF shall be designed in a manner that limits its availability so that in conjunction with “Limited capabilities (FMT_LIM.1)”, the following policy is enforced: **deploying test features after TOE delivery does not allow user data of the TOE to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed, or any substantial information about the construction of the TSF to be gathered which may enable other attacks**[131].

8.5 Class FPT: Protection of the TSF

8.5.1 FPT_EMS.1

TOE emanation

Hierarchical to: No other components.

Dependencies: No dependencies.

[130] [assignment: *limited capability and availability policy*]
[131] [assignment: *limited capability and availability policy*]

FPT_EMS.1.1:

The TOE shall not emit **any measurable emissions**[132] in excess of **intelligible thresholds**[133] enabling access to RAD[134] and SCD[135].

FPT_EMS.1.2:

The TSF shall ensure **any users**[136] are unable to use the following interface **contact-based/contactless interface and circuit contacts**[137] to gain access to RAD[138] and SCD[139].

Application Note 38 *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:

[132] [assignment: *types of emissions*]
[133] [assignment: *specified limits*]
[134] [assignment: *list of types of TSF data*]
[135] [assignment: *list of types of user data*]
[136] [assignment: *type of users*]
[137] [assignment: *type of connection*]
[138] [assignment: *list of types of TSF data*]
[139] [assignment: *list of types of user data*]

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**[140] [141].

Application Note 39 *The assignments address failures detected by a failed self-test or revealing the occurrence of a physical attack, and requiring appropriate action to prevent 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:

[140] [assignment: *list of other types of failures in the TSF*]
[141] [assignment: *list of types of failures in the TSF*]

The TSF shall resist **physical manipulation and physical probing**[142] to the **TSF**[143] by responding automatically such that the SFRs are always enforced.

Application Note 40 *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 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, signature creation or data decipherment, 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 [R20] [R21] [R22].*

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, and at the conditions: before any use of TSF data**[144] to demonstrate the correct operation of the TSF[145].

FPT_TST.1.2:

The TSF shall provide authorized users with the capability to verify the integrity of TSF data[146].

[142] [assignment: *physical tampering scenarios*]
[143] [assignment: *list of TSF devices/elements*]
[144] [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]*]
[145] [selection: *[assignment: parts of TSF], the TSF*]
[146] [selection: *[assignment: parts of TSF data], TSF data*]

FPT_TST.1.3:

The TSF shall provide authorized users with the capability to verify the integrity of TSF[147].

8.6 Class FTP: Trusted path/channels

8.6.1 FTP_ITC.1/SVD

Inter-TSF trusted channel – SVD

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[148] 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**[149] [150].

[147] [selection: *[assignment: parts of TSF], TSF*]
[148] [selection: *the TSF, another trusted IT product*]
[149] [assignment: *list of other functions for which a trusted channel is required*]
[150] [assignment: *list of functions for which a trusted channel is required*]

Application Note 41 *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.

FTP_ITC.1.1/VAD:

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

FTP_ITC.1.2/VAD:

The TSF shall permit the remote trusted IT product[151] 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;
- **import of a new value of the RAD from the HID**[152] [153].

Application Note 42 *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. Moreover,*

[151] [selection: *the TSF, another trusted IT product*]
[152] [assignment: *list of other functions for which a trusted channel is required*]
[153] [assignment: *list of functions for which a trusted channel is required*]

the TSF requires the use of the same trusted channel for the import of a new value of the RAD from the HID (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 identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2/DTBS:

The TSF shall permit the remote trusted IT product[154] 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 digital signatures to the SCA**[155] [156].

Application Note 43 *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 digital signatures to the SCA (cf. section 2.2.3).*

8.6.4 FTP_ITC.1/DTBD

Inter-TSF trusted channel – DTBD

[154] [selection: *the TSF, another trusted IT product*]
[155] [assignment: *list of other functions for which a trusted channel is required*]
[156] [assignment: *list of functions for which a trusted channel is required*]

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1/DTBD:

The TSF shall provide a communication channel between itself and another trusted IT product **DDA** 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/DTBD:

The TSF shall permit another trusted IT product[157] to initiate communication via the trusted channel.

FTP_ITC.1.3/DTBD:

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

- **data decipherment;**
- **export of deciphered data to the DDA**[158] [159].

Application Note 44 *The component FTP_ITC.1/DTBD requires the TSF to enforce a trusted channel established by the DDA to import the DTBD from the DDA. Moreover, the TSF requires the use of the same trusted channel for the export of deciphered data to the DDA (cf. section 2.2.4).*

8.6.5 FTP_ITC.1/Init

Inter-TSF trusted channel – TOE initialization data

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1/Init:

[157] [selection: *the TSF, another trusted IT product*]
[158] [assignment: *list of other functions for which a trusted channel is required*]
[159] [assignment: *list of functions for which a trusted channel is required*]

The TSF shall provide a communication channel between itself and another trusted IT product**, the Initialization 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/Init:

The TSF shall permit **another trusted IT product**[160] to initiate communication via the trusted channel.

FTP_ITC.1.3/Init:

The TSF **or the Initialization Agent's terminal** shall initiate communication via the trusted channel for **import of TOE initialization data from the terminal**[161].

Application Note 45 *The component* FTP_ITC.1/Init *requires the TSF to enforce a trusted channel established by the Initialization Agent's terminal to import TOE initialization data from the terminal. This trusted channel is established through a GIM authentication and uses cryptographic algorithms AES [R34], SHA-256 [R33]. For further information, cf. the initialization guidance.*

8.6.6 FTP_ITC.1/Pre-pers

Inter-TSF trusted channel – Pre-personalization data

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1/Pre-pers:

The TSF shall provide a communication channel between itself and another trusted IT product**, the Pre-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/Pre-pers:

[160] [selection: *the TSF, another trusted IT product*]

[161] [assignment: *list of functions for which a trusted channel is required*]

The TSF shall permit **another trusted IT product**[162] to initiate communication via the trusted channel.

FTP_ITC.1.3/Pre-pers:

The TSF **or the Pre-personalization Agent's terminal** shall initiate communication via the trusted channel for **import of pre-personalization data from the terminal**[163].

Application Note 46 *The component FTP_ITC.1/Pre-pers requires the TSF to enforce a trusted channel established by the Pre-personalization Agent's terminal to import pre-personalization data from the terminal. This trusted channel is established through a through a CPS authentication and uses cryptographic algorithm TDES [R31] [R32]. For further information, cf. EMV CPS specification [R11] and the pre-personalization guidance [R20].*

Application Note 47 *FIPS 46-3 was withdrawn in 2005. The Triple Data Encryption Algorithm with 112 bit keys is still an NIST approved cryptographic algorithm as defined in NIST SP 800-67 [R31]. NIST SP 800-38A [R32] provides recommendation for block cipher modes.*

8.6.7 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:

The TSF shall permit **another trusted IT product**[164] to initiate communication via the trusted channel.

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

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**[165].

Application Note 48 *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 through a CPS authentication and uses cryptographic algorithm TDES [R31] [R32]. For further information, cf. EMV CPS specification [R11] and the personalization guidance [R21].*

Application Note 49 *FIPS 46-3 was withdrawn in 2005. The Triple Data Encryption Algorithm with 112 bit keys is still an NIST approved cryptographic algorithm as defined in NIST SP 800-67 [R31]. NIST SP 800-38A [R32] provides recommendation for block cipher modes.*

[165] [assignment: *list of functions for which a trusted channel is required*]

9. Security assurance requirements

The Evaluation Assurance Level claimed by this security target is EAL5 augmented with ALC_DVS.2 and AVA_VAN.5 [R7] (cf. section 3.2). Moreover, the refinements to security assurance requirements for composite product evaluations are also applied [R30].
Table 9-1 summarizes the security assurance requirements enforced by this security target.

Table 9-1 Security assurance requirements: EAL5 augmented with ALC_DVS.2 and AVA_VAN.5

Assurance class	Assurance components
ADV *Development*	ADV_ARC.1 *Security architecture description*
	ADV_FSP.5 *Complete semiformal functional specification with additional error information*
	ADV_IMP.1 *Implementation representation of the TSF*
	ADV_INT.2 *Well-structured internals*
	ADV_TDS.4 *Semiformal modular design*
AGD *Guidance documents*	AGD_OPE.1 *Operational user guidance*
	AGD_PRE.1 *Preparative procedures*
ALC *Life cycle support*	ALC_CMC.4 *Production support, acceptance procedures and automation*
	ALC_CMS.5 *Development tools CM coverage*
	ALC_DEL.1 *Delivery procedures*
	ALC_DVS.2 *Sufficiency of security measures*
	ALC_LCD.1 *Developer defined life-cycle model*
	ALC_TAT.2 *Compliance with implementation standards*
ASE *Security target evaluation*	ASE_CCL.1 *Conformance claims*
	ASE_ECD.1 *Extended components definition*
	ASE_INT.1 *ST introduction*

Assurance class	Assurance components
	ASE_OBJ.2 *Security objectives*
	ASE_REQ.2 *Derived security requirements*
	ASE_SPD.1 *Security problem definition*
	ASE_TSS.1 *TOE summary specification*
ATE *Tests*	ATE_COV.2 *Analysis of coverage*
	ATE_DPT.3 *Testing: modular design*
	ATE_FUN.1 *Functional testing*
	ATE_IND.2 *Independent testing - sample*
AVA *Vulnerability assessment*	AVA_VAN.5 *Advanced methodical vulnerability analysis*

10. Security requirements rationale

10.1 Coverage of security functional requirements

Table 10-1 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 [R8], PP Part 4 [R9], PP Part 5 [R10], or this security target).

Table 10-1 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_Init	OT.AC_Pre-pers	OT.AC_Pers	OT.Abuse-Func	OT.Sigy_DecF	OT.DTBD_Integrity_TOE	OT.TOE_TC_DTBD_Imp
FCS_CKM.1	X		X	X	X																	
FCS_CKM.4	X				X																	
FCS_COP.1/Signature_Creation	X					X																
FCS_COP.1/Data_Decipherment	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_ACC.1/Data_Decipherment	X																			X		
FDP_ACF.1/Signature_Creation	X						X															
FDP_ACF.1/Data_Decipherment	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_Init	OT.AC_Pre-pers	OT.AC_Pers	OT.Abuse-Func	OT.Sigy_DecF	OT.DTBD_Integrity_TOE	OT.TOE_TC_DTBD_Imp
FDP_RIP.1					X		X													X		
FDP_SDI.2/Persistent				X	X	X																
FDP_SDI.2/DTBS							X	X														
FDP_SDI.2/DTBD																				X	X	
FDP_DAU.2/SVD													X									
FDP_UIT.1/DTBS															X							
FDP_UIT.1/DTBD																						X
FIA_UID.1		X					X									X	X	X		X		
FIA_UAU.1		X					X					X				X	X	X		X		
FIA_AFL.1/Signatory		X					X													X		
FIA_AFL.1/Admin		X																				
FIA_AFL.1/Init																X						
FIA_AFL.1/Pre-pers																	X					
FIA_AFL.1/Pers																		X				
FIA_API.1												X										
FMT_SMR.1/QSCD	X						X													X		
FMT_SMR.1/Init	X															X						
FMT_SMR.1/Pre-pers	X																X					
FMT_SMR.1/Pers	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_Init	OT.AC_Pre-pers	OT.AC_Pers	OT.Abuse-Func	OT.Sigy_DecF	OT.DTBD_Integrity_TOE	OT.TOE_TC_DTBD_Imp
FMT_SMF.1	X			X			X									X	X	X		X		
FMT_MOF.1	X						X													X		
FMT_MSA.1/Admin	X	X																				
FMT_MSA.1/Signatory	X						X													X		
FMT_MSA.2	X	X					X													X		
FMT_MSA.3	X	X		X			X													X		
FMT_MSA.4	X	X					X													X		
FMT_MTD.1/Admin	X						X													X		
FMT_MTD.1/Signatory	X						X													X		
FMT_MTD.1/Init																X						
FMT_MTD.1/Pre-pers																	X					
FMT_MTD.1/Pers																		X				
FMT_LIM.1																			X			
FMT_LIM.2																			X			
FPT_EMS.1					X				X													
FPT_FLS.1					X																	
FPT_PHP.1										X												
FPT_PHP.3					X						X											
FPT_TST.1	X				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_Init	OT.AC_Pre-pers	OT.AC_Pers	OT.Abuse-Func	OT.Sigy_DecF	OT.DTBD_Integrity_TOE	OT.TOE_TC_DTBD_Imp
FTP_ITC.1/SVD													X									
FTP_ITC.1/VAD														X								
FTP_ITC.1/DTBS															X							
FTP_ITC.1/DTBD																						X
FTP_ITC.1/Init																X						
FTP_ITC.1/Pre-pers																	X					
FTP_ITC.1/Pers																		X				

10.2 Sufficiency of security functional requirements

Here below is the rationale for the security objectives borrowed from PP Part 2 [R8].

OT.Lifecycle_Security *(Life cycle security)* is provided by the SFRs for SCD/SVD generation ***FCS_CKM.1***, SCD usage ***FCS_COP.1/Signature_Creation*** *and* ***FCS_COP.1/Data_Decipherment***, 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_ACC.1/Data_Decipherment,*** ***FDP_ACF.1/Signature_Creation*** and ***FDP_ACF.1/Data_Decipherment,*** 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***, ***FMT_SMR.1/QSCD***, ***FMT_SMR.1/Init***, ***FMT_SMR.1/Pre-pers***, and ***FMT_SMR.1/Pers***. 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.

SFRs ***FIA_AFL.1/Signatory*** and ***FIA_AFL.1/Admin*** provide protection against trial-and-error attacks (particularly, brute force attacks) with respect to the authentication credentials of either of the two roles authorized to generate SCD/SVD pairs, i.e. the Signatory and the Administrator.

OT.SCD_Unique *(Uniqueness of Signature Creation Data)* implements the requirement of practically unique SCD as laid down in [R14], 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/Signature_Creation***, 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/Signatory*** 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/QSCD*** 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 for the security objectives borrowed from PP Part 4 [R9].

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 for the security objectives borrowed from PP Part 5 [R10].

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/Init***, which specifies how unsuccessful authentication attempts are managed for the authentication as Initialization Agent;
- ***FMT_MTD.1/Init*** (based on ***FMT_SMR.1/Init*** and ***FMT_SMF.1***), which restricts the capability to write TOE initialization data to the Initialization Agent;

- ***FTP_ITC.1/Init***, which requires the TSF to enforce a trusted channel for the import of TOE initialization data, so as to ensure that the data actually written match those sent by the Initialization Agent.

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

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

OT.Abuse-Func *(Protection against abuse of functionality)* is aimed at preventing TOE functions not intended to be used after TOE delivery from manipulating or disclosing user data, TSF data, or the TSF itself. This objective is covered by ***FMT_LIM.1*** and ***FMT_LIM.2***, which prevent abuse of test features of the TOE having not to be used after TOE delivery.

OT.Sigy_DecF *(Data decipherment 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 data decipherment 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/Signatory*** 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/DTBD*** ensures the integrity of stored DTBD, and ***FDP_RIP.1*** prevents misuse of any resources containing the SCD after de-allocation (e.g. after the data decipherment process).

The security functions specified by ***FDP_ACC.1/Data_Decipherment*** and ***FDP_ACF.1/Data_Decipherment*** 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/QSCD*** 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.DTBD_Integrity_TOE *(DTBD integrity inside the TOE)* ensures that the DTBD is not altered by the TOE. The integrity functions specified by ***FDP_SDI.2/DTBD*** require that the DTBD has not been altered by the TOE.

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

10.3 Satisfaction of dependencies of security requirements

Table 10-2 Satisfaction of dependencies of security functional requirements

Requirement	Dependencies	Satisfied by
FCS_CKM.1	FCS_CKM.2 or FCS_COP.1	FCS_COP.1/Signature_Creation, FCS_COP.1/Data_Decipherment
	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/Signature_Creation	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	FCS_CKM.1
	FCS_CKM.4	FCS_CKM.4
FCS_COP.1/Data_Decipherment	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

Requirement	Dependencies	Satisfied by
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_ACC.1/ Data_Decipherment	FDP_ACF.1	FDP_ACF.1/ Data_Decipherment
FDP_ACF.1/ Signature_Creation	FDP_ACC.1	FDP_ACC.1/ Signature_Creation
	FMT_MSA.3	FMT_MSA.3
FDP_ACF.1/ Data_Decipherment	FDP_ACC.1	FDP_ACC.1/ Data_Decipherment
	FMT_MSA.3	FMT_MSA.3
FDR_RIP.1	No dependencies	-
FDP_SDI.2/Persistent	No dependencies	-
FDP_SDI.2/DTBS	No dependencies	-
FDP_SDI.2/DTBD	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
FDP_UIT.1/DTBD	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/ Data_Decipherment
	FTP_ITC.1 or FTP_TRP.1	FTP_ITC.1/DTBD
FIA_UID.1	No dependencies	-
FIA_UAU.1	FIA_UID.1	FIA_UID.1
FIA_AFL.1/Signatory	FIA_UAU.1	FIA_UAU.1

Requirement	Dependencies	Satisfied by
FIA_AFL.1/Admin	FIA_UAU.1	FIA_UAU.1
FIA_AFL.1/Init	FIA_UAU.1	FIA_UAU.1
FIA_AFL.1/Pre-pers	FIA_UAU.1	FIA_UAU.1
FIA_AFL.1/Pers	FIA_UAU.1	FIA_UAU.1
FIA_API.1	No dependencies	-
FMT_SMR.1/QSCD	FIA_UID.1	FIA_UID.1
FMT_SMR.1/Init	FIA_UID.1	FIA_UID.1
FMT_SMR.1/Pre-pers	FIA_UID.1	FIA_UID.1
FMT_SMR.1/Pers	FIA_UID.1	FIA_UID.1
FMT_SMF.1	No dependencies	-
FMT_MOF.1	FMT_SMR.1	FMT_SMR.1/QSCD
	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/QSCD
	FMT_SMF.1	FMT_SMF.1
FMT_MSA.1/Signatory	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/ Signature_Creation FDP_ACC.1/ Data_Decipherment
	FMT_SMR.1	FMT_SMR.1/QSCD
	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 FDP_ACC.1/ Data_Decipherment
	FMT_MSA.1	FMT_MSA.1/Admin, FMT_MSA.1/Signatory

Requirement	Dependencies	Satisfied by
	FMT_SMR.1	FMT_SMR.1/QSCD
FMT_MSA.3	FMT_MSA.1	FMT_MSA.1/Admin, FMT_MSA.1/Signatory
	FMT_SMR.1	FMT_SMR.1/QSCD
FMT_MSA.4	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/ SCD/SVD_Generation, FDP_ACC.1/ Signature_Creation FDP_ACC.1/ Data_Decipherment
FMT_MTD.1/Admin	FMT_SMR.1	FMT_SMR.1/QSCD
	FMT_SMF.1	FMT_SMF.1
FMT_MTD.1/Signatory	FMT_SMR.1	FMT_SMR.1/QSCD
	FMT_SMF.1	FMT_SMF.1
FMT_MTD.1/Init	FMT_SMR.1	FMT_SMR.1/Init
	FMT_SMF.1	FMT_SMF.1
FMT_MTD.1/Pre-pers	FMT_SMR.1	FMT_SMR.1/Pre-pers
	FMT_SMF.1	FMT_SMF.1
FMT_MTD.1/Pers	FMT_SMR.1	FMT_SMR.1/Pers
	FMT_SMF.1	FMT_SMF.1
FMT_LIM.1	FMT_LIM.2	FMT_LIM.2
FMT_LIM.2	FMT_LIM.1	FMT_LIM.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	-

Requirement	Dependencies	Satisfied by
FTP_ITC.1/SVD	No dependencies	-
FTP_ITC.1/VAD	No dependencies	-
FTP_ITC.1/DTBS	No dependencies	-
FTP_ITC.1/DTBD	No dependencies	-
FTP_ITC.1/Init	No dependencies	-
FTP_ITC.1/Pre-pers	No dependencies	-
FTP_ITC.1/Pers	No dependencies	-

Table 10-3 Satisfaction of dependencies of security assurance requirements

Requirement	Dependencies	Satisfied by
EAL5 package	Dependencies of the EAL5 package are not reproduced here (cf. [R7])	By construction, all dependencies are satisfied in a CC EAL package
ALC_DVS.2	No dependencies	-
AVA_VAN.5	ADV_ARC.1	ADV_ARC.1[166]
	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

10.4 Rationale for security assurance requirements

The assurance level for this security target is EAL5 augmented. EAL5 permits a developer to gain maximum assurance from security engineering based upon rigorous commercial development practises, supported by moderate application of specialist security engineering

[166] This assurance component and the subsequent ones are all included in the EAL5 package.

techniques. Such a TOE will be designed and developed with the intent of achieving EAL5 assurance. EAL5 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, OT.Sig_Secure and OT.Sigy_DecF (cf. section 5.1).

11. TOE summary specification

Table 11-1 describes how each security functional requirement claimed in this security target is satisfied by the TOE.

Table 11-1 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/Signature_Creation	As specified for SFR FCS_CKM.1.
FCS_COP.1/Data_Decipherment	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_ACC.1/Data_Decipherment	As specified for SFR FDP_ACF.1/Data_Decipherment
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 [R29]. 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_ACF.1/Data_Decipherment	The private key objects storing the private keys meant for data decipherment (cf. section 2.3) contain an access condition for data decipherment, as well as a life cycle state compliant with ISO/IEC 7816-9 [R29]. The access condition refers to Signatory's credentials (cf. section 2.2.1); moreover, data decipherment is forbidden unless the life cycle state matches "operational activated".
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_SDI.2/DTBD	Cf. Application Note 20.
FDP_DAU.2/SVD	Cf. Application Note 21.
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).
FDP_UIT.1/DTBD	DTBD import must be executed over the trusted channel opened by means of Signatory's PACE authentication step (cf. sections 2.2.1, 2.2.4).
FIA_UID.1	Cf. Application Note 22.
FIA_UAU.1	Cf. Application Note 27, Application Note 28, Application Note 29.
FIA_AFL.1/Signatory	The thresholds for authentication failures with respect to the RAD are set by the actors that write the related persistent objects (cf. Application Note 31). The behaviour occurring if the thresholds are reached is as specified in the statement of the SFR (cf. Application Note 31).

Security functional requirement	Implementation
FIA_AFL.1/Admin	The thresholds for authentication failures with respect to the Administrator's credentials are set by the actors that write the related persistent objects (cf. Application Note 33). The behaviour occurring if the thresholds are reached is as specified in the statement of the SFR (cf. Application Note 33).
FIA_AFL.1/Init	The Initialization Agent has only a limited number of authentication attempts, whether successful or unsuccessful, after which the Initialization key is blocked. The maximum number of authentications is set to 31.
FIA_AFL.1/Pre-pers	In case of unsuccessful authentication, the Pre-personalization Agent has only a limited number of authentication attempts after which the Pre-personalization keys are blocked. The maximum number of consecutive failures is set to 3.
FIA_AFL.1/Pers	The threshold for authentication failures with respect to the personalization key is set by the actor that writes the related persistent object (cf. Application Note 35). The behaviour occurring if the thresholds are reached is as specified in the statement of the SFR.
FIA_API.1	Cf. section 2.2.2.
FMT_SMR.1/QSCD	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_SMR.1/Init	The Initialization Agent role is implicitly identified via the corresponding authentication key.
FMT_SMR.1/Pre-pers	The Pre-personalization Agent role is implicitly identified via the corresponding authentication key.
FMT_SMR.1/Pers	The Personalization Agent role is implicitly identified via the corresponding authentication key.
FMT_SMF.1	Cf. section 2.3.
FMT_MOF.1	The Signatory alone can activate the signature creation and/or data decipherment 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).

Security functional requirement	Implementation
FMT_MSA.1/Signatory	The private key objects storing the private keys meant for signature creation and/or data decipherment (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.
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), data decipherment (as specified for SFRs FDP_ACF.1/SCD/SVD_Generation, FDP_ACF.1/Data_Decipherment), 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 filled by the Personalization Agent on behalf of the Administrator (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).

Security functional requirement	Implementation
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 31). The password object storing Signatory's password #2 (cf. section 2.3) contains access conditions for password modification and unblock, which refer to Signatory's credentials (cf. section 2.2.1).
FMT_MTD.1/Init	The command APDUs available for the writing of TOE initialization data (cf. section 2.3.2) are protected by an implicit access condition, which requires user authentication with respect to the initialization key.
FMT_MTD.1/Pre-pers	The command APDUs available for the writing of pre-personalization data (cf. section 2.3.2) are protected by an implicit access condition, which during pre-personalization requires user authentication with respect to the pre-personalization key.
FMT_MTD.1/Pers	The command APDUs available for the writing of personalization data (cf. section 2.3.3) are protected by an implicit access condition, which during personalization requires user authentication with respect to the personalization key.
FMT_LIM.1	The test features of the OS, as well as the authentication mechanism granting access to them, are permanently disabled in the evaluated configuration of the OS. As regards the test features of the IC, information on their limitation is provided in the TOE summary specification of the public security target of the supported IC for platform SFRs FMT_LIM.1, FMT_LIM.2 [R35].
FMT_LIM.2	As specified for SFR FMT_LIM.1.
FPT_EMS.1	Leakage of confidential data through side channels is prevented by the security features of both the IC and the OS, in accordance with the security recommendations contained in the IC guidance documentation [R36] [R37][R38].
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 [R36] [R37] [R38].

Security functional requirement	Implementation
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. [R36] [R37]), 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 41.
FTP_ITC.1/VAD	Cf. Application Note 42.
FTP_ITC.1/DTBS	Cf. Application Note 43.
FTP_ITC.1/DTBD	Cf. Application Note 44.
FTP_ITC.1/Init	Cf. Application Note 45.
FTP_ITC.1/Pre-pers	Cf. Application Note 46.
FTP_ITC.1/Pers	Cf. Application Note 48.

12. References

12.1 Acronyms

AA	Active Authentication
AES	Advanced Encryption Standard
APDU	Application Protocol Data Unit
ASCII	American Standard Code for Information Interchange
BAC	Basic Access Control
CC	Common Criteria
CGA	Certificate Generation Application
CPS	Card Personalization Specification
CRC	Cyclic Redundancy Check
CSP	Certification Service Provider
DDA	Data Decipherment Application
DDO	Deciphered Data Object
DES	Data Encryption Standard
DF	Dedicated/Directory File
DFA	Differential Power Analysis
DTBD	Data To Be Deciphered
DTBS	Data To Be Signed
DTBS/R	Data To Be Signed Representation
EAC	Extended Access Control
EAL	Evaluation Assurance Level
ECC	European Citizen Card
EF	Elementary File
FID	File Identifier
HID	Human Interface Device
IAS	Identification Authentication Signature
IC	Integrated Circuit
ICAO	International Civil Aviation Organization
IT	Information Technology
LDS	Logical Data Structure

MAC	Message Authentication Code
MF	Master File
MRTD	Machine Readable Travel Document
OS	Operating System
OSP	Organizational Security Policy
PACE	Password Authenticated Connection Establishment
PP	Protection Profile
PUC	Personal Unblocking Code
QSCD	Qualified Signature Creation Device
RAD	Reference Authentication Data
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
SSCD	Secure 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

12.2 Technical references

[R1] **BSI:** *Certification Report BSI-DSZ-CC-1136-V3-2022 for NXP Smart Card Controller N7121 with IC Dedicated Software and Crypto Library (R1/R2/R3/R4) from NXP Semiconductors Germany GmbH, 7 September 2022*

[R2] **BSI:** *Technical Guideline TR-03110-1, Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token, Part 1: eMRTDs with BAC/PACEv2 and EACv1, version 2.20, February 2015*

[R3] **BSI:** *Technical Guideline TR-03110-3, Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token, Part 3: Common Specifications, version 2.21, December 2016*

[R4] **BSI:** *Common Criteria Protection Profile, Machine Readable Travel Document using Standard Inspection Procedure with PACE (PACE PP), version 1.01, July 2014, ref. BSI-CC-PP-0068-V2-2011-MA-01*

[R5] **CCMB:** *Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, version 3.1, revision 5, April 2017, ref. CCMB-2017-04-001*

[R6] **CCMB:** *Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components, version 3.1, revision 5, April 2017, ref. CCMB-2017-04-002*

[R7] **CCMB:** *Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components, version 3.1, revision 5, April 2017, ref. CCMB-2017-04-003*

[R8] **CEN:** *Protection profiles for secure signature creation device, Part 2: Device with Key Generation, version 2.0.1, EN 419211-2:2013 (certificate BSI-CC-PP-0059-2009-MA-02)*

[R9] **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, EN 419211-4:2013 (certificate BSI-CC-PP-0071-2012-MA-01)*

[R10] **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, EN 419211-5:2013 (certificate BSI-CC-PP-0072-2012-MA-01)*

[R11] **EMV:** *Card Personalization Specification, version 1.0, June 2003*

[R12] **European Parliament:** *Regulation (EU) No 910/2014 of the European Parliament and of the Council, 23 July 2014*

[R13] **European Parliament:** *Commission Implementing Decision (EU) 2016/650, 25 April 2016*

[R14] **European Parliament:** *Directive 1999/93/EC on a Community framework for electronic signatures, December 1999*

[R15] **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*

[R16] **HID Global:** *Security Target for SOMA-c016 Machine Readable Electronic Document – ICAO Application – Basic Access Control, ref. TCAE180001*

[R17] **HID Global:** *Security Target for SOMA-c016 Machine Readable Electronic Document – ICAO Application – EAC-PACE-AA, ref. TCAE180002*

[R18] **HID Global:** *Security Target for SOMA-c016 Machine Readable Electronic Document – eIDAS QSCD Application, ref. TCAE180021*

[R19] **HID Global:** *Initialization Guidance for SOMA-c016 Machine Readable Electronic Document, ref. TCAE180013*

[R20] **HID Global:** *Pre-personalization Guidance for SOMA-c016 Machine Readable Electronic Document – eIDAS QSCD Application, ref. TCAE180015*

[R21] **HID Global:** *Personalization Guidance for SOMA-c016 Machine Readable Electronic Document – eIDAS QSCD Application, ref. TCAE180017*

[R22] **HID Global:** *Operational User Guidance for SOMA-c016 Machine Readable Electronic Document – eIDAS QSCD Application, ref. TCAE180019*

[R23] **HID Global:** *Secure Delivery Procedure for SOMA-c016 Machine Readable Electronic Document, ref. TCAE190001*

[R24] **ICAO:** *Doc 9303, Machine Readable Travel Documents, Part 10: Logical Data Structure (LDS) for Storage of Biometrics and Other Data in the Contactless Integrated Circuit (IC),* Eighth Edition, 2021

[R25] **ICAO:** *Doc 9303, Machine Readable Travel Documents, Part 11: Security Mechanisms for MRTDs,* Eighth Edition, 2021

[R26] **ICAO:** *Doc 9303, Machine Readable Travel Documents, Part 12: Public Key Infrastructure for MRTDs,* Eighth Edition, 2021

[R27] **IETF Network Working Group:** *Request for Comments 2119, Key words for use in RFCs to Indicate Requirement Levels, March 1997*

[R28] **ISO/IEC:** *International Standard 7816-4:2020. Identification cards - Integrated circuit(s) cards with contacts – Part 4: Interindustry commands for interchange*

[R29] **ISO/IEC:** *International Standard 7816-9:2017, Identification cards – Integrated circuit cards – Part 9: Commands for card management*

[R30] **JIWG:** *Joint Interpretation Library, Composite product evaluation for Smart Cards and similar devices, version 1.5.1, May 2018*

[R31] **NIST:** *Special Publication 800-67 Revision 2, Recommendation for the Triple Data Encryption Standard (TDEA) Block Cipher, November 2017*

[R32] **NIST**: *Special Publication 800-38A, Recommendation for Block Cipher Modes of Operation, December 2001*

[R33] **NIST:** *FIPS PUB 180-4, Federal Information Processing Standards Publication, Secure Hash Standard (SHS), March 2012*

[R34] **NIST:** *FIPS PUB 197, Federal Information Processing Standards Publication, Advanced Encryption Standard (AES), November 2001*

[R35] **NXP:** *NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library (R1/R2/R3/R4), Security Target Lite Rev. 2.6 – 13 June 2022*

[R36] **NXP:** *SmartMX3 family P71D321 Overview, pinning and electrical characteristics, Rev. 3.5 - 5 July 2021 ref. 416535*

[R37] **NXP:** *NXP Secure Smart Card Controller N7121 Information on Guidance and Operation Rev. 3.2 – 28 May 2019 ref. 431232*

[R38] **NXP:** *N7121 Crypto Library Information on Guidance and Operation, Product user manual, Rev. 3.4 - 4 May 2022 ref. 441534*

[R39] **RSA Laboratories:** *PKCS #1: RSA Cryptography Standard, version 2.2, October 2012*

[R40] **RSA Laboratories:** *PKCS #15: Cryptographic Token Information Syntax Standard, version 1.1, June 2000*

Appendix A Platform identification

The IC on which the TOE is based, constituting the platform for its composite evaluation (cf.[R30]), is the NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library (R1/R2/R3/R4).

The IC includes:

- N7121 IC Hardware Release B1
- IC Dedicated Test Software Release 9.2.3
- IC Dedicated Support Software Release 9.2.3 including:
  - Flashloader OS Release 1.2.5
  - Communication Library Release 6.0.0
  - CRC Library 1.1.8
  - Memory Library 1.2.3
  - Flash Loader Library 3.6.0
  - System Mode OS Release 13.2.3
  - Crypto Library Release 0.7.6.

The release packages used for the TOE SOMA-c016 is the R1/R2 and R3.

The IC has obtained a Common Criteria certification at Evaluation Assurance Level EAL6 augmented by ASE_TSS.2 and ALC_FLR.1:

- Certification ID: BSI-DSZ-CC-1136-V3-2022
- Security Target: [R35]
- Certification Report: [R1]