Signature Activation Module

Trust Remote InfoCert Signing Server – TRISS

Security Target

Version 2.2

InfoCert
TINEXTA GROUP

SUMMARY

1 DOCUMENT PRESENTATION 6
1.1 REVISION HISTORY 6
1.2 REFERENCES 7
1.3 TERMS AND DEFINITIONS 8
1.4 CONFIDENTIALITY 9
1.5 PURPOSE AND CONTENT OF THE DOCUMENT 9
2 SECURITY TARGET INTRODUCTION (ASE_INT) 11
2.1 SECURITY TARGET REFERENCE 11
2.2 TOE REFERENCE 11
2.3 TOE OVERVIEW 11
2.3.1 TOE IN ITS ENVIRONMENT 13
2.3.2 TOE TYPE 14
2.3.3 USAGE AND MAJOR SECURITY FEATURES 15
2.3.4 IDENTIFICATION OF THE CM REQUIRED BY THE TOE 16
2.3.5 IDENTIFICATION OF NON-TOE HARDWARE/SOFTWARE/FIRMWARE REQUIRED BY THE TOE 16
3 TOE DESCRIPTION (ASE_INT) 18
3.1 PHYSICAL SCOPE 18
3.1.1 LIFE-CYCLE 20
3.1.2 DELIVERY 20
3.2 LOGICAL SCOPE 21
3.2.1 SECURITY AUDIT 21
3.2.2 CRYPTOGRAPHIC SUPPORT 21
3.2.3 USER DATA PROTECTION 21
3.2.4 IDENTIFICATION AND AUTHENTICATION 21
3.2.5 SECURITY MANAGEMENT 22
3.2.6 PROTECTION OF THE TSF 22
3.2.7 TRUSTED PATH/CHANNELS 22
4 CONFORMANCE CLAIM (ASE_CCL) 23
4.1 CC CONFORMANCE CLAIM 23
4.2 PP CLAIM 23
4.3 PACKAGE CLAIM 23
5 SECURITY PROBLEM DEFINITION (ASE_SPD) 24

5.1 ASSETS 24
5.2 SUBJECTS 26
5.3 THREATS 27
5.3.1 ENROLMENT 27
5.3.2 SIGNER MANAGEMENT 28
5.3.3 USAGE 28
5.3.4 SYSTEM 30
5.4 RELATION BETWEEN THREATS AND ASSETS 31
5.5 ORGANIZATIONAL SECURITY POLICIES 33
5.6 ASSUMPTIONS 33
6 SECURITY OBJECTIVES (ASE_OBJ) 35
6.1 SECURITY OBJECTIVES FOR THE TOE 35
6.1.1 ENROLMENT 35
6.1.2 USER MANAGEMENT 35
6.1.3 USAGE 36
6.1.4 SYSTEM 37
6.2 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT 37
6.2.1 SECURITY PROBLEM DEFINITION AND SECURITY OBJECTIVES 39
6.3 RATIONALE FOR THE SECURITY OBJECTIVES 43
6.3.1 THREATS AND OBJECTIVES 43
6.3.2 ORGANIZATIONAL SECURITY POLICIES AND OBJECTIVES 49
6.3.3 ASSUMPTIONS AND OBJECTIVES 50
7 EXTENDED COMPONENT DEFINITION (ASE_ECD) 52
8 SECURITY REQUIREMENTS (ASE_REQ) 54
8.1 SECURITY FUNCTIONAL REQUIREMENTS (SFR) 58
8.1.1 SECURITY AUDIT (FAU) 58
8.1.2 CRYPTOGRAPHIC SUPPORT (FCS) 60
8.1.3 USER DATA PROTECTION (FDP) 64
8.1.4 IDENTIFICATION AND AUTHENTICATION (FIA) 78
8.1.5 SECURITY MANAGEMENT (FMT) 82
8.1.6 PROTECTION OF THE TSF (FPT) 86
8.1.7 TRUSTED PATHS/CHANNELS (FTP) 88
8.1.8 SFR DEPENDENCY ANALYSIS 91
8.2 SECURITY ASSURANCE REQUIREMENTS (SAR) 95
8.2.1 SECURITY ASSURANCE REQUIREMENTS RATIONALE 95

8.3 SECURITY REQUIREMENTS RATIONALE 97
8.3.1 SECURITY REQUIREMENTS COVERAGE 97
8.3.2 SECURITY REQUIREMENTS SUFFICIENCY 99
9 TOE SUMMARY SPECIFICATION (ASE_TSS) 102
9.1 TOE SECURITY FUNCTIONS SPECIFICATION 102
9.1.1 SECURITY AUDIT 102
9.1.2 CRYPTOGRAPHIC SUPPORT 103
9.1.3 USER DATA PROTECTION 104
9.1.4 IDENTIFICATION AND AUTHENTICATION 109
9.1.5 SECURITY MANAGEMENT 112
9.1.6 PROTECTION OF THE TSF (FPT) 116
9.1.7 TRUSTED PATH-CHANNELS 117
9.2 SFRS TO SECURITY FUNCTIONS COVERAGE 119

INDEX FIGURES

FIGURE 1 - TOE OVERVIEW 13
FIGURE 2 – PHYSICAL SCOPE 18

INDEX TABLES

TABLE 1 – TERMS AND DEFINITIONS 9
TABLE 2 - TOE REFERENCE 11
TABLE 3 – CM REFERENCE 16
TABLE 4 – TOE COMPONENTS 19
TABLE 5 – NON-TOE COMPONENTS 20
TABLE 6 - THREATS VS ASSETS 32
TABLE 7 - THREATS VS TOE SECURITY OBJECTIVES 39
TABLE 8 - TOE SECURITY OBJECTIVES AND ORGANIZATIONAL SECURITY POLICIES 41
TABLE 9 - THREATS VS SECURITY OBJECTIVES FOR THE ENVIRONMENT 42
TABLE 10 - ORGANIZATIONAL SECURITY POLICIES AND ASSUMPTIONS VS SECURITY OBJECTIVES FOR THE ENVIRONMENT 43
TABLE 11 - OPERATIONS SUPPORTED BY THE TOE 55
TABLE 12 - KEY GENERATION TABLE 60
TABLE 13 – CRYPTOGRAPHIC ALGORITHMS TABLE 62
TABLE 14 - ROLES VS OPERATIONS 85
TABLE 15 - DEPENDENCIES OF THE FUNCTIONAL REQUIREMENTS 91
TABLE 16 - SECURITY ASSURANCE REQUIREMENTS 95
TABLE 17 - SECURITY REQUIREMENTS COVERAGE 97

1 DOCUMENT PRESENTATION

1.1 Revision History

Version	Data	Author	Change/log
1.0	8th October 2020	InfoCert	First issue
1.1	2nd December 2020	InfoCert	Update after first review
1.2	14th December 2020	BRS	Second round of review
1.3	17th December 2020	InfoCert	Update after second review
1.4	1st September 2021	InfoCert	Update after design changes
1.5	11th November 2021	InfoCert	Fix after BRS review
1.6	24th January 2022	InfoCert	Update after EM1
1.7	10th February 2022	InfoCert	Update after BRS review
1.8	2nd March 2022	InfoCert	Update after EM2
2.0	15th March 2022	InfoCert	Final version
2.1	31st March 2022	InfoCert	Update TOE version in chapter 2
2.2	16th June 2022	InfoCert	Update the TOE definition to be composite

1.2 References

[1] Minimum technical specifications and procedures for assurance levels for electronic identification means [EU 2015/1502]

[2] Common Criteria for Information Technology Security Evaluation. Part 1-3, April 2017, Ver. 3.1 Revision 5 - [CC]

[3] Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC [eIDAS]

[4] Security Requirement for Trustworthy Systems Supporting Server Signing [EN 419 241-1]

[5] Cryptographic Module for Trust Services Server Signing [EN 419 221-5]

[6] Trustworthy Systems Supporting Server Signing Part 2: Protection Profile for QSCD for Server Signing – EN 419 241-2:2019

[7] NIST - Cryptographic Module Validation Program

https://csrc.nist.gov/Projects/Cryptographic-Module-Validation-Program/Certificate/2644

[8] https://csrc.nist.gov/Projects/Cryptographic-Algorithm-Validation-Program/Validation/Validation-List/DRBG (vedi #825)

[9] OCSI – Security Target NShield Thales

http://www.ocsi.isticom.it/documenti/certificazioni/thales/st_thales_nshield_v1.0_public.pd

[10] https://infocert.digital

[11] https://www.infocert.it

[12] ETSI, ETSI TS 119 312 Electronic Signatures and Infrastructures (ESI); Cryptographic Suites version v.1.2.2, 2018-09

[13] SOG-IS, SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms, version 1.1, 2018

[14] RFC 6749 The OAuth 2.0 Authorization Framework

[15] RFC 7519 JSON Web Token (JWT)

[16] RFC 3647 Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework

[17] COMMISSION IMPLEMENTING REGULATION (EU) 2015/1502 of 8 September 2015 on setting out minimum technical specifications and procedures for assurance levels for electronic identification means pursuant to Article 8(3) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market

[18] COMMISSION IMPLEMENTING DECISION (EU) 2015/1506 of 8 September 2015 laying down specifications relating to formats of advanced electronic signatures and advanced

seals to be recognised by public sector bodies pursuant to Articles 27(5) and 37(5) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market

[19] Cryptographic Module Common Criteria certificate publication, Maintenance Report "NSCIB-CC-0368256_1m1-MA-1.0.pdf" https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0368256_1m1-MA-1.0.pdf

[20] Cryptographic Module Common Criteria Security Target publication "nShield Solo XC HSM Security Target v1.1.1" https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0368256_1m1-ST.pdf

1.3 Terms and Definitions

Acronym	Description
CA	Certification Authority
CC	Common Criteria
CM	Cryptographic Module
CSR	Certificate Signing Request
DTBS	Data to be signed
EAL	Evaluation Assurance Level
IT	Information Technology
PP	Protection Profile
PU	Privileged User(s)
QSCD	Qualified Electronic Signature Creation Device or Qualified Electronic Seal Signature Creation Device
QTSP	Qualified Trust Service Provider
RA	Registration Authority
SAD	Signature Activation Data
SAM	Signature Activation Module also referred to as TOE
SAP	Signature Activation Protocol
SAR	Security Assurance Requirement
SCA	Signature Creation Application
SCD	Signature Creation Device
SF	Security Function
SFP	Security Function Policy
SFR	Security Functional Requirement

SIC	Signer Interaction Component
SPD	Security Problem Definition
SSA	Sign Server Application
ST	Security Target
SVD	Signature Verification Data
TOE	Target of Evaluation
TRISS	Trust Remote InfoCert Signing Server (TOE) also referred to as (SAM)
TSF	TOE Security Function
TSP	Trust Service Provider
TW4S	Trustworthy Systems Supporting Server Signing
OSP	Organizational Security Policies
OE	Operational Environment, intended as both technical and legal

TABLE 1 – TERMS AND DEFINITIONS

1.4 Confidentiality

The final release of the Security Target is a public document.

1.5 Purpose and content of the document

The Security Target contains the following sections.

Security Target introduction (ASE_INT) [Section 2]: this section gives an overview of the TOE based on TOE type. A unique identification for TOE and ST is also provided. This Section describes the TOE in terms of its boundaries, security features and non-TOE elements.

TOE description (ASE_INT) [Section 3]: is a description of the physical and logical scope of the TOE.

Conformance Claim (ASE_CCL) [Section 4]: states conformance to Common Criteria, Protection Profile and assurance package.

Security Problem Definition (ASE_SPD) [Section 5]: this section details assets and threats that are countered by the TOE and the environment. Here a presentation of the assumptions and the organizational policies that the TOE must fulfil is also included. This part of the ST defines the security problem to be addressed by the TOE and the operational environment of the TOE.

Security Objectives (ASE_OBJ) [Section 6]: this section details the security objectives of the TOE and of its operational environment.

Extended Component definition (ASE_ECD) (Section 7): This section presents the definition of the extended component FCS_RNG.1.

Security Requirements (ASE_REQ) [Section 8]: this section deals with security functional requirements (SFRs) for the TOE and presents details of the assurance requirements (SARs).

TOE Summary Specification (ASE_TSS) [Section 9]: this section describes the security functions of the TOE and how they satisfy the security requirements in Section 8.

2 SECURITY TARGET INTRODUCTION (ASE_INT)

2.1 Security Target reference

Title: *SECURITY TARGET FOR "TRUST REMOTE INFOCERT SIGNING SERVER (TRISS) V 1.0.2"*

*Document version **V. 2.2***

Date: 16.06.2022

2.2 TOE Reference

Product Name	TRISS Trust Remote InfoCert Signing Server
Product Version	1.0.2
TOE identification Data	1a7e91b4d70bbe968181cd7a8132cf9931743c1b0d4434883cad9bbe3628181a
Evaluation Criteria	Common Criteria version 3.1 revision 5
Protection Profile	EN 419 241-2:2019
Evaluation Assurance Level	EAL4 augmented by AVA_VAN.5
Client	InfoCert S.p.A.
Developer	InfoCert S.p.A.
Certification Body	NSCIB
Certification ID	0490158

TABLE 2 - TOE REFERENCE

2.3 TOE Overview

Trust Remote InfoCert Signing Server (TRISS) is an InfoCert proprietary Signature Activation Module (SAM), which is part of a Trustworthy System Supporting Server Signing (TW4S) as defined in [5]. TRISS makes use of a Cryptographic Module that has been certified against Protection Profile [5]. Indeed, TRISS is a software application running inside the CPU of the Cryptographic Module. Together, TRISS and the CM constitutes a composite product, in this document referred to as the "TOE".

A TW4S is a system that offers creation of remote qualified electronic signatures and seals as a service. The system:

- consists of a signer local environment and a remote environment. The signer in the local environment makes use of a device (e.g. laptop, tablet or smart phone) to provide Signature Activation Data (SAD) to the Server Signing Application (SSA) in the remote environment and request signing/sealing services.
- The Signature Activation Module (SAM) ensures that signing key(s) are only used under the sole control of the signer for their intended purpose. To this end, the signature operation needs to be authorised before being executed.

The TOE implements the server-side endpoint of the dedicated Signature Activation Protocol (SAP) for secure reception of SAD, verifies the SAD and activates the signing key within the Cryptographic Module that has been certified against Protection Profile [5]. The Cryptographic Module, which is part of the TOE, generates the signing key and creates the digital signature value.

The TOE, i.e. the software application running inside the Cryptographic Module together with the Cryptographic Module itself, constitute a QSigCD/QSealCD, depending on whether the signature is bound to a natural person (QSigCD) or to a legal person (QSealCD).

The TOE receives the SAD from the user's environment over a trusted channel along the whole hardware, network and software infrastructure. Once received, the TOE verifies the SAD by

1. checking the binding between the three SAD elements (signer authentication, signing key and representation (such as hash) of one or more documents to be signed) and by
2. checking that the Signer is authenticated according to one of the three methods described in [4] SCAL.2 for qualified signatures. In this ST, the TOE makes use of the indirect Signer authentication method, where an external authentication service as part of the TW4S or as a delegated party verifies the Signer's authentication factor(s) and issues an assertion that ensures that the Signer has been authenticated. Then the TOE verifies such assertion passed by SSA – Server Signing Application.

After verification of the SAD, the TOE uses the signing key in many ways:

a. Multi-session → the signing key can be used in several signing sessions with previous authentication and authorization. The Signer has the sole control of his/her signing keys.
b. Asynchronous → a Signer authentication and authorization can last for an established range of time. This means that the signature can be created within a certain period after successful authentication. The Signer has still the sole control of his/her signing keys.
c. One-time → the signing key is created, certified and used within a signing session limited in time. At the end of the session the signing key is destroyed. The sole control is valid only once and the Signer can sign more than one document in the unique activity session.
d. Seal → it is issued to a legal person and is suitable for automated processing. The signing key can be used in several signing sessions with previous authorization. As stated in eIDAS [3], electronic seals should serve as evidence that an electronic document was issued by a legal person, ensuring certainty of the document's origin and integrity.
e. Batch → it is possible to sign a batch of documents, without requiring the Signer to open each document before signing it. As the legal applicability of batch signing depends on the legal and application environment, the TW4S has configuration profiles to allow or disallow batch signing for digital signatures. This mode is applicable in some EU Member States and the Signer has only to apply sole controls to the signing process for a batch rather than each individual document [4].

2.3.1 TOE in its environment

The illustration below provides a general overview of the environment of the TOE (i.e. TRISS software application and CM in bold light-blue borders) and of the overall Server Signing System.

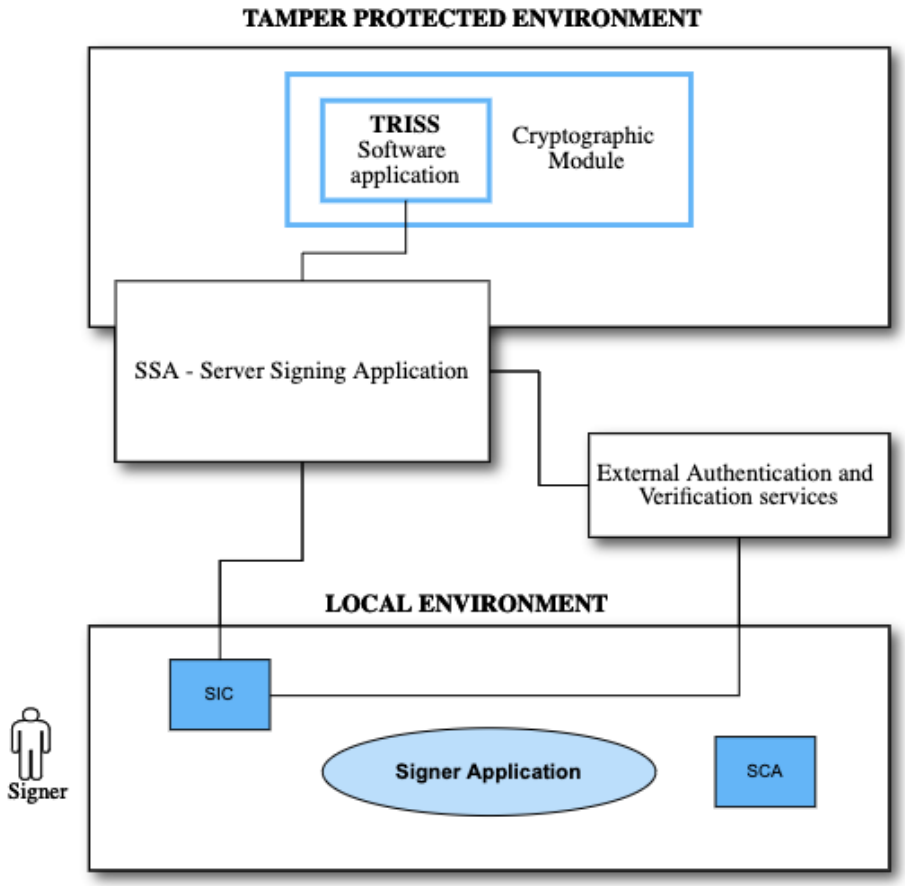


FIGURE 1 - TOE OVERVIEW

In the context of this evaluation, the TRISS software application is located within the physical boundary of the Cryptographic Module, thus leveraging on its FIPS 3 perimeter. Therefore, as specified in [5] and [6], a secure trusted channel for communication between the software application and the Cryptographic Module's services is not required as the physical protected environment and the local nature of the connection provide the integrity and confidentiality protection of the data, as well as mutual authentication of the two IT entities. The Cryptographic Module is located in a tamper-protected environment.

The TOE is used by a TSP (InfoCert itself or other) applying security policies as required by TSPs providing signature creation services. InfoCert is a Qualified Trust Service Provider that provides a TW4S as a service, meeting the higher standard on this issue. For more information see Ref. [10] and [11].

The Signer is located in its Local Environment where a user interface is available to the Signer on a device where he/she can display documents.

In its local environment, the Signer also has a component under his/her sole control by which he/she can authenticate him/herself. In order to complete the Signer authentication, at least two authentication factors of different classes are required, choosing between *what I know*, *what I have*, *what I am*. This component is a fundamental part of Signature Activation Protocol (SAP) in order to authenticate the Signer by means of an assertion that identifies the Signer and that will be used in the SAD generation.

This component (whose acronym is SIC) is usually a combination of software and hardware units. SIC could be an application executed by a browser, by a mobile device, or it could be a secure element of a mobile phone, a cryptographic device owned by the Signer or anything else that can be provided to the Signer.

The SIC communicates in a secure way with the External Authentication Service and obtains an assertion that identifies the Signer and that is provided to SSA. SSA is the application that shields the TOE and handles inward and outward messages.

The TOE verifies the SAD and if the verification succeeds, it authorizes the activation of the signing key within the Cryptographic Module (CM) in order to produce a digital signature value. The digital signature value is sent to the SSA and further delivered to the local application. Finally, the signed document can land on the Signer's user interface.

SSA forwards to the TOE two different kinds of messages

a. Signature requests coming from the local environment

b. Key pair generation requests during the enrolment process

The TOE keeps track of events by generating audit records.

The InfoCert TOE and TW4S rely also on other services:

- Identification and Registration Service: any Signer shall be identified and registered by the InfoCert Registration Authority. This operation can involve an authentication mechanism for the Signer.
- InfoCert Certification Authority (CA) Service: it issues certificates for the newly generated signing keys.
- Signature Creation Application (SCA) that is responsible for requesting one or more signatures and creating the signed document by using the signature values returned by the TW4S.

2.3.2 TOE Type

As defined in [6], the TOE is a combination of software and hardware components implementing a Signature Activation Module (SAM). The SAM then implements the Signature Activation Protocol (SAP), and it uses the Signature Activation Data (SAD) and Authorisation Data to activate

the signing key to be used in the Cryptographic Module. The TOE is a composite product consisting of a Cryptographic Module with a software application deployed within its tamper protected part, in a dedicated tamper-protected environment. The local nature of the communication between the software application and the Cryptographic Module within the same physically protected environment ensures integrity and confidentiality protection of the exchanged data, as well as mutual authentication of the communicating IT entities.

Together, the TRISS software application and the Cryptographic Module (i.e. the TOE) are a QSCD.

2.3.3 Usage and major security features

The major security features of the TOE are:

- Operator management:
  - Privileged Users [see 5.2 and 5.6] can create other Privileged Users.
- System management
  - Privileged Users can handle system configuration.
- Signer management covers:
  - Privileged Users can create Signers
  - Privileged Users set up the indirect authentication scheme that is assigned to all Signers.
  - Privileged Users or Signers can generate signing keys and signature Verification Data (SVD) using a Cryptographic Module and assign the signing key identifier and SVD to a Signer. Signer can generate signing keys only for him/herself.
  - Privileged Users or Signers can disable a signing key identifier to be used by a Signer. Signer can disable the own signing key identifier.
- Signature operations
  - Signers can supply a DTBS/R(s) to be signed, Privileged User can't.
  - The link between Signer authentication, DTBS/R(s) and signing key identifier is handled by the Signature Activation Data (SAD). The SSA securely exchange the SAD with the TOE by using the Signature Activation Protocol (SAP). The following actions are performed within the TOE:
    - The SAD is verified in integrity.
    - The SAD is verified that it binds together the Signer authentication, the DTBS/R(s) and the signing key identifier.
    - The Signer identified in the SAD is authenticated by using indirect authentication scheme only.
    - It is verified that the DTBS/R(s) used for signature operations is bound to the SAD.
    - It is verified that the signing key identifier is assigned to the Signer.

- The TOE uses Authorisation Data to activate the signing key within the Cryptographic Module.
- The TOE uses its Cryptographic Module services to create signatures.
- The TOE generates audit records for all security-related events [see chapter 8.1.1] and relies on the SSA to store and provide access control for the records.

The TOE handles data assets as specified in chapter 5.1.

2.3.4 Identification of the CM required by the TOE

As already stated in the introduction in 2.3, the TOE is a composite product which includes a CC-certified CM. The CM identification and certification data are the following:

Product Name	**nShield Solo XC Hardware Security Module**	
Product Version	12.60.15	
Identification Data	nC3025E-000 rev 06	nShield Solo XC F2. PCIe board
	nC4035E-000 rev 06	nShield Solo XC F3. PCIe board
	nC4335N-000 rev 06	nShield Solo XC for nShield Connect XC. This module is embedded in the nShield Connect XC appliance with model number NH2075-x or NH2089-x (where x is B, M or H). PCIe board embedded in nShield Connect XC appliance
Protection Profile	EN 419 221-5	
Evaluation Assurance Level	EAL4 augmented by AVA_VAN.5 and ALC_FLR.2	
Certificate Number	CC-21-0368256-MA	
Certificate Reference	[19]	
ST reference	nShield Solo XC HSM Security Target v1.1.1 [20] Document number LSEC0579	
Certificate Holder	Entrust	
Certification Body	NSCIB	

TABLE 3 – CM REFERENCE

2.3.5 Identification of non-TOE hardware/software/firmware required by the TOE

The TOE needs, at least, the following hardware/software/firmware to operate:

- A Signature Creation Application (SCA) that manages the document to be signed and transfers its DTBS/R to the SSA, either directly or through the user application.

The purpose of SCA is to manage a signature creation upon a digital object (i.e. a document) and to associate it to the document itself. SCA prepares the document for the signature request and handles the response shaping the required signature format. The Signer can choose several options according to the signature policy.

- A Server Signing Application (SSA) component that handles communications between the TOE and SIC in the Signer device.

SSA is an architectural server component located in a remote environment which is located between the external environment (i.e. internet) and the TOE. The principal goal of SSA is to shield the TOE applying the recommendation provided by PP EN 419 241 part. 1.

- A Signer's Interaction Component (SIC) is used locally by the Signer for authentication reasons. The TOE works with indirect authentication means according to the established requirements.
- An External Authentication Service (EAS) that verifies the Signer's authentication and issues an assertion stating that the Signer has been authenticated.

EAS is not part of the TOE and it is a service that can be provided by the QTSP as part of TW4S or delegated to a third party.

3 TOE DESCRIPTION (ASE_INT)

3.1 Physical scope

The composite TOE consists of a software application, named TRISS, which is installed and running on the Cryptographic Module nShield Solo XC operating system. It makes use of libraries supporting the HSM cryptographic functions. The Cryptographic Module is placed inside a tamper-protected environment, where there's also a cluster MongoDB database and part of SSA. In Figure 2 the TOE perimeter is highlighted in red color: it includes both the Cryptographic Module and the TRISS software application running inside its CPU.

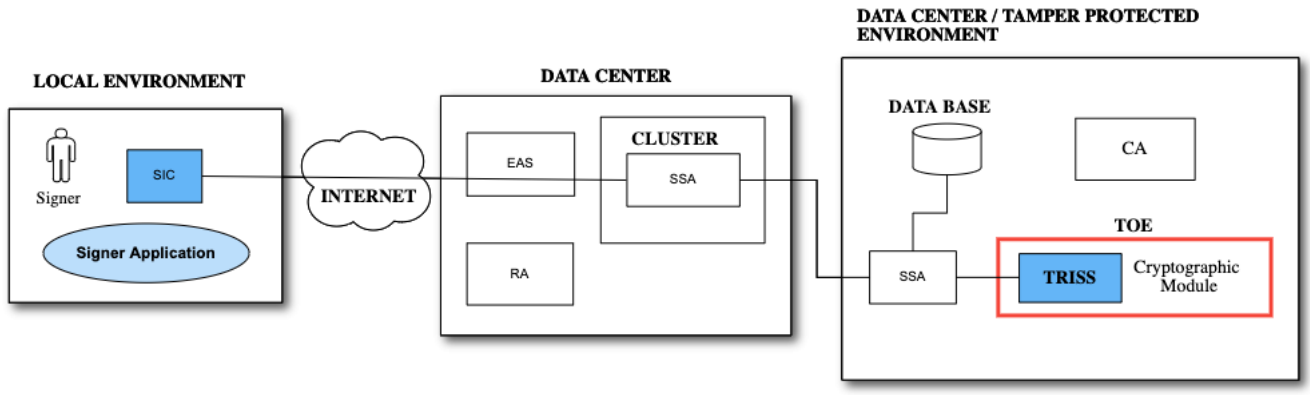


FIGURE 2 – PHYSICAL SCOPE

The TOE includes the following:

Type	Identifier	Description	Format	Delivery
Software	SHA-256 fingerprint of TRISS software application executable file	Software bundle	.zip	USB key, burned CD-ROM or external Hard Disk
Documentation	TRISS Security Target v2.2	Common Criteria Security Target	.pdf	Web download
Documentation	TRISS AGD_PRE v2.1	Common Criteria installation manual	.pdf	Encrypted email, download from documentation space with restricted access, or a shipped USB key containing the encrypted files

Documentation	TRISS AGD_OPE v2.1	Common Criteria operating manual	.pdf	Encrypted email, download from documentation space with restricted access, or a shipped USB key containing the encrypted files
Hardware	nC3025E-000 rev 06	CC-certified CM nShield Solo XC F2 v12.60.15	PCIe board	Courier (shipping by the CM vendor)
	nC4035E-000 rev 06	CC-certified CM nShield Solo XC F3 v12.60.15	PCIe board	
	nC4335N-000 rev 06	CC-certified CM nShield Solo XC for nShield Connect XC v12.60.15. This module is embedded in the nShield Connect XC appliance with model number NH2075-x or NH2089-x (where x is B, M or H)	PCIe board embedded in nShield Connect XC appliance	
Software	nShield Solo XC Firmware v12.60.15	Common Criteria Certified Firmware	.iso	Email, secure web download by the CM vendor
Software	CodeSafe v12.63.0	Tool for embedded TOE deployment on nShield HSMs	.iso	Email, secure web download by the CM vendor

TABLE 4 – TOE COMPONENTS

The TOE configuration can be viewed and modified by Privileged Users with a specific role.

The TOE is initially installed with default configuration settings that are accessible only to Privileged Users and that are stored into a static internal file. These settings can be modified by Privileged Users with specific role (e.g. issuers' SVDs for R.Reference_Signer_Authentication_Data verification). Each configuration change is performed under a controlled flow: there is no setting that may bring the TOE outside its CC-evaluated configuration.

The non-TOE parts include the following:

Type	Name and version	Description
Software	Security World Software with one of the	Security World Software for the CC-certified

	following versions: v12.60.11, v12.70.4, v12.71.0, v12.80.4, v12.81.2	CM
License	CodeSafe license	1-year license for using CodeSafe
License	ECC license	NCipher license to use elliptic curves

TABLE 5 – NON-TOE COMPONENTS

3.1.1 Life-cycle

This section is taken as is in section 3.3.2 of Protection Profile [6] with no refinements, additions or deletions.

The TOE life cycle consists of successive phase for development, production, preparation and operational use.

Development: The TOE developer develops the TOE application and its guidance documentation using any appropriate guidance documentation for components working with the TOE, including the Cryptographic Module.

Delivery: The TOE is securely delivered from the TOE developer to the TSP.

Installation and configuration: The TSP installs and configures the TOE with the appropriate configuration and initialisation data. Installation may allow creating the Privileged Users.

Operational phase: In operation, the TOE can be used by Privileged Users to create Privileged Users and Signers. Privileged Users can maintain TOE configuration. Privileged Users and Signers may generate signature keys for a Signer. Privileged Users and Signers can supply the data to be signed to the TOE, but only Signer can authorise a signature creation.

The TOE end of life is out of the scope of this document.

3.1.2 Delivery

The TOE is meant to be installed at the TSP (such as InfoCert itself) premises, in secure tamper-proof data center environments (CA bunker), compliant to the guidelines defined in [5] and [6]. The TOE services are available to SSA in the same tamper-protected environment. SSA components expose their endpoints to the public network, so that end-customers can integrate their own services on these endpoints, through HTTPS calls.

The complete Common Criteria documentation is shared with internal personnel of InfoCert and of the companies belonging to InfoCert group, either via encrypted email, or by an internal documentation space with restricted access, or via a shipped USB key containing the encrypted files. If there's a specific project requiring the TOE to be deployed at an external customer premises, Common Criteria guidance is shared under a license or with an NDA.

The delivery of TOE and documentation takes place over two distinct channels: the documentation with the SHA-256 of the TOE software application is provided on one channel (either via encrypted email or by downloading it from the InfoCert website), while the actual TOE software is given

through a second channel (e.g. on a USB key) for security reasons, with a secure delivery method performed by trusted personnel. It is mandatory for the user to verify that the SHA-256 provided with the documentation coincides with the one of the software before the TOE installation. The Cryptographic Module must be purchased directly from the vendor of nShield Solo XC model series (see Table 4 for the version).

A system administrator, allowed to access the CA bunker with a personal smartcard, is in charge of installing the TOE by following the guidelines describing a step-by-step deployment of the TOE software application embedded into the Cryptographic Module that are provided as part of the Common Criteria documentation. For completion, the installation procedure includes also cryptographic module's configuration, as well as database and firewall setting, and network ports opening. Once the installation procedure is complete, the TOE runs in a non-operational mode. An authorized Privileged User can later start-up the TOE to make it operative. Guidelines on the TOE initialization procedure are provided in the Common Criteria documentation as well.

3.2 Logical scope

3.2.1 Security audit

The TOE generates an audit record for every security function defined in this Security Target. Each record is created inside the TOE and it is returned outside by maintaining its integrity.

3.2.2 Cryptographic support

The TOE includes the cryptographic module. The CM with certified algorithms is used for a wide range of cryptographic functions including asymmetric keys generation and establishment, symmetric keys generation, encryption/decryption, cryptographic hashing, and keyed-hash message authentication. These functions are supported with suitable random bit generation, key derivation, salt generation, secure key storage, and protected data destruction. These primitive cryptographic functions are used to implement security protocols such as TLS, and also to encrypt Data-At-Rest (including the generation and protection of keys and key encryption keys) used by the TOE. Many of these cryptographic functions are also accessible as services to the applications which run on the TOE.

3.2.3 User data protection

The TOE is designed to control accesses to the system services by means of hosted applications. Additionally, the TOE is designed to protect the user and other sensitive data by using encryption so that even if a device is physically lost, the data remain protected.

3.2.4 Identification and authentication

The TOE supports several features related to identification and authentication. From a Privileged User perspective, a signed assertion must be correctly provided to unlock the TOE. If a Privileged User wants to access the TOE, a public key must be created and associated to the Privileged User in configuration data, an operation that can be made by another Privileged User with a specific

role.

3.2.5 Security management

The TOE provides all the interfaces necessary to manage the security functions identified throughout this Security Target. Many of these functions are available to the TOE users, while many others are restricted only to Privileged Users operating through a dedicated User Interface once the TOE has been enrolled.

3.2.6 Protection of the TSF

The TOE implements several features designed to protect itself and to ensure the reliability and integrity of its security features. Specifically, it protects sensitive data such as cryptographic keys so that they are not accessible outside the cryptographic module. It also uses a timing mechanism provided by the HSM real-time internal clock to provide reliable time information (e.g. for log accountability). It is also designed to protect itself from modification executed by applications as well as to isolate the address spaces of applications from one another to protect those applications. The TOE also includes mechanisms (i.e. the verification of the digital signature of each new image) so that the TOE itself can be updated while ensuring that the updates will not introduce malicious or other unexpected changes in the TOE. Digital signature checking also extends to verifying applications prior to their installation as all applications must have signatures (even if self-signed).

3.2.7 Trusted path/channels

The TOE can support the use of EAP-TLS, mutual TLS to secure communications channels between itself and other trusted network devices.

4 CONFORMANCE CLAIM (ASE_CCL)

4.1 CC Conformance claim

This Security Target claims conformance to version 3.1 (Revision 5) of the Common Criteria for Information Technology Security Evaluation.

The following conformance is claimed (Ref. [2]):

- Common Criteria for Information Technology Security Evaluation. Part 1: Introduction and general model, Version 3.1 Rev. 5;
- Common Criteria for Information Technology Security Evaluation, Part 2 extended: Security functional requirements, Version 3.1 Rev. 5;
- Common Criteria for Information Technology Security Evaluation, Part 3 conformant: Security assurance requirements, Version 3.1 Rev. 5.

4.2 PP Claim

This Security Target is strictly compliant to the Protection Profile prEN 419 241-2:2019 [6].

4.3 Package claim

This security target claims conformance to Evaluation Assurance Level EAL4, augmented with the following security assurance requirements defined in CC Part 3: **AVA_VAN.5** "Advanced methodical vulnerability analysis".

5 SECURITY PROBLEM DEFINITION (ASE_SPD)

The purpose of this section is to define the nature and scope of "security needs" to be addressed by the TOE. Therefore, this section will involve any assumptions that are made regarding the TOE's environment, the assets requiring protection, the identified threat agents and the threats they pose to the assets, and organizational security policies or rules with which the TOE complies in addressing the security needs.

In the following the assets, subjects and the threat agents will be defined first.

5.1 Assets

This section is largely taken from section 5.1 of Protection Profile [6] with some refinements in response to the Application Notes in the Protection Profile.

The TOE has the following assets, which are to be protected in integrity and confidentiality as described below. The TOE ensures that whenever an asset is persisted outside the TOE, the TOE performs the necessary cryptographic operations to enforce confidentiality and detect if an asset has been modified. Access control to TOE assets outside the TOE is enforced by the environment.

R.Signing_Key_Id: the signing key is the private key of an asymmetric key pair used to create a digital signature under the Signer's sole control. The signing key is used by the Cryptographic Module only. The TOE uses the asset R.Signing_Key_Id, which identifies a signing key in the Cryptographic Module. The binding of the R.Signing_Key_Id with R.Signer is protected in integrity.

The integrity and confidentiality of the signing key and the link between the R.Signing_Key_Id and the signing key is under the responsibility of the Cryptographic Module (the TOE). The TOE ensures that only the Signer can use the signing key under his sole control. The TOE manages different types of signing keys due to the usage scope:

a. One-time signing keys, which are generated, certified and used within a limited signing session. At the end of the session the signing key is destroyed;

b. Batch signing keys, only for some EU Member States, which are used to sign a batch of documents and the sole control is applied to the signing process, see [4];

c. Signing keys, which can be used for a given period or a given number of signatures.

The R.Signing_Key_Id scope is defined in the enrolment process and can't be changed later.

R.Authorisation_Data: is data used by the TOE to activate a signing key in the Cryptographic Module. The signing key is identified by R.Signing_Key_Id. R.Authorisation_Data is protected in integrity and confidentiality.

R.Authorisation_Data is verified by the Cryptographic Module to activate a signing key. The TOE has been implemented in order to use R.Authorisation_Data either as an asset or as a data derived from the SAD. In both cases, the TOE verifies the SAD before the R.Authorisation_Data is used to activate the signing key in the Cryptographic Module.

R.SVD: signature verification data is the public part, associated with the signing key, to perform digital signature verification. The R.SVD is protected in integrity.

The TOE uses its Cryptographic Module for signing key pair generation. As part of the signing key pair generation, Cryptographic Module creates R.Signing_Key_Id and R.SVD. The TOE provides the R.SVD to the SSA for further handling for the key pair to be certified.

R.DTBS/R: a set of data which is transmitted to the TOE for digital signature value creation on behalf of the Signer. The DTBS/R(s) is transmitted to the TOE. The R.DTBS/R is protected in integrity. The transmission of the DTBS/R(s) to the TOE requires the Signer to be authenticated.

NOTE: the confidentiality of the R.DTBS/R is not required by Regulation (EU) No 910/2014 [3].

R.SAD: signature activation data is a set of data involved in the signature activation protocol, which activates the signature creation data in order to create digital signature values under the Signer's sole control. The R.SAD combines:

- The Signer's strong authentication as specified in [EN 419 241-1]
- A unique reference to R.Signing_Key_Id
- The given R.DTBS/R(s)

The R.SAD is protected in integrity and confidentiality.

The R.SAD includes evidence from other systems (i.e. external authentication service) that Signer's authentication has been verified.

The unique reference to R.Signing_Key_Id in the SAD is a key identifier.

A person either natural or legal can have bound one or more R.Signing_Key_Id(s), each one linked to a pair of signing keys and to a Signer. The client application lists the certificate references (Signers) to this person before he/she uses one of them. The person selects one available signing key to sign the R.DTBS/R(s). A relation is established between the person identity and all the associated Signers / R.Signing_Key_Id(s).

One-time signing keys are generated, certified and used within a limited signing session. At the end of the session, the signing key is reliably destroyed.

The TOE handles one signing key for each signer, the reference to the R.Signing_Key_Id is unique and included into the SAD.

R.Signature: is the result of the signature operation and is a digital signature value. R.Signature is created on the R.DTBS/R, using the signing key identified by means of R.Signing_Key_Id, by the Cryptographic Module under the Signer's sole control as part of the SAP. The R.Signature is protected in integrity. The R.Signature can be verified outside the TOE using R.SVD.

R.Audit: it is the collection of records containing the logs that are registered during the events requiring to be audited (e.g. signature operation log). The logs are produced by the TOE and stored externally. The R.Audit is protected in integrity.

R.Signer: is a TOE subject containing the set of data that uniquely identifies the Signer within the TOE. The R.Signer is protected in integrity and it does not require encrypted data, hence no

part of R.Signer shall be protected in confidentiality.

The R.Signer is unique within the TOE. The TOE is not responsible for a connection between the R.Signer and the Signer's identity. The Signer is said to own the R.Signer object which uniquely identifies him within the TOE.

The R.Signer can be bound with zero or one R.Signing_Key_Id and related R.SVD.

Summing up the concept, each Signer's identity can be bound to one or more R.Signer objects that univocally identify the signing keys.

R.Reference_Signer_Authentication_Data: it refers to the set of data used by the TOE to authenticate the Signer. It contains all the data and keys (e.g. SVD) used by the TOE to authenticate the Signer. The SVD is needed to verify the assertion provided as a result of a delegated authentication.

The R.Reference_Signer_Authentication_Data is protected in integrity and confidentiality.

The R.Reference_Signer_Authentication_Data is used by the TOE to authenticate the Signer, and the R.Authorisation_Data is used by the TOE to activate a signing key in the Cryptographic Module as part of the TOE.

R.TSF_DATA: is the set of TOE configuration data used to operate the TOE. It is protected in integrity. The TOE configuration data includes cryptographic algorithms, key lengths, trusted certificate roots etc.

R.Privileged_User: is a TOE subject containing the set of data that uniquely identifies a Privileged User within the TOE. It is protected in integrity.

R.Reference_Privileged_User_Authentication_Data: it refers to the set of data used by the TOE to authenticate the Privileged User. It is protected in integrity. It is not required that R.Reference_Privileged_User_Authentication_Data is encrypted because the access to the TOE requires a public key owned by the Privileged User.

R.Random: is random secrets, e.g. keys, used by the TOE to operate and communicate with external parties. It is protected in integrity and confidentiality.

5.2 Subjects

This section is largely taken from section 5.2 of Protection Profile [6] with some refinements.

This following list of subjects interacts with the TOE.

- Signer, which is the natural or legal person who uses the TOE through the SAP where he/she provides the SAD in order to sign DTBS/R(s) using his/her signing key in the Cryptographic Module.
- Privileged User, which performs the administrative functions of the TOE.

The TOE implements specific roles for the Privileged User and the authorisation rules are described in the SFRs:

- ✓ System Administrators: are authorized to install, configure and maintain the TW4S but with controlled access to security-related information. It is also responsible for operating on a day-to-day basis including the TW4S system backup and recovery.
- ✓ System Operators: are responsible for operating the TW4S on a day-to-day basis and are authorized to perform system backup and recovery.

The SSA plays a special role as it interacts directly with the TOE. Privileged Users interact with the TOE via the SSA. If the SSA as a service can perform administrative functions, e.g. creating signer, this is in this PP considered as Privileged User.

The creation of Signers, management of reference Signer authentication data and signing key generation is carried out together with a Registration Authority (RA) providing a registration service using the SSA, as specified in ETSI EN 319 411-1 and ETSI EN 319 411-2.

5.3 Threats

This section is taken as is in section 5.3 of Protection Profile [6] with no refinements, additions or deletions.

The following threats are defined for the TOE. An attacker described in each of the threats is a subject that is not authorised for the relevant operation, but may present himself as an unknown user or as one of the other defined subjects.

5.3.1 Enrolment

The threats during enrolment are:

T.ENROLMENT_SIGNER_IMPERSONATION

An attacker impersonates Signer during enrolment. As examples, it could be:

- by transferring wrong R.Signer to TOE from RA
- by transferring wrong R.Reference_Signer_Authentication_Data to TOE from RA

The assets R.Signer and R.Reference_Signer_Authentication_Data are threatened.

Such impersonation may allow a potential incorrect Signer authentication leading to unauthorised signature operation on behalf of Signer.

T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_DISCLOSED

An attacker is able to obtain whole or part of R.Reference_Signer_Authentication_Data during enrolment. This can be during generation, storage or transfer to the TOE or transfer between Signer and TOE. As examples it could be:

- by reading the data
- by changing the data, e. g. to a known value

The asset R.Reference_Signer_Authentication_Data is threatened.

Such data disclosure may allow a potential incorrect Signer authentication leading to unauthorised signature operation on behalf of Signer.

The threats on enrolment are threats to the environment in case external authentication is supported by the TOE.

T.SVD_FORGERY

An attacker modifies the R.SVD during transmission to the RA or CA. This results in loss of R.SVD integrity in the binding of R.SVD to signing key and to R.Signer.

The asset R.SVD is threatened.

If the CA relies on the generation of the key pair controlled by the TOE as specified in [ETSI EN 319 411-1] clause 6.3.3 d) then an attacker can forge signatures masquerading as the Signer.

There is a secure transport of R.SVD from TOE to RA or CA. The SAM produces a CSR.

5.3.2 Signer Management

T.ADMIN_IMPERSONATION

Attacker impersonates a Privileged User and updates R.Reference_Signer_Authentication_Data, R.Signing_Key_Id or R.SVD.

The assets R.Reference_Signer_Authentication_Data, R.SVD and R.Signing_Key_Id are threatened.

Such data modification may allow a potential incorrect Signer authentication leading to unauthorised signature operation on behalf of Signer.

T.MAINTENANCE_AUTHENTICATION_DISCLOSE

Attacker discloses or changes (e.g. to a known value) R.Reference_Signer_Authentication_Data during update and is able to create a signature.

The assets R.Reference_Signer_Authentication_Data and R.Signing_Key_Id are threatened.

Such data disclosure may allow a potential incorrect Signer authentication leading to unauthorised signature operation on behalf of Signer.

5.3.3 Usage

This section describes threats for signature operation including authentication.

T.AUTHENTICATION_SIGNER_IMPERSONATION

An attacker impersonates Signer using forged R.Reference_Signer_Authentication_Data and

transmits it to the TOE during SAP and uses it to sign the same or modified DTBS/R(s).

The assets R.Reference_Signer_Authentication_Data, R.SAD and R.Signing_Key_Id are threatened.

T.SIGNER_AUTHENTICATION_DATA_MODIFIED

An attacker is able to modify R.Reference_Signer_Authentication_Data inside the TOE or during maintenance.

The asset R.Reference_Signer_Authentification_Data is threatened.

Such data modification may allow a potential incorrect Signer authentication leading to unauthorised signature operation on behalf of Signer.

T.SAP_BYPASS

An attacker bypasses one or more steps in the SAP and is able to create a signature without the Signer having authorised the operation.

The asset R.SAD is threatened.

T.SAP_REPLAY

An attacker replays one or more steps of SAP and is able to create a signature without the Signer having authorised the operation.

The asset R.SAD is threatened.

T.SAD_FORGERY

An attacker forges or manipulates R.SAD during transfer in SAP and is able to create a signature without the Signer having authorised the operation.

The asset R.SAD is threatened.

T.SIGNATURE_REQUEST_DISCLOSURE

An attacker obtains knowledge of R.DTBS/R or R.SAD during transfer to TOE.

The assets R.DTBS/R and R.SAD are threatened.

If the R.DTBS/R or R.SAD do not require encrypted data then this threat is mitigated.

T.DTBSR_FORGERY

An attacker modifies R.DTBS/R during transfer to TOE and is able to create a signature on this modified R.DTBS/R without the Signer having authorised the operation on this R.DTBS/R.

The asset R.DTBS/R is threatened.

T.SIGNATURE_FORGERY

An attacker modifies R.Signature during or after creation or during transfer outside the TOE.

The asset R.Signature is threatened.

The modification of a signature can be detected by the SSA or any relying party by validation of the signature.

5.3.4 System

T.PRIVILEGED_USER_INSERTION

An attacker is able to create R.Privileged_User including R.Reference_Privileged_User_Authentication_Data and is able to log on to the TOE as a Privileged User.

The assets R.Privileged_User and R.Reference_Privileged_User_Authentication_Data are threatened.

T.REFERENCE_PRIVILEGED_USER_AUTHENTICATION_DATA_MODIFICATION

An attacker modifies R.Reference_Privileged_User_Authentication_Data and is able to log on to the TOE as the Privileged User.

The asset R.Reference_Privileged_User_Authentication_Data is threatened.

T.AUTHORISATION_DATA_UPDATE

Attacker impersonates Privileged User and updates R.Authorisation_Data and may be able to activate a signing key.

The assets R.Authorisation_Data and R.Signing_Key_Id are threatened.

T. AUTHORISATION_DATA _DISCLOSE

Attacker discloses R.Authorisation_Data during update and is able to activate a signing key. The assets R.Authorisation_Data and R.Signing_Key_Id are threatened.

T.CONTEXT_ALTERATION

An attacker modifies system configuration R.TSF_DATA in order to perform an unauthorised operation.

The assets R.Signing_Key_Id, R.SVD, R.SAD, R.Reference_Signer_Authentication_Data and R.TSF_DATA are threatened.

T.AUDIT_ALTERATION

An attacker modifies system audit and is able to hide trace of TOE modification or usage.

The assets R.SVD, R.SAD, R.Signer, R.Reference_Signer_Authentication_Data, R.DTBS/R, R.Signature, R.AUDIT and R.TSF_DATA are threatened.

T.RANDOM

An attacker is able to guess system secrets R.RANDOM and able to create or modify TOE objects or participate in communication with external systems.

5.4 Relation between threats and assets

This section is taken as is in section 5.4 of Protection Profile [6] with no refinements, additions or deletions.

This following table provides an overview of the relationships between asset, associated security properties and threats. For details consult the individual threats in the previous sections.

Asset	Security Dimensions	Threats
R.Signing_Key_Id	Integrity	T.ADMIN_IMPERSONATION T.MAINTENANCE_AUTHENTICATION_DISCLOSE T.AUTHENTICATION_SIGNER_IMPERSONATION T.CONTEXT_ALTERATION T.AUTHORISATION_DATA_UPDATE
	Confidentiality	T.AUTHORISATION_DATA_UPDATE T.AUTHORISATION_DATA _DISCLOSE
R.Authorisation_Data	Integrity	T.AUTHORISATION_DATA_UPDATE
	Confidentiality	T.AUTHORISATION_DATA_UPDATE T. AUTHORISATION_DATA _DISCLOSE
R.SVD	Integrity	T.SVD_FORGERY T.ADMIN_IMPERSONATION T.CONTEXT_ALTERATION T.AUDIT_ALTERATION
R.DTBS/R	Integrity	T.SIGNATURE_REQUEST_DISCLOSURE T.DTBSR_FORGERY T.AUDIT_ALTERATION
	Origin authentication	T.DTBSR_FORGERY

Asset	Security Dimensions	Threats
R.SAD	Integrity	T.AUTHENTICATION_SIGNER_IMPERSONATION T.CONTEXT_ALTERATION T.AUDIT_ALTERATION T.SAP_BYPASS T.SAP_REPLAY T.SAD_FORGERY T.SIGNATURE_REQUEST_DISCLOSURE
	Confidentiality	T.AUTHENTICATION_SIGNER_IMPERSONATION T.CONTEXT_ALTERATION
R.Signature	Integrity	T.SIGNATURE_FORGERY T.AUDIT_ALTERATION
R.Audit	Integrity	T.AUDIT_ALTERATION
R.Signer	Integrity	T.ENROLMENT_SIGNER_IMPERSONATION T.AUDIT_ALTERATION
R.Reference_Signer_Authentication_Data	Integrity	T.ENROLMENT_SIGNER_IMPERSONATION T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_DISCLOSED T.SIGNER_AUTEHNTICATION_DATA_MODIFIED T.ADMIN_IMPERSONATION T.MAINTENANCE_AUTHENTICATION_DISCLOSE T.AUTHENTICATION_SIGNER_IMPERSONATION T.CONTEXT_ALTERATION T.AUDIT_ALTERATION
	Confidentiality	T.ENROLMENT_SIGNER_IMPERSONATION T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_DISCLOSED T.SIGNER_AUTEHNTICATION_DATA_MODIFIED T.ADMIN_IMPERSONATION T.MAINTENANCE_AUTHENTICATION_DISCLOSE T.AUTHENTICATION_SIGNER_IMPERSONATION T.CONTEXT_ALTERATION
R.Privileged_User	Integrity	T.PRIVILEGED_USER_INSERTION
R.Reference_Privileged_User_Authentication_Data	Integrity	T.PRIVILEGED_USER_INSERTION T.REFERENCE_PRIVILEGED_USER_AUTHENTICATION_DATA_MODIFICATION
	Confidentiality	T.PRIVILEGED_USER_INSERTION T.REFERENCE_PRIVILEGED_USER_AUTHENTICATION_DATA_MODIFICATION
R.RANDOM	Integrity	T.RANDOM
	Confidentiality	T.RANDOM
R.TSF_DATA	Integrity	T.CONTEXT_ALTERATION T.AUDIT_ALTERATION

TABLE 6 - THREATS VS ASSETS

5.5 Organizational Security Policies

This section is largely taken from section 5.5 of Protection Profile [6] with some refinements.

The TOE complies with the following Organizational Security Policies (OSP) as security rules, procedures, practices, or guidelines imposed by an organization upon its operations.

OSP.RANDOM

The TOE generates random numbers that meet a specific quality metric. These random numbers are suitable for use as keys, authentication/authorisation data, or seed data for another random number generator that is used for these purposes.

OSP.CRYPTO

The TOE uses algorithms, algorithm parameters and key lengths endorsed by recognized authorities as appropriate by TSPs. This includes generation of random numbers, signing key pairs and signatures as well as the integrity and confidentiality of TOE assets. The TOE adopts algorithms listed in the document [12].

5.6 Assumptions

This section is largely taken from section 5.6 of Protection Profile [6] with some refinements.

A.PRIVILEGED_USER

It is assumed that all personnel administering the TOE is trusted, competent and possesses the resources and skills required for his/her tasks and is trained to conduct the activities he/she is responsible for.

A.SIGNER_ENROLMENT

The signer shall be enrolled and certificates managed in conformance with the regulations given in [eIDAS]. Guidance for how to implement an enrolment and certificate management system in conformance with [eIDAS] are given in e.g. [EN 319 411-1] or for qualified certificate in e.g. [EN 319 411-2]. The TSP deploying and managing the TOE is a qualified TSP supervised/accredited for issuing qualified certificates according to article 3 of Regulation (EU) No 910/2014 and audited to be compliant with the requirements stated in ETSI EN 319 401, EN 319 411-1 and EN 319 411-2.

A.SIGNER_AUTHENTICATION_DATA_PROTECTION

It is assumed that the Signer will not disclose his authentication factors.

A.SIGNER_DEVICE

It is assumed that the device and SIC used by Signer to interact with the SSA and the TOE is under the Signer's control for the signature operation, i.e. protected against malicious code.

A.CA

It is assumed that the qualified TSP that issues qualified certificates is compliant with the relevant requirements for qualified TSP's as defined in [3].

A.ACCESS_PROTECTED

It is assumed that the TOE operates in a protected environment that limits physical access to the TOE to authorised Privileged Users. The TOE software and hardware environment (including client applications) is installed and maintained by Privileged Users in a secure state that mitigates against the specific risks applicable to the deployment environment.

It is assumed that any audit record generated by the TOE is only handled by authorised personnel in a physically secured environment. The personnel that carries these activities acts under established practices.

Any audit generated by the TOE does not allow signing keys to be used and any information needed to activate a signing key remains protected in integrity and confidentiality.

It is assumed that where copies of data protected by the TOE are managed outside of the TOE, client applications and other entities must provide appropriate protection for that data to a level required by the application context and the risks in the deployment environment.

The only data that can be managed outside the TOE is the Audit log.

A.AUTH_DATA

It is assumed that the SAP is designed in such a way that the activation of the signing key is under the sole control of the signer with a high level of confidence. If SAD is received by the TOE, it must be assumed that the SAD was submitted under the full control of the signer by means that are in possession of the signer.

A.TSP_AUDITED

It is assumed that the TSP deploying the SSA and the TOE is a qualified TSP according to article 3 (20) of Regulation (EU) No 910/2014 [3] and audited to be compliant with the requirements for TSP's given by [3].

A.SEC_REQ

It is assumed that the TSP establishes an operating environment according to the security requirements for SCAL2 defined in EN 419 241-1 [4]

6 SECURITY OBJECTIVES (ASE_OBJ)

This section is largely taken from chapter 6 of Protection Profile [6] with some refinements in response to application notes in the Protection Profile.

This section identifies and defines the security objectives for the TOE and its operational environment. These security objectives reflect the stated intent and counter the identified threats, as well as comply with the identified organisational security policies and assumptions.

6.1 Security Objectives for the TOE

The security objectives for the TOE determine (to the desired extent) the responsibility of the TOE in countering the threats and in supporting the OSPs. Each objective must be traced back to aspects of identified threats to be countered by the TOE and to aspects of OSPs to be met by the TOE. The security objectives may be viewed as providing the reader with a link from the identified security needs to the IT security requirements.

6.1.1 Enrolment

OT.SIGNER_PROTECTION

The TOE ensures that data associated to R.Signer are protected in integrity.

OT.REFERENCE_SIGNER_AUTHENTICATION_DATA

The TOE is able to securely handle signature authentication data, R.Reference_Signer_ Authentication_Data, as part of R.Signer.

OT.SIGNER_KEY_PAIR_GENERATION

The TOE is able to securely use the Cryptographic Module to generate Signer signing key pairs and assign R.Signing_Key_Id and R.SVD to R.Signer.

OT.SVD

The TOE ensures that the R.SVD linked to R.Signer is not modified before it is certified.

6.1.2 User Management

OT.PRIVILEGED_USER_MANAGEMENT

The TOE ensures that any modification to R.Privileged_User and R.Reference_Privileged_User_Authentication_Data are performed under control of a Privileged User.

OT.PRIVILEGED_USER_AUTHENTICATION

The TOE ensures that an administrator with a Privileged User is authenticated before any action on the TOE is performed.

The exception to this objective is when the initial set of Privileged Users are created as part of system initialisation.

OT.PRIVILEGED_USER _PROTECTION

The TOE ensures that data associated to R.Privileged_User are protected in integrity.

OT.SIGNER_MANAGEMENT

The TOE ensures that any modification to R.Signer, R.Reference_Signer_Authentication_Data, R.Signing_Key_Id and R.SVD are performed under control of the Signer or Privileged User.

6.1.3 Usage

OT.SAD_VERIFICATION

The TOE verifies the SAD. That is, it checks there is a link between the SAD elements and ensures the Signer is strongly authenticated. The R.Authorisation_Data is controlled by the Cryptographic Module.

Requirements for authentication are described in [EN 419 241-1] SRA_SAP.1.1.

OT.SAP

The TOE implements the server-side endpoint of a Signature Activation Protocol (SAP), which provides the following:

- Signer authentication
- Integrity of the transmitted SAD
- Confidentiality of at least the elements of the SAD which contains sensitive information
- Protection against replay, bypass of one or more steps and forgery

Signer authentication is conducted according to [4] SCAL2 for qualified signatures. Signer authentication is carried out in the following way:

- Indirectly by the TOE. An external authentication service as part of the TW4S or a delegated party that verifies the Signer's authentication factor(s) and issues an assertion that the Signer has been authenticated. The TOE verifies the assertion.

OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION

The TOE ensures signature authentication data is protected against attacks when transmitted to the TOE which would compromise its use for authentication.

OT.DTBSR_INTEGRITY

The TOE ensures that the R.DTBS/R is protected in integrity when transmitted to the TOE.

OT.SIGNATURE_INTEGRITY

The TOE ensures that a signature can't be modified inside the TOE.

OT.CRYPTO

The TOE only uses algorithms, algorithm parameters and key lengths endorsed by recognized authorities. This includes generation of random numbers, signing key pairs and signatures as well as the integrity and confidentiality of TOE assets.

6.1.4 System

OT.RANDOM

Random numbers generated used by the TOE for use as keys, in protocols or seed data for another random number generator that is used for these purposes shall meet a defined quality metric in order to ensure that random numbers are not predictable and have enough entropy.

OT.SYSTEM_PROTECTION

The TOE ensures that modification of R.TSF_DATA is authorised by Privileged User and that unauthorized modification can be detected.

OT.AUDIT_PROTECTION

The TOE ensures that modifications to R.AUDIT can be detected.

6.2 Security Objectives for the operational environment

OE.SVD_AUTHENTICITY

The operational environment ensures the SVD integrity during transit outside the TOE to the CA.

OE.CA_REQUEST_CERTIFICATE

The operational environment ensures that the qualified TSP that issues qualified certificates is compliant with the relevant requirements for qualified TSP's as defined in [3].

The operational environment uses a process for requesting a certificate, including SVD and Signer information, and CA signature in a way, which demonstrates the Signer is in control of the signing key associated with the SVD presented for certification. The integrity of the request shall be protected.

OE.CERTIFICATE_VERIFICATION

The operational environment verifies that the certificate for the R.SVD contains the R.SVD.

OE.SIGNER_AUTHENTICATION_DATA

The Signer’s management of authentication factors data outside the TOE is carried out in a secure manner.

OE.DELEGATED_AUTHENTICATION

If the TOE has support for and is configured to use delegated authentication then the TSP deploying the SSA and TOE shall ensure that all requirements in [EN 419 241-1] SRA_SAP.1.1 are met.

In addition, the QTSP ensures that:

- the delegated party fulfils all the relevant requirements of this standard and the requirements for registration according to the Regulation (EU) No 910/2014 [3], or
- the authentication process delegated to the external party uses an electronic identification means issued under a notified scheme that is included in the list published by the Commission pursuant to Article 9 of the Regulation (EU) No 910/2014 [3]

If the Signer is only authenticated using a delegated party, the TSP ensures that the secret key material used to authenticate the delegated party to the TOE resides in a certified cryptographic module consistent with the requirement as defined in [4] SRG_KM.1.1.

The audit of the QTSP, according to EN 419 241-1, ensures that any delegated party meets requirements from EN 419 241-1 SRA_SAP.1.1. and optionally SRG_KM.1.1 in case the Signer is only authenticated using a delegated party.

OE.DEVICE

The device, computer/tablet/smartphone containing the SIC and which is used by the Signer to interact with the TOE is protected against malicious code. It participates using SIC as local part of the SAP and may calculate SAD as described in [4]. It may be used to view the document to be signed.

OE.ENV

The TSP deploying the SSA and TOE is a qualified TSP according to article 3 (20) of Regulation (EU) No 910/2014 [3] and audited to be compliant with the requirements for TSP's given by [3]. The audit of the qualified TSP shall cover the security objectives for the operational environment specified in this clause. Generating or managing electronic signature creation data on behalf of the signatory may only be done by a qualified trust service provider.

The TOE operates in a protected environment that limits physical access to the TOE to authorised privileged users. The TOE software and hardware environment (including client applications) is installed and maintained by Administrators in a secure state that mitigates against the specific risks applicable to the deployment environment, including (when applicable):

- Protection against loss or theft of the TOE or any of its externally stored assets
- Inspections to deter and detect tampering (including attempts to access side-channels, or to access connections between physically separate parts of the TOE, or parts of the hardware appliance)
- Protection against the possibility of attacks based on emanations from the TOE (e.g. electromagnetic emanations) according to risks assessed for the operating environment
- Protection against unauthorised software and configuration changes on the TOE and the hardware appliance
- Protection to an equivalent level of all instances of the TOE holding the same assets (e.g. where a key is present as a backup in more than one instance of the TOE).

OE.CRYPTOMODULE_CERTIFIED

The TOE is implemented as a local application within the same physical boundary as the cryptographic module defined in [5], hence the TOE relies on the cryptographic module for cryptographic functionality and random number generation. The physical boundary physically protects the TOE conformant to FPT_PHP.1 and FPT_PHP.3 in [5].

Application Note CE (ST): the TOE is a composite product which includes the cryptographic module in addition to the local application. The cryptographic module identification data are outlined in paragraph 2.3.4.

OE.TW4S_CONFORMANT

The TOE is operated by a qualified TSP in an operating environment conformant with [4].

6.2.1 Security Problem Definition and Security Objectives

The following tables map security objectives with the security problem definition.

Table 6: Threats (T) vs TOE Security Objectives (OT)

Table 7: Organization Security Policies (OSP) vs TOE Security Objectives (OT)

Table 8: Threats (T) vs Security Objectives for the Environment (OE)

Table 9: Organization Security Policies (OSP) and Assumptions (A) vs Security Objectives for the Environment (OE)

TABLE 7 - THREATS VS TOE SECURITY OBJECTIVES

	Enrolment	OT.SIGNER_PROTECTION	OT.REFERENCE_SIGNER_AUTHENTICATION_DATA	OT.SIGNER_KEY_PAIR_GENERATION	OT.SVD	User Management	OT.PRIVILEGED_USER_MANAGEMENT	OT.PRIVILEGED_USER_AUTHENTICATION	OT.PRIVILEGED_USER_PROTECTION	OT.SIGNER_MANAGEMENT	System	OT.RANDOM	OT.SYSTEM_PROTECTION	OT.AUDIT_PROTECTION	Usage	OT.SAD_VERIFICATION	OT.SAP	OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION	OT.DTBSR_INTEGRITY	OT.SIGNATURE_INTEGRITY	OT.CRYPTO
Enrolment																					

	Enrolment	OT.SIGNER_PROTECTION	OT.REFERENCE_SIGNER_AUTHENTICATION_DATA	OT.SIGNER_KEY_PAIR_GENERATION	OT.SVD	User Management	OT.PRIVILEGED_USER_MANAGEMENT	OT.PRIVILEGED_USER_AUTHENTICATION	OT.PRIVILEGED_USER_PROTECTION	OT.SIGNER_MANAGEMENT	System	OT.RANDOM	OT.SYSTEM_PROTECTION	OT.AUDIT_PROTECTION	Usage	OT.SAD_VERIFICATION	OT.SAP	OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION	OT.DTBSR_INTEGRITY	OT.SIGNATURE_INTEGRITY	OT.CRYPTO
T.ENROLMENT_SIGNER_IMPERSONATION		X	X							X											
T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_DISCLOSED		X	X																		
T.SVD_FORGERY				X	X																X
Signer Management																					
T.ADMIN_IMPERSONATION								X		X											
T.MAINTENANCE_AUTHENTICATION_DISCLOSE			X																		
Usage																					
T.AUTHENTICATION_SIGNER_IMPERSONATION																X					
T.SIGNER_AUTHENTICATION_DATA_MODIFIED			X														X	X			
T.SAP_BYPASS																	X				
T.SAP_REPLAY																	X				
T.SAD_FORGERY																	X	X			
T.SIGNATURE_REQUEST_DISCLOSURE																	X				
T.DTBSR_FORGERY																			X		
T.SIGNATURE_FORGERY																				X	X
System																					
T.PRIVILEGED_USER_INSERTION							X	X													
T.REFERENCE_PRIVILEGED_USER_AUTHENTICATION_DATA_MODIFICATION							X	X	X												
T.AUTHORISATION_DATA_UPDATE													X								
T.AUTHORISATION_DATA_DISCLOSE													X								
T.CONTEXT_ALTERATION													X								
T.AUDIT_ALTERATION														X							

	Enrolment	OT.SIGNER_PROTECTION	OT.REFERENCE_SIGNER_AUTHENTICATION_DATA	OT.SIGNER_KEY_PAIR_GENERATION	OT.SVD	User Management	OT.PRIVILEGED_USER_MANAGEMENT	OT.PRIVILEGED_USER_AUTHENTICATION	OT.PRIVILEGED_USER_PROTECTION	OT.SIGNER_MANAGEMENT	System	OT.RANDOM	OT.SYSTEM_PROTECTION	OT.AUDIT_PROTECTION	Usage	OT.SAD_VERIFICATION	OT.SAP	OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION	OT.DTBSR_INTEGRITY	OT.SIGNATURE_INTEGRITY	OT.CRYPTO
T.RANDOM												X									

TABLE 8 - TOE SECURITY OBJECTIVES AND ORGANIZATIONAL SECURITY POLICIES

	Enrolment	OT.SIGNER_PROTECTION	OT.REFERENCE_SIGNER_AUTHENTI CATION_DATA	OT.SIGNER_KEY_PAIR_GENERATION	OT.SVD	OT.RANDOM	OT.CRYPTO
OSP.RANDOM						X	
OSP.CRYPTO							X

TABLE 9 - THREATS VS SECURITY OBJECTIVES FOR THE ENVIRONMENT

	OE.SVD_AUTHENTICITY	OE.CA_REQUEST_CERTIFICATE	OE.SIGNER_AUTHENTICATION_DATA	OE.DEVICE	OE.ENV	OE.CRYPTOMODULE_CERTIFIED	OE.TW4S_CONFORMANT
Enrolment							
T.ENROLMENT_SIGNER_IMPERSONATION							X
T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_DISCLOSED			X	X			
T.SVD_FORGERY	X	X					
Signer Management							
T.ADMIN_IMPERSONATION							
T.MAINTENANCE_AUTHENTICATION_DISCLOSE							
Usage							
T.AUTHENTICATION_SIGNER_IMPERSONATION							
T.SIGNER_AUTHENTICATION_DATA_MODIFIED							
T.SAP_BYPASS				X			
T.SAP_REPLAY				X			
T.SAD_FORGERY			X	X			
T.DTBSR_FORGERY				X			
T.SIGNATURE_FORGERY							
System							
T.PRIVILEGED_USER_INSERTION							
T.REFERENCE_PRIVILEGED_USER_AUTHENTICATION_DATA_MODIFICATION							
T.AUTHORISATION_DATA_UPDATE							
T.AUTHORISATION_DATA_DISCLOSE							
T.CONTEXT_ALTERATION							
T.AUDIT_ALTERATION							

TABLE 10 - ORGANIZATIONAL SECURITY POLICIES AND ASSUMPTIONS VS SECURITY OBJECTIVES FOR THE ENVIRONMENT

	OE.SVD_AUTHENTICITY	OE.CA_REQUEST_CERTIFICATE	OE.SIGNER_AUTHENTICATION_DATA	OE.DEVICE	OE.ENV	OE.CRYPTOMODULE_CERTIFIED	OE.TW4S_CONFORMANT
Organisational Security Policies							
~~OSP.TSP_AUDITED~~[1]							~~X~~
OSP.RANDOM							
OSP.CRYPTO						X	
Assumptions							
A.PRIVILEGED_USER							X
A.SIGNER_ENROLMENT					X		
A.SIGNER_AUTHENTICATION_DATA_PROTECTION			X				
A.SIGNATURE_REQUEST_DISCLOSURE				X			
A.SIGNER_DEVICE				X			
A.CA		X					
A.ACCESS_PROTECTED					X		
A.AUTH_DATA				X			
A.TSP_AUDITED					X		
A.SEC_REQ							X

[1] OSP.TSP_AUDITED has been struck-through as it is an editorial mistake present in [6]. This OSP is actually not defined in section 5.5 of [6] and in section 6.3.2 of this Security Target, though the table in [6] contains this entry.

6.3 Rationale for the security objectives

This section provides the Rationale Objectives and covers each threat, organizational security policy and assumption.

6.3.1 Threats and objectives

T.ENROLMENT_SIGNER_IMPERSONATION

T.ENROLMENT_SIGNER_IMPERSONATION is covered by OT.SIGNER_PROTECTION requiring R.Signer to be protected in integrity and for sensitive parts in confidentiality.

It is also covered by OT.SIGNER_MANAGEMENT requiring the signer to be securely created.

It is also covered by OT.REFERENCE_SIGNER_AUTHENTICATION_DATA requiring the TOE to be able to assign signer authentication data to the Signer.

It is also covered by OE.TW4S_CONFORMANT as that requires Signer enrolment to be handled in accordance with [17] for level at least substantial.

Application Note BI (ST): after the Signer's identification, the request made by the SSA to the TOE containing R.Signer and R.Reference_Signer_Authentication_Data is protected in integrity and R.Reference_Signer_Authentication_Data is also protected in confidentiality through a TLS connection. The TOE verifies the integrity of both of them and proceeds with the creation of the signature only if the integrity is verified.

T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_DISCLOSED

T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_DISCLOSED is covered by OT.REFERENCE_SIGNER_AUTHENTICATION_DATA requiring that authentication data be securely handled.

It is also covered by OT.SIGNER_PROTECTION requiring that the attributes, including Signer authentication data, be protected in integrity and if needed in confidentiality.

It is also covered by OE.SIGNER_AUTHENTICATION_DATA requiring the Signer to keep his authentication data secret.

It is also covered by OE.DEVICE requiring the device used by the Signer not to disclose authentication data.

Application Note BJ (ST): R.Reference_Signer_Authentication_Data is protected in integrity and confidentiality through TLS connection from SIC to SSA.

The Privileged User is authenticated indirectly by the verification of his/her signed assertion and is authorized after checking his/her roles. The Privileged User can modify R.Reference_Signer_Authentication_Data and R.TSF_DATA.

The Signer is authenticated before any activity and can modify only R.Authorisation_Data as asset of TW4S. It is not allowed for a Signer to modify any other data.

It is assumed that the device is protected against malicious code.

T.SVD_FORGERY

T.SVD_FORGERY is covered by OT.SIGNER_KEY_PAIR_GENERATION requiring a Cryptographic Module to generate Signer key pair.

It is also covered by OT.SVD requiring the SVD to be protected while inside the TOE.

It is also covered by OT.CRYPTO requiring the usage of endorsed algorithms. It is also covered by OE.SVD_AUTHENTICITY requiring the environment to protect the SVD during transit from the TOE to the CA.

It is also covered by OE.CA_REQUEST_CERTIFICATE requiring the certification request to be protected in integrity.

Application Note BK (ST): the TOE sends the request for generating the signing key to the Cryptographic Module, which returns the R.Signing_Key_Id. The TOE saves it ensuring the correspondence of R.Signer and R.Signing_Key_Id. The TOE returns the public key in the form of a CSR to the RA and the RA then requests the certificate to the CA. The transmission is secured with a TLS connection.

T.ADMIN_IMPERSONATION

T.ADMIN_IMPERSONATION is covered by OT.SIGNER_MANAGEMENT and OT.PRIVILEGED_USER_AUTHENTICATION requiring any changes to the Signer representation and attributes are carried out in an authorised manner.

Application Note BL (ST): Privileged User is authenticated indirectly by the verification of his/her signed assertion and is authorized after his/her roles checking. Each role is bound to one or more functions available to the Privileged User. No activities are permitted to Privileged Users before authentication and authorisation. The Privileged User can modify R.Reference_Signer_Authentication_Data and R.TSF_DATA.

The Signer is authenticated before any activity and can modify only its own R.Authorisation_Data as component of TW4S. It isn't allowed to a Signer to modify any other data like R.Signing_Key_ID, R.SVD and R.Signer.

T.MAINTENANCE_AUTHENTICATION_DISCLOSE

T.MAINTENANCE_AUTHENTICATION_DISCLOSE is covered by OT.REFERENCE_SIGNER_AUTHENTICATION_DATA requiring that authentication data be securely handled.

Application Note BM (ST): R.Reference_Signer_Authentication_Data can be modified only by a Privileged User. The Privileged User acts inside a tampered environment and authenticates before any operation. Also, he must have a specific role (APPLICATION-OPERATOR) to modify the R.Reference_Signer_Authentication_Data.

T.AUTHENTICATION_SIGNER_IMPERSONATION

T.AUTHENTICATION_SIGNER_IMPERSONATION is covered by OT.SAD_VERIFICATION requiring that the TOE checks the SAD received in the SAP.

Application Note BN (ST): the TOE verifies the SAD in input and the integrity of the elements bound to it. The SAD has RSA or ECDSA signature, which the TOE verifies against the EAS SVD included in a white list into R.TSF_DATA.

T.SIGNER_AUTHENTICATION_DATA_MODIFIED

T.SIGNER_AUTHENTICATION_DATA_MODIFIED is covered by OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION requiring the SAD transported protected in the SAP.

It is also covered by OT.REFERENCE_SIGNER_AUTHENTICATION_DATA requiring that authentication data be securely handled.

It is also covered by OT.SAP requiring the integrity of the SAD is protected during transmit in the SAP.

Application Note BO (ST): the R.SAD is verified in integrity together with its bound elements. R.Reference_Signer Authentication_Data is checked before the usage of R.Authorisation_Data.

T.SAP_BYPASS

T.SAP_BYPASS is covered by OT.SAP requiring that all steps, including SAD verification, of the SAP must completed.

It is also covered by OE.DEVICE requiring the SIC to participate in the SAP.

Application Note BP (ST): the TOE verifies the SAD integrity and validity, and it checks the bound elements before performing any other operation.

T.SAP_REPLAY

T.SAP_REPLAY is covered by OT.SAP requiring that the signature activation protocol must be able to resist whole or part of it being replayed.

It is also covered by OE.DEVICE requiring the SIC to participate in the SAP.

Application Note BQ (ST): the TOE keeps the SAD in the system memory until its expiration to prevent a second usage of it. An attempt to replay an already performed operation using the same SAD is blocked.

T.SIGNATURE_REQUEST_DISCLOSURE

T.SIGNATURE_REQUEST_DISCLOSURE is covered by the OT.SAP requiring the protocol to be able to transmit data securely.

Application Note BR (ST): this threat is covered by the OE.DEVICE requiring the SIC to participate in the SAP. It is not required that R.DTBS/R or SAD are encrypted. It is assumed that the device is protected against malicious code.

T.SAD_FORGERY

T.SAD_FORGERY is covered by OT.SAP requiring the TOE to be able to detect if the SAD has been modified during transit to the TOE.

It is also covered by OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION requiring signature authentication data to be protected during transit to the TOE.

It is also covered by OE.SIGNER_AUTHENTICATION_DATA requiring the Signer to protect his authentication data.

It is also covered by OE.DEVICE requiring the device used by the Signer to participate correctly in the SAP, in particular the device shall not disclose authentication data.

Application Note BS (ST): the TOE detects any attempt to modify the SAD by verifying the signature of R.Reference_Signer_Authentication_Data against the white list of allowed SVDs into R.TSF_DATA. It also checks the validity of the elements bound to the SAD.

T.DTBSR_FORGERY

T.DTBSR_FORGERY is covered by OT.DTBSR_INTEGRITY requiring the R.DTBS/R to be protected in integrity during transit to the TOE.

It is also covered by OE.DEVICE requiring the SIC to participate in the SAP.

Application Note BT (ST): the TOE accepts a R.DTBS/R that is protected in integrity and bound to other data inside the SAD. The SAP contains a hash (or a HMAC) of the SAD. The TOE verifies the SAD. It is assumed that the device is protected against malicious code.

T.SIGNATURE_FORGERY

T.SIGNATURE_FORGERY is covered by OT.SIGNATURE_INTEGRITY requiring that the signature is protected in integrity inside the TOE.

It is also covered by OT.CRYPTO requiring the usage of endorsed algorithms.

Application Note BU (ST): a signature can be verified by using R.SVD. The TOE does not perform signature verification after signing operation. The signature can also be verified by audit logs, since an audit log is generated with signing operation.

T.PRIVILEGED_USER_INSERTION

T.PRIVILEGED_USER_INSERTION is covered by OT.PRIVILEGED_USER_MANAGEMENT requiring only Privileged User can create new R.Privileged_User and OT.PRIVILEGED_USER_AUTHENTICATION that requires a Privileged User to be authenticated.

Application Note BV (ST): only an authorized Privileged User with a specific role (APPLICATION-OPERATOR) can create a new Privileged User. Privileged User is authenticated by the TOE indirectly with the verification of his/her signed assertion by his/her public key stored in R.TSF_DATA. During installation phase, a System Administrator creates the first Privileged User on the TOE with permissions, thus enabling the system to operate.

T.REFERENCE_PRIVILEGED_USER_AUTHENTICATION_DATA_MODIFICATION

T.REFERENCE_PRIVILEGED_USER_AUTHENTICATION_DATA_MODIFICATION is covered by OT.PRIVILEGED_USER_MANAGEMENT requiring only Privileged User can modify R.Privileged_User and OT.PRIVILEGED_USER_AUTHENTICATION that requires a Privileged User to be authenticated.

It is also covered by OT.PRIVILEGED_USER_PROTECTION requiring the Privileged User to be protected in integrity.

Application Note BW (ST): apart from the one-time initialisation phase, during the normal operational management a Privileged User is authenticated to the TOE before any operation, indirectly with the verification of his/her signed assertion. The allowed public keys for identification are securely stored into the TOE.

T.AUTHORISATION_DATA_UPDATE

T.AUTHORISATION_DATA_UPDATE is covered by OT.SYSTEM_PROTECTION requiring any unauthorised modification to TOE configuration to be detectable.

Application Note BX (ST): Privileged Users cannot update R.Authorisation_Data of Signers. He/she is authorized to modify only R.TSF_DATA. The R.TSF_DATA is protected in integrity and confidentiality and does not contain R.Authorisation_Data.

Data such as R.Signing_Key_Id and R.Authorisation_Data are protected in integrity and in confidentiality.

Each modification generates an audit log.

T.AUTHORISATION_DATA_DISCLOSE

T.AUTHORISATION_DATA_DISCLOSE is covered by OT.SYSTEM_PROTECTION requiring any unauthorised modification to TOE configuration to be detectable.

Application Note BY (ST): R.Authorisation_Data is protected in integrity by the external authentication system SVD and confidentiality through the TLS connection.

The keys used to guarantee protection are not the same of the R.TSF_DATA.

Each modification generates an audit log.

T.CONTEXT_ALTERATION

T.CONTEXT_ALTERATION is covered by OT.SYSTEM_PROTECTION requiring any unauthorised modification to TOE configuration to be detectable.

Application Note BZ (ST): only Privileged Users with specific role (APPLICATION-OPERATOR) are authorized to modify R.TSF_DATA. The R.TSF_DATA is protected in integrity and confidentiality.

T.AUDIT_ALTERATION

T.AUDIT_ALTERATION is covered by OT.AUDIT_PROTECTION requiring any audit modification can be detected.

Application Note CA (ST): the R.AUDIT is protected in integrity using an asymmetric key. A system auditor is able to detect modification of R.AUDIT with the public key.

T.RANDOM

T.RANDOM is covered by OT.RANDOM requiring that random numbers are not predictable and have sufficient entropy.

Application Note CB (ST): the TOE relies on the Cryptographic Module for the random number generation.

Application Note CC (ST): InfoCert or any TSP is compliant with the requirements for QTSP, namely with the requirements identified in ETSI EN 319 401 General Policy Requirements for Trust Service Providers.

A Privileged User is authorised to operate with the TOE, and each activity generates an audit record that is stored by passing through SSA.

The audit record contains: identity entity (e.g. R.Signer, R.Privileged_User), event date and time, type of event, status of event (success, failure).

The sole control of the Signer is guaranteed according to the adopted authentication schema used, that is the indirect schema whereas the delegated party verifies authentication, and the SAD is generated after the verification.

To meet the quality metric to ensure that random numbers are not predictable and have enough entropy, the TOE uses the library provided by the Cryptographic Module vendor.

The TOE is implemented within the same physical boundary of the Cryptographic Module, thus relying on the latter for cryptographic functionalities and random number generation.

6.3.2 Organizational security policies and objectives

OSP.RANDOM

OSP.RANDOM is covered by OT.RANDOM requiring that random numbers are not predictable and have sufficient entropy.

OSP.CRYPTO

OSP.CRYPTO is covered by OT.CRYPTO requiring the usage of endorsed algorithms and OE.CRYPTOMODULE_CERTIFIED requiring a cryptographic module to provide a tamper-protected environment and for cryptographic functionality and random number generation.

6.3.3 Assumptions and objectives

A.PRIVILEGED_USER

A.PRIVILEGED_USER is covered by OE.TW4S_CONFORMANT which requires that the system where the TOE operates is compliant with [4] where clause SRG_M.1.8 requires that administrators are trained.

Personnel administering the TOE is trusted namely without judicial records, is trained to conduct the activities. Human Resources Department keeps assessment records for each person involved. The access authorization list is managed by Security Officer.

A.SIGNER_ENROLMENT

A.SIGNER_ENROLMENT is covered by OE.ENV requiring the TSP to be audited, as a qualified TSP according to eIDAS.

The request for key generation is executed by the TOE and the key generation is performed by cryptographic module. TOE and cryptographic module operate in a protected environment where the control access is strictly regulated by internal procedure for QTSP certified ISO 27001 and compliant with eIDAS regulation and regularly audited by CAB.

A.SIGNER_AUTHENTICATION_DATA_PROTECTION

A.SIGNER_AUTHENTICATION_DATA_PROTECTION is covered by OE.SIGNER_AUTHENTICATION_DATA requiring the Signer to protect his authentication data.

The Signer is conscious regarding the correct use of authentication means because he/she signs a contract having a specific clause (Signer's Obligation) regarding authentication data protection.

A.SIGNER_DEVICE

A.SIGNER_DEVICE is covered by OE.DEVICE requiring the Signer's device to be protected against malicious code.

The device, computer/tablet/smart phone containing the SIC and which is used by the Signer to interact with the TOE is protected against malicious code. It participates using SIC as local part of the SAP. It can also be used to view the document to be signed.

A.CA

A.CA is covered by OE.CA_REQUEST_CERTIFICATE requiring that the CA will issue certificates containing the SVD.

The operational environment issues a certificate including SVD, Signer information and CA signature. The request process is able demonstrate the Signer is in control of the signing key providing audit record of certificate request. The integrity of the request is protected by a mutual authentication using TLS or a signature.

A.ACCESS_PROTECTED

A.ACCESS_PROTECTED is covered by OE.ENV requiring the TOE be operated in an environment with physical access controls.

The TOE software and cryptographic module operate in a protected environment where the

control access is strictly regulated by internal procedure for QTSP certified ISO 27001 and compliant with eIDAS regulation and regularly audited by CAB.

The access authorization list is managed by Security Officer.

A.AUTH_DATA

A.AUTH_DATA is covered by OE.DEVICE requiring the device to participate correctly in the SAP.

A.TSP_AUDITED

A.TSP_AUDITED is covered by OE.ENV requiring that the TOE is operated by a qualified TSP.

A.SEC_REQ

A.SEC_REQ is covered by OE.TW4S_CONFORMANT requiring the system where the TOE operates is compliant with [EN 419 241-1].

Application Note CD (ST): each of these Assumptions is directly matched by a security objective for the operational environment in section 6.2. The wording of each objective for the operational environment includes the wording of each assumption, and no further rationale is therefore given here.

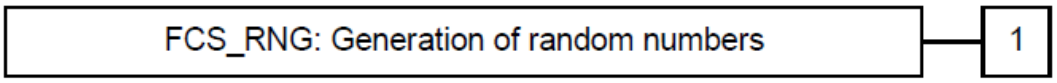
7 EXTENDED COMPONENT DEFINITION (ASE_ECD)

In analogy with the content presented in Section 7 of PP, this ST extends the Class FCS: Cryptographic Support with a new family: Generation of Random Numbers (FCS_RNG). This family describes the functional requirements for random number generation used for cryptographic purposes.

Family behaviour:

This family defines quality requirements for the generation of random numbers, which are intended to be use for cryptographic purposes.

Component levelling:



Management: FCS_RNG.1

There are no foreseen management activities.

Audit: FCS_RNG.1

There are no actions defined to be auditable.

FCS_RNG.1 Generation of random numbers

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FCS_RNG.1.1	The TSF shall provide a [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*] random number generator that implements: [assignment: *list of security capabilities*].
FCS_RNG.1.2	The TSF shall provide [selection: *bits, octets of bits, numbers [assignment: format of the numbers]*] that meet [assignment: *a defined quality metric*].

Application Note 27 (from PP)

A physical random number generator (RNG) produces the random number by a noise source based on physical random processes. A non-physical true RNG uses a noise source based on non-physical random processes like human interaction (key strokes, mouse movement). A

deterministic RNG uses a random seed to produce a pseudorandom output. A hybrid RNG combines the principles of physical and deterministic RNGs where a hybrid physical RNG produces at least the amount of entropy the RNG output may contain and the internal state of a hybrid deterministic RNG output contains fresh entropy but less than the output of RNG may contain.

8 SECURITY REQUIREMENTS (ASE_REQ)

This section defines the security requirements satisfied by the TOE. Each requirement has been extracted from version 3.1 of the Common Criteria, part 2 providing functional requirements and part 3 providing assurance requirements.

Part 2 of the Common Criteria defines an approved set of operations that may be applied to security functional requirements.

The following are the approved operations and the document conventions that are used within this ST to depict their application:

Refinement: The refinement operation allows the addition of extra detail to a requirement or deletions. Refinements are indicated using **bolded text for additions**, and strike through text (~~example~~) for deletions.

Assignment: The assignment operation provides the ability to specify an identified parameter within a requirement. Assignments are depicted using bolded text and are surrounded by square brackets as follows **[assignment]**. Note that Assignments that are made in the Protection Profile and copied as is in this ST are written in italics, as follows *assignment*.

Selection: The selection operation allows the specification of one or more items from a list. Selections are depicted using bold italics text and are surrounded by square brackets as follows **[*selection*]**. Note that Selections that are made in the Protection Profile and copied as is in this ST are written in italics, as follows *selection.*

Iteration: The iteration operation allows a component to be used more than once with varying operations. Iterations are depicted by placing a double slash "//" at the end of the component identifier and a unique name for the iteration. Note that Iterations that are made in the Protection Profile and copied as is in this ST are marked by a single slash "/" instead.

The below tables describe the subjects, object and operations supported by the TOE, as described in the Protection Profile.

Topic	Value	Description
Subject	R.Signer	Represents within the TOE, the end user that wants to create a digital signature
	R.Privileged_User	Represents within the TOE, a privileged user that can administer the TOE and a few operations relevant for R.Signer
Object	R.Reference_Privileged_User_Authentication_Data	Data used by the TOE to authenticate a Privileged_User
	R.Reference_Signer_Authe	Data used by the TOE to authenticate a Signer

Topic	Value	Description
	ntication_Data	
	R.SVD	The public part of a R.Signer signature key pair
	R.Signing_Key_Id	An identifier representing the private part of a R.Signer signature key pair
	R.DTBS/R	Data to be signed representation
	R.Authorisation_Data	Data used by the Cryptographic Module to activate the private part of a R.Signer signature key pair
	R.Signature	The result of a signature operation
	R.TSF_DATA	TOE Configuration Data

TABLE 11 - OPERATIONS SUPPORTED BY THE TOE

Subject/Operation	Object	Description from PP	ST remarks
R.Privileged_User			
Create_New_ Privileged_User	R.Privileged_User R.Reference_Privileged_User_Authentication_Data	A new privileged user can be created which covers the object representing the new privileged user as well as the object used to authenticate the newly created privileged user.	
Create_New_Signer	R.Signer R.Reference_Signer_Authentication_Data	A new signer can be created which covers the object representing the new signer as well as the object used to authenticate the newly created signer.	
Supply_DTBS/ R	R.Signer R.DTBS/R	Data to be signed by a signer can be supplied by a privileged user.	Privileged User cannot supply a DTBS/R on behalf of the Signer
TOE_Maintenance	R.TSF_DATA	The TOE configuration can be maintained by a privileged user.	
R.Signer			
Signing	R.Authorisation_Data R.Signer R.Signing_Key_Id R.DTBS/R R.Signature	A signer can sign data to be signed resulting in a signature.	
R.Privileged_User, R.Signer			
Generate_Signer_	R.Signer	A key pair can be generated and assigned to a signer.	A key pair can be generated and assigned

Subject/Operation	Object	Description from PP	ST remarks
Key_Pair	R.SVD R.Signing_Key_Id		to a Signer by a Privileged User.
Signer_Maintenance	R.Signer R.SVD R.Signing_Key_Id	A key pair can be deleted from a signer.	

The following list gives an overview of how the SFRs are related to handling TOE usage scenarios and Signer object, as described in the Protection Profile.

Signer object

- FIA_ATD.1 and FIA_USB.1 requires that the R.Signer object is maintained by the TOE.
- FDP_ITC.2/Signer describes requirements for importing the R.Signer object.
- FDP_ETC.2/Signer describes requirements for exporting the R.Signer object.
- FDP_UIT.1 requires the R.Signer object to be protected in integrity when imported and exported.
- FPT_TDC.1 requires the TOE to be able to interpret R.Signer object related data when shared with SSA.
- FMT_MSA.1, FMT_MSA.2 and FMT_MSA.3 describe rules for creation, maintaining and usage of the R.Signer object as well as requirements to its values.

Authentication

- FIA_AFL.1 limits the amount of authentication attempts
- FDP_UCT.1 ensure that access control and information flow data are transmitted in a confidential way.
- FIA_UID.2 and FIA_UAU.1 require that each user is identified and authenticated before any action on behalf of the user can take place.
- FIA_UAU.5/Signer and FIA_UAU.5/Privileged User describe the list of authentication mechanism

Create Signer

- FDP_ACC.1/Signer Creation using FDP_ACF.1/Signer Creation describes access control requirements for creating a R.Signer object.
- FIA_USB.1 defines authorisation rules for creating new R.Signer objects.

Signer Key Pair Generation

- FDP_ACC.1/Signer Key Pair Generation using FDP_ACF.1/Signer Key Pair Generation describes access control requirements for signing key pair generation.
- FCS_CKM.1 describes rules for how signing key pair are generated

Signer Key Pair Deletion

- FDP_ACC.1/Signer Key Pair Deletion using FDP_ACF.1/Signer Key Pair Deletion describes access control requirements for signing key pair deletion.
- FCS_CKM.4 requires keys to be securely destructed.

Signer Maintenance

- FDP_ACC.1/Signer Maintenance using FDP_ACF.1/Signer Maintenance describes access control requirements for updating the R.Reference_Signer_Authenticaton_Data of a R.Signer object.

Supply DTBS/R

- FDP_ACC.1/Supply DTBS/R using FDP_ACF.1/Supply DTBS/R describes access control requirements for a Privileged User to supply a DTBS/R(s).

Signing

- FDP_IFF.1/Signer and FDP_IFC.1/Signer describing requirements on preconditions for a signature operation can be carried out.
- FDP_UIT.1 requires the R.SAD object to be protected from modification and replay.
- FDP_ACC.1/Signing using FDP_ACF.1/Signing describes access control requirements for signing.
- FCS_COP.1 requires the TOE to perform cryptographic operation conformant with a ST specified list of algorithms.

Privileged User object

- FIA_ATD.1 and FIA_USB.1 require that the R.Privileged_User object is maintained by the TOE.
- FDP_ITC.2/Privileged User describes requirements for importing the R.Privileged_User object.
- FDP_ETC.2/Privileged User describes requirements for exporting the R.Privileged_User object
- FDP_UIT.1 requires the R.Privileged_User object to be protected in integrity when imported and exported.
- FPT_TDC.1 requires the TOE to be able to interpret R.Privileged_User object when shared with a trusted IT product the SSA.
- FMT_MSA.1, FMT_MSA.2, FMT_MSA.3 describes rules for creation, maintaining and usage of the R.Privileged_User object as well as requirements to its values.

Privileged User Creation

- FDP_ACC.1/Privileged User Creation using FDP_ACF.1/ Privileged User Creation describes access control requirements for creating a R.Privileged_User object.
- FIA_USB.1 defines authorisation rules for creating new R.Privileged_User objects.

TOE Maintenace

- FDP_ACC.1/TOE Maintenance using FDP_ACF.1/TOE Maintenance
- FMT_SMF.1 and FMT_SMF.2 require the TOE to be able to carry out management functions and maintain users and roles.

Audit

- FAU_GEN.1 and FAU_GEN.2 describe what shall be audited.

Communication

- FPT_ITC.2 requires that all communication to the TOE comes from the SSA.
- FTP_TRP.1/SSA and FTP_TRP.1/SIC require that either the Privileged User or the Signer initiates the communication.

8.1 Security Functional Requirements (SFR)

The individual security functional requirements are specified in the sections below.

8.1.1 Security Audit (FAU)

Functional Requirements
Security Audit (FAU)
FAU_GEN.1 Audit Generation
FAU_GEN.2 User identity association

FAU_GEN.1 Audit Generation

Hierarchical to:	No other components.
Dependencies:	FPT_STM.1 Reliable time stamps
FAU_GEN.1.1	The TSF shall be able to generate an audit record of the following auditable events: a) Start-up and shutdown of the audit functions; b) All auditable events for the **[*not specified*]** level of audit; and *c) Privileged User management;* *d) Privileged User authentication;* *e) Signer management;* *f) Signer authentication;* *g) Signing key generation;* *h) Signing key destruction;* *i) Signing key activation and usage including the hash of the DTBS/R(s); and R.Signature;* *j) ~~Change of TOE configuration;~~*

	k) **[None].**
Application Note AA (ST):	This SFR has been refined at letter j), as the TOE doesn't have any configuration object to be modified. The behaviour of the TOE is set by design. Management of R.Signer objects includes all events which creates and modifies the R.Signer objects. Signer authentication includes failed verification of an assertion provided by a delegated party. Generation of a certification request is usage of the signing key and mandates an audit trail. The R.DTBS/R is recorded in the audit log. As the R.DTBS/R is a hash representation, there are no privacy concerns for it being recorded. It is used to demonstrate that a particular DTBS/R(s) has been signed.
FAU_GEN.1.2	The TSF shall record within each audit record at least the following information: a) Date and time of the event, type of event, subject R.Privileged_User or R.Signer, and the outcome (success or failure) of the event; and b) For each audit event type, based on the auditable event definitions of the functional components included in the ST, **[Type of action performed (success or failure), [identity of the role which performs the operation], [R.DTBS/R for signature operation], [certificate serial number and certificate issuerDn for signature operation]]**.
Application Note AB (ST)	Audit trail does not include any data which allow retrieving sensitive data like R.SAD, R.Reference_Signer_Authentication_Data and R.Authorisation_Data.

FAU_GEN.2 User identity association

Hierarchical to:	No other components.
Dependencies:	FAU_GEN.1 Audit data generation FIA_UID.1 Timing of identification
FAU_GEN.2.1	For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.
Application Note AC (ST):	The identity of the user is mapped to R.Privileged_User or R.Signer according to the event.

8.1.2 Cryptographic Support (FCS)

Functional Requirements
Cryptographic Support (FCS)
FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FCS_COP.1 Cryptographic operation
FCS_RNG.1 Generation of random numbers

FCS_CKM.1 Cryptographic key generation

Hierarchical to:	No other components.
Dependencies:	FCS_COP.1 Cryptographic operation FCS_CKM.4 Cryptographic key destruction
FCS_CKM.1.1	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **[as shown in Table 12 - Key Generation Table]** and specified cryptographic key sizes **[as shown in the Table 12 - Key Generation Table]** that meet the following: **[standards as shown in the Table 12 - Key Generation Table]**.
Application Note 32 (from PP)	The TOE is expected to use a cryptographic module certified in conformance with [EN 419 221-5], see also OE.CRYPTOMODULE_CERTIFIED for key generation. Although the TSF may not generate keys itself, this SFR expresses the requirement for the TSF to invoke the cryptographic module with the appropriate parameters whenever key generation is required. Guidance on cryptographic algorithms can be found in [ETSI TS 119 312] and [SOGIS].
Application Note AD (ST)	The ST is expected to use cryptographic keys for different purposes, e.g. application, infrastructure, session etc. Key types are stated in Table 12 - Key Generation Table.

TABLE 12 - KEY GENERATION TABLE

Key Generation Algorithm	Key Sizes	Applicable Standards
RSA – generation of probable primes	2048-bit to 4096-bit	IETF RFC 8017
ECDSA – generation of key Pairs BrainpoolP256r1	256-bit	ANSSI curves | IETF RFC 5639 | FIPS Publication 186-4 | ISO/IEC 14888-3

ECDSA – generation of key Pairs BrainpoolP320r1	320-bit	ANSSI curves | IETF RFC 5639 | FIPS Publication 186-4 | ISO/IEC 14888-3
ECDSA – generation of key Pairs BrainpoolP384r1	384-bit	ANSSI curves | IETF RFC 5639 | FIPS Publication 186-4 | ISO/IEC 14888-3
ECDSA – generation of key Pairs BrainpoolP512r1	512-bit	ANSSI curves | IETF RFC 5639 | FIPS Publication 186-4 | ISO/IEC 14888-3
ECDSA – generation of key Pairs Secp256r1	256-bit	ANSSI curves | IETF RFC 5639 | FIPS Publication 186-4 | ISO/IEC 14888-3
ECDSA – generation of key Pairs Secp384r1	384-bit	ANSSI curves | IETF RFC 5639 | FIPS Publication 186-4 | ISO/IEC 14888-3
ECDSA – generation of key Pairs Secp521r1	521-bit	ANSSI curves | IETF RFC 5639 | FIPS Publication 186-4 | ISO/IEC 14888-3
AES – generation of symmetric key	256-bit	FIPS Publication 197

FCS_CKM.4 Cryptographic key destruction

Hierarchical to:	No other components.
Dependencies:	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 **[zeroization]** that meets the following: **[standard FIPS 140-2 Level 3]**.
Application Note 34 (from PP)	The TOE is expected to use a cryptographic module certified in conformance with [EN 419 221-5] for key destruction. Although the TSF may not destruct keys, this SFR expresses the requirement for the TSF to invoke the cryptographic module with the appropriate parameters whenever key destruction is required. The Security Target must specify the method(s) of secure destruction of all secret keys and all support keys and must ensure that all are covered by a secure destruction method. If necessary, then more than one iteration of FCS_CKM.4 may be included to describe different standards for secure deletion. The 'list of standards' in the final assignment may be met in the Security Target by simply providing a description of the action taken to zeroise the keys rather than referencing an external standard.
Application Note 35	The ST writer should include an iteration of this SFR for purposes of keys that

(from PP)	it destructs itself.
Application Note AE (ST)	The TOE relies on the cryptographic module implementation for keys destruction.

FCS_COP.1 Cryptographic operation

Hierarchical to:	No other components.
Dependencies:	FCS_CKM.1 Cryptographic key generation FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1	The TSF shall perform **[digital signature-generation, symmetric encryption, symmetric decryption]** in accordance with a specified cryptographic algorithm **[as shown in Table 13 – Cryptographic Algorithms Table]** and cryptographic key sizes **[as shown in the Table 13 – Cryptographic Algorithms Table]** that meet the following: **[standards as shown in the Table 13 – Cryptographic Algorithms Table]**.
Application Note 36 (from PP)	The TOE is expected to use a cryptographic module certified in conformance with [EN 419 221-5] for cryptographic operations.
Application Note AF (ST)	The relevant authorities and endorsements for completion of the SFRs are determined by the context of the client applications that use the TOE. For digital signatures within the European Union, this is as indicated in Regulation (EU) No 910/2014 [eIDAS] and a list of approved signature and seal formats are given in [18]. Digital signature-generation is applied to R.DTBS/R of R.Signer. Symmetric encryption and decryption are applied to Signer data for protecting R.Signer, R.Signing_Key_Id and R.SVD in integrity. R.Signing_Key_Id is also protected in confidentiality.

TABLE 13 – CRYPTOGRAPHIC ALGORITHMS TABLE

Cryptographic Operation	Algorithm	Key Sizes	Padding	Hash Algorithm	Applicable Standards
Digital signature generation	RSA PKCS#1 v1.5	2048-bit to 4096-bit	RSASSAPKCS1-v1.5	SHA256 SHA384 SHA512 SHA3-256 SHA3-384 SHA3-512	IETF RFC 3447
Digital signature generation	RSA PKCS#1 PSS	2048-bit to 4096-bit	RSASSA-PSS	SHA256 SHA384 SHA512	IETF RFC 3447
Digital signature generation	ECDSA BrainpoolP256r1	256-bit	Not Applicable	SHA-256 SHA3-256	SOGIS v1.2
Digital signature	ECDSA	320-bit	Not Applicable	SHA-384 SHA3-384	SOGIS v1.2

generation	BrainpoolP320r1				
Digital signature generation	ECDSA BrainpoolP384r1	384-bit	Not Applicable	SHA-384 SHA3-384	SOGIS v1.2
Digital signature generation	ECDSA BrainpoolP512r1	512-bit	Not Applicable	SHA-512 SHA3-512	SOGIS v1.2
Digital signature generation	ECDSA Secp256r1	256-bit	Not Applicable	SHA-256 SHA3-256	SOGIS v1.2
Digital signature generation	ECDSA Secp384r1	384-bit	Not Applicable	SHA-384 SHA3-384	SOGIS v1.2
Digital signature generation	ECDSA Secp521r1	521-bit	Not Applicable	SHA-512 SHA3-512	SOGIS v1.2
Symmetric encryption/decryption	AES CBC	256-bit	PKCS#5	Not Applicable	FIPS Publication 197, SP 800-38A

FCS_RNG.1 Generation of random numbers

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FCS_RNG.1.1	The TSF shall provide a **[*physical*]** random number generator that implements: **[the list of security capabilities implemented by the [EN 419-221-5]-certified Cryptographic Module, and in particular according to requirement FCS_RNG.1.1 in the CM Security Target]**.
FCS_RNG.1.2	The TSF shall provide **[*octets of bits*]** that meet **[the defined quality metric implemented by the [EN 419-221-5]-certified Cryptographic Module, and in particular according of requirement FCS_RNG.1.2 in the CM Security Target]**.
Application Note 38 (from PP)	For more information on the selections and assignments, see the SFR definition in section 7.1.1 of PP.
Application Note AG (ST)	The SFRs defined in [EN 419-221-5] already provide requirements on generation of random numbers. The TOE relies on the cryptographic module [5] for the generation of random numbers. The TOE is a composite product consisting of an application installed in the same physical boundary of the cryptographic module and of the cryptographic module itself. The TOE makes use of the cryptographic module specified in Table 4 – TOE Components.

8.1.3 User Data Protection (FDP)

Functional Requirements
User Data Protection (FDP)
FDP_ACC.1/Privileged User Creation Subset access control
FDP_ACF.1/Privileged User Creation Security attribute-based access control
FDP_ACC.1/Signer Creation Subset access control
FDP_ACF.1/ Signer Creation Security attribute-based access control
FDP_ACC.1/Signer Maintenance Subset access control
FDP_ACF.1/Signer Maintenance Security attribute-based access control
FDP_ACC.1/Signer Key Pair Generation Subset access control
FDP_ACF.1/Signer Key Pair Generation Security attribute-based access control
FDP_ACC.1/Signer Key Pair Deletion Subset access control
FDP_ACF.1/Signer Key Pair Deletion Security attribute-based access control
FDP_ACC.1/Supply DTBS/R Subset access control
FDP_ACF.1/Supply DTBS/R Security attribute-based access control
FDP_ACC.1/Signing Subset access control
FDP_ACF.1/Signing Security attribute-based access control
FDP_ACC.1/TOE Maintenance Subset access control
FDP_ACF.1/TOE Maintenance Security attribute-based access control
FDP_ETC.2/Signer Export of user data with security attributes
FDP_IFC.1/Signer Subset information flow control
FDP_IFF.1/Signer Simple security attributes
FDP_ETC.2/ Privileged User Export of user data with security attributes
FDP_IFC.1/Privileged User Subset information flow control
FDP_IFF.1/Privileged User Simple security attributes
FDP_ITC.2/Signer Import of user data with security attributes
FDP_ITC.2/ Privileged User Import of user data with security attributes
FDP_UCT.1 Basic data exchange confidentiality

Functional Requirements
FDP_UIT.1 Data exchange integrity

FDP_ACC.1/Privileged User Creation Subset access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute-based access control
FDP_ACC.1.1/ Privileged User Creation	The TSF shall enforce the *Privileged User Creation SFP* on: *Subjects: Privileged User* *Objects: New security attributes for the Privileged User to be created.* *Operations: Create_New_Privileged_User:* *The TOE creates R.Privileged_User and R.Reference_Privileged_User_Authentication_Data with information transmitted by Privileged User*.
Application Note AH (ST)	As part of the TOE, there is a static data set file which includes, among other data, the list of Privileged Users recognized by the system. This file is created and filled with information by an authorized operator during the TOE installation phase. The operators who install and maintain the TOE are Privileged Users with role APPLICATION-OPERATOR and they administrate the TOE by authenticating themselves with a smartcard. Only a Privileged User with this specific role can access and modify the data file. Within the TOE installation, an authorized operator sets a list of Privileged Users with their associated roles and public keys in the file. During this stage, the TOE is not ready to operate, it must be started by the same or another authorized operator with the start-up command. With this command, the TOE is ready to work and from that time onwards, it rejects any request who is not made by a Privileged User set in the data file. A Privileged User is always created by another one, if the latter has the necessary role to do so. No quorum of Privileged Users is required to create a new Privileged User.

FDP_ACF.1/Privileged User Creation Security attribute-based access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1 / Privileged User	The TSF shall enforce the *Privileged User Creation SFP* to objects based on the following: *(1) whether the subject is a Privileged User authorized to create a new Privileged User*.

Creation	
FDP_ACF.1.2 / Privileged User Creation	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *(1) Only a Privileged User who has been authorised for creation of new users can carry out the Create_New_Privileged_User operation*.
FDP_ACF.1.3 / Privileged User Creation	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *None*.
FDP_ACF.1.4 / Privileged User Creation	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *None*.

The following security functional requirements in the FDP_ACC.1/ series Signer Creation, Signer Key Pair Generation, Signer Maintenance, Supply DTBS/R and Signing are intended as building blocks to describe Signer management and the signature operation within the TOE.

FDP_ACC.1/Signer Creation Subset access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1	The TSF shall enforce the *Signer Creation SFP* on *Subjects: Privileged User* *Objects: R.Signer and R.Reference_Signer_Authentication_Data* *Operations: Create_New_Signer].* *The TOE creates R.Signer and R.Reference_Signer_Authentication_Data with information transmitted by Privileged User*

FDP_ACF.1/ Signer Creation Security attribute based access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1 / Signer Creation	The TSF shall enforce the *Signer Creation SFP* to objects based on the following: *(1) whether the subject is a Privileged User authorized to create a new Signer.*
FDP_ACF.1.2 /	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *(1) Only a Privileged*

Signer Creation	*User who has been authorised for creation of new users can carry out the Create_New_Signer operation.*
FDP_ACF.1.3 / Signer Creation	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *None*.
FDP_ACF.1.4 / Signer Creation	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *None*.

FDP_ACC.1/Signer Maintenance Subset access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute-based access control
FDP_ACC.1.1/ Signer Maintenance	The TSF shall enforce the *Signer Maintenance SFP* on *Subjects: Privileged User and Signer* *Objects: The security attributes R.Reference_Signer_Authentication_Data of R.Signer* *Operations: Signer_Maintenance:* *The Privileged User or Signer instructs the TOE to update R.Reference_Signer_Authentication_Data of R.Signer.*
Application Note AI (ST):	R.Reference_Signer_Authentication_Data includes the EAS SVDs or certificates that the TOE uses to verify an assertion provided as a result of delegated authentication. The Signer cannot instruct the TOE about update of R.Reference_Signer_Authentication_Data, only a Privileged User with role APPLICATION-OPERATOR can.

FDP_ACF.1/Signer Maintenance Security attribute based access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1/ Signer Maintenance	The TSF shall enforce the *Signer Maintenance SFP* to objects based on the following: *(1) Whether the subject is a Privileged User or Signer authorised to maintain the Signer security attributes.*
FDP_ACF.1.2/ Signer Maintenance	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *(1) Only a Privileged User or Signer who has been authorised to maintain a Signer can carry out the Signer_Maintenance operation.*

FDP_ACF.1.3/ Signer Maintenance	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *(1) The Signer must be the owner of the R.Signer object to be maintained.*
FDP_ACF.1.4/ Signer Maintenance	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *(1) If the Signer does not own the R.Signer object, it can't be maintained.*
Application Note AJ (ST):	Established that this ST adopts indirect authentication schema, R.Reference_Signer_Authentication_Data can only be maintained by Privileged User and it cannot be maintained by Signer. Signer can maintain his own R.Authorisation_Data object.

FDP_ACC.1/Signer Key Pair Generation Subset access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1/ Signer Key Pair Generation	The TSF shall enforce the *Signer Key Pair Generation SFP* on *Subjects: Privileged User and Signer.* *Objects: The security attributes R.SVD and R.Signing_Key_Id as part of R.Signer.* *Operations: Generate_Signer_Key_Pair:* *The Privileged User or Signer instructs the TOE to request the Cryptographic Module to generate a signing key pair R.Signing_Key_Id and R.SVD and assign them to the R.Signer.*
Application Note AK (ST):	R.Authorisation_Data is passed to the TOE by the Privileged User in key pair generation request, as Privileged User generates the key pair on behalf of the Signer. Signing keys can be used by several cryptographic modules and the keys are protected by encryption outside the module. How keys are protected is described in [nShield Solo XC HSM Security Target version 1.0]. Both R.Signer and R.Authorisation_Data are protected in integrity. The cryptographic module always checks the validity of R.Authorisation_Data. The TOE does not make use of pre-generated keys. Signing keys are not generated by the Cryptographic Module in advance. The environment ensures, if needed, any transformation of R.SVD to a certification request and transport to CA.

FDP_ACF.1/Signer Key Pair Generation Security attribute based access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1 / Signer Key Pair Generation	The TSF shall enforce the *Signer Key Pair Generation SFP* to objects based on the following: *(1) whether the subject is a Privileged User or Signer authorised to generate a key pair.*
FDP_ACF.1.2 / Signer Key Pair Generation	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *(1) Only a Privileged User or Signer who has been authorised to generate the key pair can carry out the Generate_Signer_Key_Pair operation.*
FDP_ACF.1.3/ Signer Key Pair Generation	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *(1) The Signer must be the owner of the R.Signer object where the key pair is to be generated.*
FDP_ACF.1.4/ Signer Key Pair Generation	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *(1) If the Signer does not own the R.Signer object, key pair shall not be generated.*
Application Note AL (ST):	No pre-generated keys are used by the TOE.
Application Note 47 (from PP)	Owning a R.Signer object is described in FIA_UAU.5/Signer.

FDP_ACC.1/Signer Key Pair Deletion Subset access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1/ Signer Key Pair Deletion	The TSF shall enforce the *Signer Key Pair Deletion SFP* on *Subjects: Privileged User and Signer* *Objects: The security attributes R.Signing_Key_Id and R.SVD of R.Signer.* *Operations: Signer_Key_Pair_Deletion:* *The Privileged User or Signer instructs the TOE to delete the R.Signing_Key_Id and R.SVD from R.Signer.*
Application Note 48	Deletion of R.Signing_Key_Id may also require that the signing key is deleted

(from PP)	by the Cryptographic Module. This SFR is limited to covering deletion of the R.Signing_Key_Id and R.SVD of R.Signer performed using one of the interfaces provided by the TOE and where authorisation to perform operations is managed by TOE.
Application Note AM (ST):	The key pair is always deleted from the cryptographic module after usage.

FDP_ACF.1/Signer Key Pair Deletion Security attribute based access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1 / Signer Key Pair Deletion	The TSF shall enforce the *Signer Key Pair Deletion SFP* to objects based on the following: *(1) whether the subject is a Privileged User or Signer authorised to delete the Signer security attributes.*
FDP_ACF.1.2 / Signer Key Pair Deletion	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *(1) Only a Privileged User or Signer who has been authorised to delete a key pair can carry out the Signer_Key_Pair_Deletion operation].*
FDP_ACF.1.3/ Signer Key Pair Deletion	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *(1) The Signer must be the owner of the R.Signer object containing the key pair to be deleted.*
FDP_ACF.1.4/ Signer Key Pair Deletion	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *(1) If the Signer does not own the R.Signer object, key pair can't be deleted.*
Application Note AN (ST):	Owning a R.Signer object is described in FIA_UAU.5/Signer.

FDP_ACC.1/Supply DTBS/R Subset access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1/ Supply DTBS/R	The TSF shall enforce the *Supply DTBS/R SFP* on *Subjects: Privileged User* *Objects: The security attributes R.DTBS/R of R.Signer.*

	Operations: Supply_DTBS/R: *The Privileged User instructs the TOE to link the supplied DTBS/R(s) to the next signature operation for R.Signer.*
Application Note AO (ST):	The Privileged User cannot supply R.DTBS/R on behalf of the Signer.

FDP_ACF.1/Supply DTBS/R Security attribute based access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1	The TSF shall enforce the *Supply DTBS/R SFP* to objects based on the following: *(1) Whether the subject is a Privileged User authorised to supply a DTBS/R(s).*
FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *(1) Only a Privileged User who has been authorised to supply a DTBS/R(s) can carry out the Supply_DTBS/R operation.*
FDP_ACF.1.3	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *None.*
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *None.*
Application Note AP (ST):	The Privileged User cannot supply R.DTBS/R on behalf of the Signer.

FDP_ACC.1/Signing Subset access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.1/ Signing	The TSF shall enforce the *Signing SFP* on *Subjects: Signer* *Objects: R.Authorisation_Data, security attributes R.Signing_Key_Id and R.DTBS/R of R.Signer and R.Signature.* *Operations: Signing:* *The Signer instructs the TOE to perform a signature operation containing the following steps:* • *The TOE establishes R.Authorisation_Data for the R.Signing_Key_Id.* • *The TOE uses the R.Authorisation_Data, and R.Signing_Key_Id to*

	activate a signing key in the Cryptographic Module and signs the R.DTBS/R resulting in R.Signature. • *The TOE deactivates the signing key when the signature operation is completed.*
Application Note AQ (ST):	R.Authorisation_Data is an input data passed to the TOE to activate signing keys. The R.DTBS/R is contained into the R.SAD that is sent to the TOE. A R.SAD can contain a single R.DTBS/R or an array of different R.DTBS/R(s). FDP_ACC.1/supply DTBS/R does not supply R.DTBS/R. Signing key deactivating means that the signer shall authorise any subsequent use of it. After the signing session is completed, the related signing key cannot be used without the Signer authorization.

FDP_ACF.1/Signing Security attribute based access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1/ Signing	The TSF shall enforce the *Signing SFP* to objects based on the following: *(1) Whether the subject is a Signer authorised to create a signature*.
FDP_ACF.1.2/ Signing	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *(1) The R.SAD is verified in integrity.* *(2) The R.SAD is verified that it binds together the Signer authentication, a set of R.DTBS/R and R.Signing_Key_Id.* *(3) The R.DTBS/R used for signature operations is bound to the R.SAD.* *(4) The Signer identified in the SAD is authenticated according to the rules specified in FIA_UAU.5/Signer.* *(5) Only a R.Signing_Key_Id as bound in the SAD, and which is part of the R.Signer security attributes, can be used to create a signature*.
FDP_ACF.1.3/ Signing	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *(1) The Signer must be the owner of the R.Signer object used to generate the signature.*
FDP_ACF.1.4/	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *(1) If the Signer does not own the R.Signer object,*

Signing	*it can't be used to create a signature.*
Application Note 54 (from PP)	In FDP_ACF.1.2/Signing the R.Signing_Key_Id can be implied if the signing uses a one-time keys or a signing key is known to be the default.
Application Note AR (ST):	The R.Signer is unique inside the TOE domain. This rule is applied to all signature types.

FDP_ACC.1/TOE Maintenance Subset access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute-based access control
FDP_ACC.1.1/ TOE Maintenance	The TSF shall enforce the *TOE Maintenance SFP* on *Subjects: Privileged User* *Objects: R.TSF_DATA.* *Operations: TOE_Maintenance:* *The Privileged User transmits information to the TOE to manage R.TSF_DATA.*

FDP_ACF.1/TOE Maintenance Security attribute based access control

Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1 / TOE Maintenance	The TSF shall enforce the *TOE Maintenance SFP* to objects based on the following: *(1) Whether the subject is a Privileged User authorised to maintain the TOE configuration data*.
FDP_ACF.1.2 / TOE Maintenance	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *(1) Only a Privileged User who has been authorised to maintain the TOE can carry out the TOE_Maintenance operation.*
FDP_ACF.1.3 / TOE Maintenance	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *None.*
FDP_ACF.1.4 / TOE Maintenance	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *None.*

FDP_ETC.2/Signer Export of user data with security attributes

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control
FDP_ETC.2.1/ Signer	The TSF shall enforce the *Signer Creation SFP, Signer Key Pair Generation SFP, Signer Key Pair Deletion SFP, Signer Maintenance SFP, Supply DTBS/R SFP and Signing SFP* when exporting user data, controlled under the SFP(s), outside of the TSF.
FDP_ETC.2.2/ Signer	The TSF shall export the user data with the user data's associated security attributes.
FDP_ETC.2.3/ Signer	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.
FDP_ETC.2.4/ Signer	The TSF shall enforce the following rules when user data is exported from the TOE: *None.*
Application Note AS (ST):	Signer exports his/her own keys and certificate data containing the security attribute R.SVD bound to R.Signer. R.Signature is also exported after signing.

FDP_IFC.1/Signer Subset information flow control

Hierarchical to:	No other components.
Dependencies:	FDP_IFF.1 Simple security attributes
FDP_IFC.1.1 / Signer	The TSF shall enforce the *Signer Flow SFP* on *Privileged User and Signer accessing Signer security attributes for all operations.*

FDP_IFF.1/Signer Simple security attributes

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation
FDP_IFF.1.1/ Signer	The TSF shall enforce the *Signer Flow SFP* based on the following types of subject and information security attributes: *Privileged User and Signer accessing the Signer security attributes.*
FDP_IFF.1.2/ Signer	The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: *The TOE shall be initialized with FDP_ACC.1/TOE Maintenance.* *To allow a Signer to sign, the Signer shall be created in the TOE by FDP_ACC.1/Signer Creation followed by FDP_ACC.1/Signer key Pair*

	Generation. *After Signer is created the following operations can be done: FDP_ACC.1/Signer Key Pair Generation, FDP_ACC.1/Signer Key Pair Deletion, FDP_ACC.1/Supply DTBS/R, FDP_ACC.1/Signer Maintenance and FDP_ACC.1/Signing.*
FDP_IFF.1.3/ Signer	The TSF shall enforce the *None.*
FDP_IFF.1.4/ Signer	The TSF shall explicitly authorise an information flow based on the following rules: *None.*
FDP_IFF.1.5/ Signer	The TSF shall explicitly deny an information flow based on the following rules: *None.*

FDP_ETC.2/ Privileged User Export of user data with security attributes

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control
FDP_ETC.2.1 / Privileged User	The TSF shall enforce the *Privileged User Creation SFP* when exporting user data, controlled under the SFP(s), outside of the TOE.
FDP_ETC.2.2 / Privileged User	The TSF shall export the user data with the user data's associated security attributes
FDP_ETC.2.3 / Privileged User	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.
FDP_ETC.2.4 / Privileged User	The TSF shall enforce the following rules when user data is exported from the TOE: *None*
Application Note AT (ST):	Privileged User can export R.Reference_Privileged_User_Authentication_Data, provided he/she has the role to do so. Privileged User can also export R.Signer, R.Signing_Key_Id and Signer certificate, the latter containing the security attribute R.SVD.

FDP_IFC.1/Privileged User Subset information flow control

Hierarchical to:	No other components.
Dependencies:	FDP_IFF.1 Simple security attributes
FDP_IFC.1.1 /	The TSF shall enforce the *Privileged User Flow SFP* on *Privileged User accessing*

Privileged User	*Privileged User security attributes for all operations.*

FDP_IFF.1/Privileged User Simple security attributes

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation
FDP_IFF.1.1/ Privileged User	The TSF shall enforce the *Privileged User Flow SFP*] based on the following types of subject and information security attributes: *Privileged User accessing the Privileged User security attributes.*
FDP_IFF.1.2/ Privileged User	The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: *The TOE shall be initialized with FDP_ACC.1/TOE Maintenance.*
FDP_IFF.1.3/ Privileged User	The TSF shall enforce the *None*
FDP_IFF.1.4/ Privileged User	The TSF shall explicitly authorise an information flow based on the following rules: *None*
FDP_IFF.1.5/ Privileged User	The TSF shall explicitly deny an information flow based on the following rules: *None*

FDP_ITC.2/Signer Import of user data with security attributes

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FTP_TRP.1 Trusted path FPT_TDC.1 Inter-TSF basic TSF data consistency
FDP_ITC.2.1/ Signer	The TSF shall enforce the *Signer Creation SFP, Signer Key Pair Generation SFP, Signer Key Pair Deletion, Signer Maintenance SFP, Supply DTBS/R SFP and Signing SFP* when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.2.2/ Signer	The TSF shall use the security attributes associated with the imported user data.
FDP_ITC.2.3/ Signer	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.
FDP_ITC.2.4/ Signer	The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.

FDP_ITC.2.5/ Signer	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: *None*.
Application Note AU (ST):	The Signer imports his/her own keys and certificate data containing the security attribute R.SVD bound to R.Signer. The TOE checks the integrity of these data, which have a signature attached, by verifying the signature with its integrity key.

FDP_ITC.2/ Privileged User Import of user data with security attributes

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FTP_TRP.1 Trusted path FPT_TDC.1 Inter-TSF basic TSF data consistency
FDP_ITC.2.1/ Privileged User	The TSF shall enforce the *Privileged User Creation SFP* when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.2.2/ Privileged User	The TSF shall use the security attributes associated with the imported user data.
FDP_ITC.2.3/ Privileged User	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received
FDP_ITC.2.4/ Privileged User	The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.
FDP_ITC.2.5/ Privileged User	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: *None*
Application Note AV (ST):	Privileged User imports R.Reference_Privileged_User_Authentication_Data when creating a new Privileged User or when updating the certificate of an existing one. Privileged User imports Signer certificate bound to R.Signer, provided he/she has the role to do so. The certificate contains the security attribute R.SVD. Privileged User can also import R.TSF_DATA.

FDP_UCT.1 Basic data exchange confidentiality

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FTP_TRP.1 Trusted path

FDP_UCT.1.1	The TSF shall enforce the *Signer Flow SFP and Privileged User Flow SFP* to *transmit and receive* user data in a manner protected from unauthorised disclosure.

FDP_UIT.1 Data exchange integrity

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FTP_TRP.1 Trusted path
FDP_UIT.1.1	The TSF shall enforce the *Signer Flow SFP and Privileged User Flow SFP* to *transmit and receive* user data in a manner protected from *modification and insertion* errors **for R.Signer and R.Privileged User and for R.SAD also** from *modification and replay* errors.
FDP_UIT.1.2	The TSF shall be able to determine on receipt of user data, whether *modification, deletion and insertion* **for R.Signer and R.Privileged_User and for R.SAD** whether *modification and replay* has occurred.
Application Note AW (ST):	Insertion of objects means that authorised creation of Signer and Privileged User is possible. The TOE verifies the integrity of data exchange before any action.

8.1.4 Identification and Authentication (FIA)

Functional Requirements
Identification and Authentication (FIA)
FIA_AFL.1 Authentication failure handling
FIA_ATD.1 User attribute definition
FIA_UAU.1 Timing of authentication
FIA_UAU.5/Signer Multiple authentication mechanisms
FIA_UAU.5/Privileged User Multiple authentication mechanisms
FIA_UID.2 User identification before any action
FIA_USB.1 User-subject binding

FIA_AFL.1 Authentication failure handling

Hierarchical to:	No other components.
Dependencies:	FIA_UID.1 Timing of identification
FIA_AFL.1.1	~~The TSF shall detect when [selection: *[assignment: positive integer number]*, **a TOE Maintenance** configurable positive integer within *[assignment: range of acceptable values]*] unsuccessful authentication attempts occur related to *Privileged User and Signer authentication*.~~
FIA_AFL.1.2	~~When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall *suspend the Privileged User and when it is a Signer suspend the usage of R.Signing_Key_Id*.~~
Application Note AX (ST):	The SFR only applies when the TOE uses any direct authentication. Therefore, the SFR is not applicable to Signer and Privileged User authentication because the TOE uses an indirect authentication schema for both.

FIA_ATD.1 User attribute definition

Hierarchical to:	No other components.
Dependencies:	No dependencies
FIA_ATD.1.1	The TSF shall maintain the following list of security attributes belonging to individual users: *the security attribute as defined in FIA_USB.1.*

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 **[health checks diagnostic, starting-up command, shutting-down command]** 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.

FIA_UAU.5/Signer Multiple authentication mechanisms

Hierarchical to:	No other components.
Dependencies:	No dependencies
FIA_UAU.5.1/Signer	The TSF shall provide ***[[a delegated authentication mechanism conformant to [4] SRA_SAP.1.1 based on a signed authorization token]]*** to support **Signer**

	authentication.
FIA_UAU.5.2/Signer	The TSF shall authenticate any **Signer's** claimed identity according to the ***[[result of the signed authorization token verification against the list of the public keys contained into R.TSF_DATA]]***.
Application Note AY (ST):	This SFR only applies to signer authentication for maintaining signer (FDP_ACC.1/Signer Maintenance, FDP_ACC.1/Signer Key Pair Generation and FDP_ACC.1/Signer Key Pair Deletion) and for signing (FDP_ACC.1/Signing). For Signer authentication, the TOE uses a delegated authentication mechanism conformant to [4]. Signer performs a 2-Factor authentication on the external authentication service, as part of SSA, by providing his/her identity and an OTP. The protocol used is compliant to [14]. If the 2-Factor verification is successful, the delegated party releases a signed authentication assertion, which is the R.SAD. The TOE verifies the assertion signature with the external authentication service SVD, which is stored into R.TSF_DATA. Successful authentication gives Signer access to the relevant R.Signer object as the owner.

FIA_UAU.5/Privileged User Multiple authentication mechanisms

Hierarchical to:	No other components.
Dependencies:	No dependencies
FIA_UAU.5.1/Privileged User	The TSF shall provide **[the verification of an assertion signed by Privileged User with its R.SVD stored in R.TSF_DATA]** to support **Privileged User** authentication.
FIA_UAU.5.2/Privileged User	The TSF shall authenticate any user's claimed identity according to the **[successful verification of the user assertion's signature with the public key of the claimed Privileged User which is stored in R.TSF_DATA]**.

FIA_UID.2 User identification before any action

Hierarchical to:	FIA_UID.1 Timing of identification
Dependencies:	No dependencies
FIA_UID.2.1	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

FIA_USB.1 User-subject binding

Hierarchical to:	No other components.
Dependencies:	FIA_ATD.1 User attribute definition
FIA_USB.1.1	The TSF shall associate the following user security attributes with subjects

	acting on the behalf of that user: *(1) R.Reference_Signer_Authentication_Data* *(2) R.Signing_Key_Id* *(3) R.SVD* *(4) R.Signer* *(5)* **[R.Authorisation_Data** **(6) R.DTBS/R]** *to Signer* *(1) R.Reference_Privileged_User_Authentication_Data* *(2)* **[R.Privileged_User]** *to Privileged User*.
FIA_USB.1.2	The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: *(1) Whether the subject is a Privileged User authorized to create a new Signer.* *(2) Whether the subject is a Privileged User authorized to create a new Privileged User.* *(3)* **[Whether the subject is a Privileged User assigning valid SVD for the verification of R.Reference_Signer_Authentication_Data** ***(4)*** **Whether the subject is a Privileged User assigning valid R.Reference_Privileged_User_Authentication_Data]**
FIA_USB.1.3	The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: *(1) Whether the subject is a Privileged User authorized to modify a R.Signer object.* *(2) Whether the subject is a Signer authorized to modify his own R.Signer* **and R.Authorisation_Data** *object.* *(3)* **[Whether the subject is a Privileged User providing valid modified R.Authorisation_Data to Signer** ***(4)*** **Whether the subject is a Signer providing valid modified R.Authorisation_Data to himself/herself** ***(5)*** **Whether the subject is a Privileged User providing valid modified R.Reference_Privileged_User_Authentication_Data to another Privileged User]**.
Application Note 63 (from PP)	In FIA_USB.1.2 several attributes including R.Signing_Key_ID, R.SVD and R.DTBS/R may initially be empty.
Application Note AZ (ST):	R.Authrorisation_Data is included as a security attribute of the Signer. R.DTBS/R is a Signer attribute, as only the Signer submits it to the TOE. Privileged User cannot submit it on behalf of the Signer.

8.1.5 Security Management (FMT)

Functional Requirements
Security Management (FMT)
FMT_MSA.1/Signer Management of security attributes
FMT_MSA.1/Privileged User Management of security attributes
FMT_MSA.2 Secure security attributes
FMT_MSA.3/Signer Static attribute initialisation
FMT_MSA.3/Privileged User Static attribute initialisation
FMT_MTD.1 Management of TSF data
FMT_SMF.1 Specification of Management Functions
FMT_SMR.2 Restrictions on security roles

FMT_MSA.1/Signer Management of security attribute

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MSA.1.1/ Signer	The TSF shall enforce the *(1) Signer Creation SFP* to restrict the ability to *create* the security attributes *listed in FIA_USB.1 for Signer* to *authorised Privileged User.* (2) *Generate Signer Key Pair SFP* to restrict the ability to *generate* the security attributes *R.SVD and R.Signing_Key_Id* to *authorised Privileged User and Signer.* *(3) Signer Key Pair Deletion SFP* to restrict the ability to *destruct* the security attribute *R.SVD and R.Signing_Key_Id as part of R.Signer* to *authorised Signer* *(4) Supply DTBS/R SFP* to restrict the ability to *create* the security attribute *R.DTBS/R as part of R.Signer* to authorised *Privileged User*. *(5) Signing SFP* to restrict the ability to *create* the security attribute *R.Signature as part of R.Signer* to authorised *Signer*. *(6) Signing SFP* to restrict the ability to *query* the security attributes *as listed in FIA_USB.1* to authorised *Signer*. *(7) Signer Maintenance SFP* to restrict the ability to *change* the security attributes *R.Reference_Signer_Authentication_Data as part of R.Signer* to authorised *Privileged User and Signer*.

FMT_MSA.1/Privileged User Management of security attributes

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MSA.1.1/ Privileged User	The TSF shall enforce the (1) *Privileged User Creation SFP* to restrict the ability to *create and query* the security attributes *listed in FIA_USB.1 for Privileged User* to *authorised Privileged User.*

FMT_MSA.2 Secure security attributes

Hierarchical to:	No other components.
Dependencies:	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 *all security attributes listed in FIA_USB.1*.

FMT_MSA.3/Signer Static attribute initialisation

Hierarchical to:	No other components.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.3.1/ Signer	The TSF shall enforce the *Signer Creation SFP* to provide *restrictive* default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/ Signer	The TSF shall allow the *Privileged User* to specify alternative initial values to override the default values when an object or information is created.

Application Note BA (ST):	• CREATED (signer has been created) default • WITH_KEYS (signer has a key pair) • WITH_REQUEST (signer produced the certification request) • DISABLED (signer has a certificate but hasn't signed yet) • ENABLED (signer has a certificate and has already signed) • LOCKED (signer attempted to sign with a wrong secret too many times and has been definitely blocked)

FMT_MSA.3/Privileged User Static attribute initialisation

Hierarchical to:	No other components.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.3.1/ Privileged User	The TSF shall enforce the *Privileged User Creation SFP* to provide *restrictive* default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/ Privileged User	The TSF shall allow the *Privileged User* to specify alternative initial values to override the default values when an object or information is created.
Application Note BB (ST):	• The default initial value for Privileged User security attributes is role AS, but it can be overridden by single value A or single value S. Role AS (Privileged User has been created) default

FMT_MTD.1 Management of TSF data

Hierarchical to:	No other components.
Dependencies:	FMT_SMF.1 Specification of Management Functions FMT_SMR.1 Security roles
FMT_MTD.1.1	The TSF shall restrict the ability to *modify* the *R.TSF_DATA* to *Privileged User.*
Application Note 66 (from PP):	The TSF data includes configuration of administrator roles.

FMT_SMF.1 Specification of Management Functions

Hierarchical to:	No other components.
Dependencies:	No dependencies
FMT_SMF.1.1	The TSF shall be capable of performing the following management functions: *(1) Signer management,* *(2) Privileged User management and* *(3) Configuration management* *(4)* **[None]**

FMT_SMR.2 Restrictions on security roles

Hierarchical to:	FMT_SMR.1 Security roles
Dependencies:	FIA_UID.1 Timing of identification
FMT_SMR.2.1	The TSF shall maintain the roles: *Signer and Privileged User* **(as detailed in Table 14 - Roles vs Operations), [None]**.
FMT_SMR.2.2	The TSF shall be able to associate users with roles.
FMT_SMR.2.3	The TSF shall ensure that the conditions *Signer cannot be a Privileged User* are satisfied.
Application Note BC (ST):	Table 14 - Roles vs Operations describes which Privileged User roles are defined in the TOE and which operations the role can perform.

TABLE 14 - ROLES VS OPERATIONS

Privileged_User	Roles	Operations
System Administrator	**APPLICATION-OPERATOR**	Start-up and shut-down the TOE; Create other privileged users; Assign one or more roles to them; Retrieve, update or delete any data related to them; Retrieve, add or delete a R.SVD associated to a specific external authentication service; Retrieve, add or delete any data in R.TSF_DATA.
System Operator	**SIGNERS-MANAGER**	Create Signers; Retrieve, update or delete any data related to them according to the defined services.

8.1.6 Protection of the TSF (FPT)

Functional Requirements
Protection of the TSF (FPT)
FPT_PHP.1 Passive
FPT_PHP.3 Resistance
FPT_RPL.1 Replay detection
FPT_STM.1 Reliable time stamps
FPT_TDC.1 Inter-TSF basic TSF data consistency

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.
Application Note 68 (from PP)	Passive detection of a physical attack is typically achieved by using physical seals and an appropriate physical design of the TOE that allows the TOE administrator to verify the physical integrity of the TOE as part of a routine inspection procedure. Because of the requirement for a physically secure environment with regular inspections (cf. OE.ENV), the level of protection (and hence resistance to attack potential) that is required by the implementation of FPT_PHP.1 for this TOE is equivalent to the physical security mechanisms for tamper detection and response required by section 7.7.2 Physical security general requirements and section 7.7.3 Physical security requirements for each physical security embodiment in [ISO/IEC 19790] for Security Level 3.
Application Note BD (ST):	The TOE includes the Cryptographic Module certified against requirements in Protection Profile [5], thus being within the same secure perimeter that meets requirements of FIPS 140-2 Level 3.

FPT_PHP.3 Resistance to physical attack

Hierarchical to:	No other components.
Dependencies:	No dependencies

FPT_PHP.3.1	The TSF shall resist **[physical penetration attempts]** to **[the hard opaque potted enclosure]** by responding automatically such that the SFRs are always enforced.
Application Note BE (ST):	Since the TOE consists of a local application within the same physical boundary as the cryptographic module defined in [EN 419-221-5], the SFRs FTP_PHP.* relies on the similar SFRs described in the ST for the cryptographic module. Details are given in the nShield Solo XC HSM Security Target [20].
Application Note 70 (from PP)	This SFR is linked to the requirements for passive detection of physical attacks in FPT_PHP.1, and should identify the relevant responses of the TOE involved in meeting the key zeroisation requirements of [ISO/IEC 19790] Security Level 3. As in the case of FPT_PHP.1, because of the requirement for a physically secure environment with regular inspections (cf. OE.ENV), the level of protection (and hence resistance to attack potential) that is required by the implementation of FPT_PHP.3 for this TOE is equivalent to the level of assessment for this aspect of tamper detection and response required for section 7.7.2 Physical security general requirements and section 7.7.3 Physical security requirements by each physical security embodiment in [ISO/IEC 19790] for Security Level 3.

FPT_RPL.1 Replay detection

Hierarchical to:	No other components.
Dependencies:	No dependencies
FPT_RPL.1.1	The TSF shall detect replay for the following entities: *R.SAD.*
FPT_RPL.1.2	The TSF shall perform *reject the signature operation* when replay is detected.

FPT_STM.1 Reliable time stamps

Hierarchical to:	No other components.
Dependencies:	No dependencies
FPT_STM.1.1	The TSF shall be able to provide reliable time stamps.
Application Note BF (ST):	The TOE retrieves a reliable time source from the CM real-time internal clock.

FPT_TDC.1 Inter-TSF basic TSF data consistency

Hierarchical to:	No other components.
Dependencies:	No dependencies
FPT_TDC.1.1	The TSF shall provide the capability to consistently interpret

	1. *R.Signer,* 2. *R.Reference_Signer_Authentication_Data,* 3. *R.SAD,* 4. *R.DTBS/R* 5. *R.SVD* 6. *R.Privileged_User* 7. *R.Reference_Privileged_User_Authentication_Data* 8. *R.TSF_DATA* when shared between the TSF and another trusted IT product.
FPT_TDC.1.2	The TSF shall use *data integrity either on data or on communication channel* when interpreting the TSF data from another trusted IT product.
Application Note 72 (from PP):	The SFR is used to handle the situation where the whole or part of the above data are stored outside the TOE.

8.1.7 Trusted Paths/Channels (FTP)

Functional Requirements
Trusted Paths/Channels (FTP)
FTP_TRP.1/SSA Inter-TSF Trusted path
FTP_TRP.1/SIC Inter-TSF Trusted path
FTP_ITC.1/CM Inter-TSF trusted channel

FTP_TRP.1/SSA Inter-TSF Trusted path

Hierarchical to:	No other components.
Dependencies:	No dependencies
FTP_TRP.1.1/SSA	The TSF shall provide a communication path between itself and ***Privileged User through SSA*** users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from *modification*.
FTP_TRP.1.2/SSA	The TSF shall permit ***Privileged User through SSA*** to initiate communication via the trusted path.
FTP_TRP.1.3/SSA	The TSF shall require the use of the trusted path for *(1) FDP_ACC.1.1/Privileged User Creation* *(2) FDP_ACC.1/Signer Creation*

	(3) FDP_ACC.1/Signer Maintenance *(4) FDP_ACC.1/Signer Key Pair Generation* *(5) FDP_ACC.1/Signer Key Pair Deletion* *(6) FDP_ACC.1/Supply DTBS/R* *(7) FDP_ACC.1/TOE Maintenance* **(8) [None]**
Application Note 73 (from PP):	Since it is not all data transmitted to the TOE that needs to be protected in confidentiality, FTP_TRP.1/SSA only requires protection from modification.

FTP_TRP.1/SIC Trusted path

Hierarchical to:	No other components.
Dependencies:	No dependencies
FTP_TRP.1.1/SIC	The TSF shall provide a communication path between itself and ***Remote Signer through SIC*** users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from *modification*.
FTP_TRP.1.2/SIC	The TSF shall permit ***Remote Signer through SIC*** to initiate communication via the trusted path.
FTP_TRP.1.3/SIC	The TSF shall require the use of the trusted path for *(1) FDP_ACC.1/Signer Maintenance* *(2) FDP_ACC.1/Signer Key Pair Generation* *(3) FDP_ACC.1/Signer Key Pair Deletion* *(4) FDP_ACC.1/Signing.* **(5) [None]**
Application Note BG (ST)	Since it is not all data transmitted to the TOE that needs to be protected in confidentiality, FTP_TRP.1.1/SIC only requires protection from modification. The transmission of sensitive data, such as R.Authorisation_Data and R.Reference_Signer_Authentication_Data, is protected in confidentiality by the TLS channel. The TOE is not expected to verify the SIC as a communication end point and it may rely on the signer authentication. The TOE does not verify SIC as a communication endpoint because the TOE communicates only with SSA and it relies on the Signer authentication.

FTP_ITC.1/CM Inter-TSF trusted channel

Hierarchical to:	No other components.
Dependencies:	No dependencies
FTP_ITC.1.1/CM	The TSF shall provide a communication path between itself and **a cryptographic module certified according to EN 419 221-5** 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/CM	The TSF shall permit **the TSF and a cryptographic module certified according to EN 419 221-5** to initiate communication via the trusted channel.
FTP_ITC.1.3/CM	The TSF shall initiate communication via the trusted channel for **[** **(1) Cryptographic key generation** **(2) Cryptographic key destruction** **(3) Digital signature generation** **(4) Random number generation** **(5) R.Authorisation_Data change]**.
Application Note 75 (from PP)	FTP_ITC.1/CM must be completed in a Security Target to reflect the way that the TOE communicates with the cryptographic module, and to justify its security. Where the TOE and the cryptographic module are located within the same hardware appliance (e.g. the TOE being a local application running on a server and communicating with a PCI card on the server's internal PCI bus) then the trusted channel may be mapped in the Security Target to the physical configuration, and no additional authentication or cryptographic protection are required (because of the physical security assumed in the appliance environment).
Application Note BH (ST)	The TOE is operated within the same physical boundary of the cryptographic module, relying on the local nature of the communication to ensure integrity and confidentiality protection. No additional authentication or cryptographic protection are required.

8.1.8 SFR Dependency Analysis

The following table (taken from the PP without additions, deletions or modification) gives an overview of the dependencies and shows how they are fulfilled.

TABLE 15 - DEPENDENCIES OF THE FUNCTIONAL REQUIREMENTS

Functional Requirements	*CC Required Dependencies*	*Fulfilled by*
FAU_GEN.1 Audit Generation	FPT_STM.1	FPT_STM.1
FAU_GEN.2 User identity association	FAU_GEN.1 and FIA_UID.1	FAU_GEN.1 and FIA_UID.1
FCS_CKM.1 Cryptographic key generation	[FCS_CKM.2 or FCS_COP.1] and FCS_CKM.4	FCS_COP.1 and FCS_CKM.4
FCS_CKM.4 Cryptographic key destruction	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1
FCS_COP.1 Cryptographic operation	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1], FCS_CKM.4	FCS_CKM.1 FCS_CKM.4
FCS_RNG.1	None	No dependencies
FDP_ACC.1/Privileged User Creation	FDP_ACF.1	FDP_ACF.1/Privileged User Creation
FDP_ACC.1/Signer Creation	FDP_ACF.1	FDP_ACF.1/ Signer Creation
FDP_ACC.1/Signer Maintenance	FDP_ACF.1	FDP_ACF.1/Signer Maintenance
FDP_ACC.1/Signer Key Pair Generation	FDP_ACF.1	FDP_ACF.1/Signer Key Pair Generation
FDP_ACC.1/Signer Key Pair Deletion	FDP_ACF.1	FDP_ACF.1/Signer Key Pair Deletion
FDP_ACC.1/Supply DTBS/R	FDP_ACF.1	FDP_ACF.1/Supply DTBS/R
FDP_ACC.1/Signing	FDP_ACF.1	FDP_ACF.1/Signing
FDP_ACC.1/TOE Maintenance	FDP_ACF.1	FDP_ACF.1/TOE Maintenance
FDP_ACF.1/Privileged User Creation	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Privileged User Creation FMT_MSA.3/Privileged User
FDP_ACF.1/ Signer Creation	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Signer Creation FMT_MSA.3/Signer
FDP_ACF.1/Signer Maintenance	FDP_ACC.1	FDP_ACC.1/Signer Maintenance

Functional Requirements	CC Required Dependencies	Fulfilled by
	FMT_MSA.3	FMT_MSA.3/Signer
FDP_ACF.1/Signer Key Pair Generation	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Signer Key Pair Generation FMT_MSA.3/Signer
FDP_ACF.1/Signer Key Pair Deletion	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Signer Key Pair Deletion FMT_MSA.3/Signer
FDP_ACF.1/Supply DTBS/R	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Supply DTBS/R FMT_MSA.3/Signer
FDP_ACF.1/Signing	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Signing FMT_MSA.3/Signer
FDP_ACF.1/TOE Maintenance	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/TOE Maintenance FMT_MSA.3/Privileged User
FDP_ETC.2/Signer	[FDP_ACC.1 or FDP_IFC.1]	FDP_IFC.1/Signer
FDP_ETC.2/Privileged User	[FDP_ACC.1 or FDP_IFC.1]	FDP_IFC.1/ Privileged User
FDP_IFC.1/Signer	FDP_IFF.1	FDP_IFF.1/Signer
FDP_IFF.1/Signer	FDP_IFC.1 FMT_MSA.3	FDP_IFC.1/Signer FMT_MSA.3/Signer
FDP_IFC.1/Privileged User	FDP_IFF.1	FDP_IFF.1/ Privileged User
FDP_IFF.1/Privileged User	FDP_IFC.1 FMT_MSA.3	FDP_IFC.1/Privileged User FMT_MSA.3/Privileged User
FDP_ITC.2/Signer	[FDP_ACC.1 or FDP_IFC.1 FTP_ITC.1 or FTP_TRP.1], FTP_TDC.1	FDP_IFC.1/Signer FTP_TRP.1/SSA and FTp_TRP.1/SIC FPT_TDC.1
FDP_ITC.2/ Privileged User	[FDP_ACC.1 or FDP_IFC.1 FTP_ITC.1 or FTP_TRP.1], FTP_TDC.1	FDP_IFC.1/ Privileged User FTP_TRP.1/SSA FPT_TDC.1
FDP_UCT.1	[FTP_ITC.1 or FTP_TRP.1], [FDP_ACC.1 or FDP_IFC.1]	FTP_TRP.1/SSA and FTP_TRP.1/SIC FDP_IFC.1/Signer FDP_IFC.1/Privileged User
FDP_UIT.1	[FDP_ACC.1 or FDP_IFC.1],	FDP_IFC.1/Signer

Functional Requirements	CC Required Dependencies	Fulfilled by
	[FTP_ITC.1 or FTP_TRP.1]	FDP_IFC.1/Privileged User FTP_TRP.1/SSA and FTP_TRP.1/SIC
FIA_AFL.1	FIA_UAU.1	FIA_UAU.1
FIA_ATD.1	None	No dependencies
FIA_UAU.1	FIA_UID.1	FIA_UID.2
FIA_UAU.5/Signer	None	No dependencies
FIA_UAU.5/Privileged User	None	No dependencies
FIA_UID.2	None	No dependencies
FIA_USB.1	FIA_ATD.1	FIA_ATD.1
FMT_MSA.1/Signer	[FDP_ACC.1 or FDP_IFC.1], FMT_SMR.1 , FMT_SMF.1	FDP_IFC.1/Signer FMT_SMR.2 FMT_SMF.1
FMT_MSA.1/Privileged User	[FDP_ACC.1 or FDP_IFC.1], FMT_SMR.1, FMT_SMF.1	FDP_IFC.1/Privileged User FMT_SMR.2 FMT_SMF.1
FMT_MSA.2	[FDP_ACC.1 or FDP_IFC.1], FMT_MSA.1, FMT_SMR.1	FDP_IFC.1/Signer FDP_IFC.1/Privileged User FMT_MSA.1/Signer FMT_MSA.1/Privileged User FMT_SMR.2
FMT_MSA.3/Signer	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1/Signer FMT_SMR.2
FMT_MSA.3/Privileged User	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1/Privileged user FMT_SMR.2
FMT_MTD.1	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 FMT_SMF.1
FMT_SMF.1	None	No dependencies
FMT_SMR.2	FIA_UID.1	FIA_UID.2
FPT_PHP.1	None	No dependencies
FPT_PHP.3	None	No dependencies

Functional Requirements	CC Required Dependencies	Fulfilled by
FPT_RPL.1	None	No dependencies
FPT_STM.1	None	No dependencies
FPT_TDC.1	None	No dependencies
FTP_TRP.1/SSA	None	No dependencies
FTP_TRP.1/SIC	None	No dependencies
FTP_ITC.1/CM	None	No dependencies

8.2 Security Assurance Requirements (SAR)

Security Assurance Requirement level is EAL4 augmented with AVA_VAN.5. The assurance components are identified in the table below with the augmented item in bold.

TABLE 16 - SECURITY ASSURANCE REQUIREMENTS

Assurance Class	Assurance components
ADV: Development	ADV_ARC.1 Security architecture description
	ADV_FSP.4 Complete functional specification
	ADV_IMP.1 Implementation representation of the TSF
	ADV_TDS.3 Basic modular design
AGD: 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.4 Problem tracking CM coverage
	ALC_DEL.1 Delivery procedures
	ALC_DVS.1 Identification of security measures
	ALC_LCD.1 Developer defined life-cycle model
	ALC_TAT.1 Well-defined development tools
ASE: Security Target Evaluation	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Stated security requirements
	ASE_SPD.1 Security Problem Definition
	ASE_TSS.1 TOE summary specification
ATE: Tests	ATE_COV.2 Analysis of coverage
	ATE_DPT.1 Testing: basic design
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing - sample
AVA: Vulnerability assessment	**AVA_VAN.5 Advanced methodical vulnerability analysis**

8.2.1 Security Assurance Requirements Rationale

As argued in section 9.2.1 of Protection Profile [6], EAL4 permits a developer to gain maximum assurance from positive security engineering based on good commercial development practices which do not require substantial specialist knowledge, skills, and other resources. EAL4 is the highest level at which it is likely to be economically feasible to retrofit to an existing product line.

As the TOE manages signature creation data generation and authorises its use, it manages security attributes which can only be ensured by the TOE. While the TOE is assumed to be in a physically

protected environment, it is still subject to logical remote attacks and should be evaluated to deal with High attack potential.

EAL4 is therefore augmented with AVA_VAN.5.

8.3 Security requirements rationale

8.3.1 Security Requirements Coverage

This section is largely taken from section 9.1.1 of Protection Profile [6] with some refinements.

As presented in section 9.1.1 of Protection Profile [6], the following table provides a mapping of SFR to the security objectives, showing that each security functional requirement addresses at least one security objective.

TABLE 17 - SECURITY REQUIREMENTS COVERAGE

	OT.SIGNER_PROTECTION	OT.REFERENCE_SIGNER_AUTHENTICATION_DATA	OT.SIGNER_KEY_PAIR_GENERATION	OT.SVD	OT.PRIVILEGED_USER_MANAGEMENT	OT.PRIVILEGED_USER_AUTHENTICATION	OT.PRIVILEGED_USER_PROTECTION	OT.SIGNER_MANAGEMENT	OT.SYSTEM_PROTECTION	OT.AUDIT_PROTECTION	OT.SAD_VERIFICATION	OT.SAP	OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION	OT.DTBSR_INTEGRITY	OT.SIGNATURE_INTEGRITY	OT.CRYPTO	OT.RANDOM
		Security Audit															
FAU_GEN.1										X							
FAU_GEN.2										X							
		Cryptographic Support															
FCS_CKM.1			X													X	
FCS_CKM.4			X														
FCS_COP.1			X												X	X	
FCS.RNG.1			X														X
		User Data Protection															
FDP_ACC.1/Privileged User Creation					X												
FDP_ACF.1/Privileged User Creation					X												
FDP_ACC.1/Signer Creation		X						X									
FDP_ACF.1/Signer Creation		X						X									
FDP_ACC.1/Signer Maintenance		X															
FDP_ACF.1/Signer Maintenance		X															
FDP_ACC.1/Signer Key Pair Generation			X	X													

	OT.SIGNER_PROTECTION	OT.REFERENCE_SIGNER_AUTHENTICATION_DATA	OT.SIGNER_KEY_PAIR_GENERATION	OT.SVD	OT.PRIVILEGED_USER_MANAGEMENT	OT.PRIVILEGED_USER_AUTHENTICATION	OT.PRIVILEGED_USER_PROTECTION	OT.SIGNER_MANAGEMENT	OT.SYSTEM_PROTECTION	OT.AUDIT_PROTECTION	OT.SAD_VERIFICATION	OT.SAP	OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION	OT.DTBSR_INTEGRITY	OT.SIGNATURE_INTEGRITY	OT.CRYPTO	OT.RANDOM
FDP_ACF.1/Signer Key Pair Generation			X	X													
FDP_ACC.1/Signer Key Pair Deletion								X									
FDP_ACF.1/Signer Key Pair Deletion								X									
FDP_ACC.1/Supply DTBS/R														X			
FDP_ACF.1/Supply DTBS/R														X			
FDP_ACC.1/Signing											X				X		
FDP_ACF.1/Signing											X				X		
FDP_ACC.1/ TOE Maintenance									X								
FDP_ACF.1/TOE Maintenance									X								
FDP_ETC.2/Signer	X																
FDP_IFC.1/Signer	X																
FDP_IFF.1/Signer	X																
FDP_ETC.2/Privileged User					X		X										
FDP_IFC.1/Privileged User					X		X										
FDP_IFF.1/Privileged User					X		X										
FDP_ITC.2/Signer	X																
FDP_ITC.2/Privileged User					X		X										
FDP_UCT.1	X																
FDP_UIT.1	X																
		Identification and Authentication															
~~*FIA_AFL.1*~~	-	-	-	-	-	~~X~~		-	-	-	~~X~~	-	-	-	-		-
FIA_ATD.1	X				X		X										
FIA_UAU.1						X					X						
FIA_UAU.5/Signer											X						
FIA_UAU.5/Privileged User						X											
FIA_UID.2					X		X	X									

	OT.SIGNER_PROTECTION	OT.REFERENCE_SIGNER_AUTHENTICATION_DATA	OT.SIGNER_KEY_PAIR_GENERATION	OT.SVD	OT.PRIVILEGED_USER_MANAGEMENT	OT.PRIVILEGED_USER_AUTHENTICATION	OT.PRIVILEGED_USER_PROTECTION	OT.SIGNER_MANAGEMENT	OT.SYSTEM_PROTECTION	OT.AUDIT_PROTECTION	OT.SAD_VERIFICATION	OT.SAP	OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION	OT.DTBSR_INTEGRITY	OT.SIGNATURE_INTEGRITY	OT.CRYPTO	OT.RANDOM
FIA_USB.1	X		X		X		X										
		Security Management															
FMT_MSA.1/Signer								X									
FMT_MSA.1/Privileged User					X			X									
FMT_MSA.2					X			X									
FMT_MSA.3/Signer								X									
FMT_MSA.3/Privileged User					X			X									
FMT_MTD.1									X								
FMT_SMF.1									X								
FMT_SMR.2									X								
		Protection of the TSF															
FPT_PHP.1									X								
FPT_PHP.3									X								
FPT_RPL.1												X					
FPT_STM.1										X							
FPT_TDC.1	X				X												
		Trusted Path/Channels															
FTP_TRP.1/SSA									X					X			
FTP_TRP.1/SIC												X	X	X			
FTP_ITC.1/CM			X												X		

8.3.2 Security Requirements Sufficiency

This paragraph describes the rationale for SFRs and Security Objectives for the TOE.

OT.SIGNER_PROTECTION is handled by requirements export and import of R.Signer in a secure way. (FDP_ETC.2/Signer, FDP_IFC.1/Signer, FDP_IFF.1/Signer, FDP_ITC.2/Signer, FDP_UCT.1

FDP_UIT.1 and FPT_TDC.1). The actual description of the data are described in FIA_ATD.1 and FIA_USB.1.

OT.REFERENCE_SIGNER_AUTHENTICATION_DATA are handled by FDP_ACC.1/Signer Creation, FDP_ACF.1/Signer Creation, FDP_ACC.1/Signer Maintenance and FDP_ACF.1/Signer Maintenance which describes access control for creating and updating R.Signer and R.Reference_Signer_Authenticaton_Data.

OT.SIGNER_KEY_PAIR_GENERATION is handled by the requirements for key generation and cryptographic algorithms in FCS_CKM.1 and FCS_COP.1. FCS_RNG.1 provides a random source for key generation. FCS_CKM.4 describes the requirements for key destruction. FDP_ACC.1/Signer Key Pair Generation and FDP_ACF.1/Signer Key Pair Generation describes access control for creating a key pair. FIA_USB.1 describes that R.Signing_Key_Id is associated with Signer. FTP_ITC.1/CM can be used to communicate securely with a Cryptographic Module.

OT.SVD is handled by the requirements in FDP_ACC.1/Signer Key Pair Generation and FDP_ACF.1/Signer Key Pair Generation.

OT.PRIVILEGED_USER_MANAGEMENT is handled by requirements for export and import of R.Privileged User in a secure way (FDP_ETC.2/Privileged User, FDP_IFC.1/Privileged User, FDP_IFF.1/Privileged User, FDP_ITC.2/Privileged User and FPT_TDC.1). The actual description of the data are described in FIA_ATD.1 and FIA_USB.1. Authentication of Privileged User is handled by FIA_UID.2, FMT_MSA.1/Privileged User, FMT_MSA.2 and FMT_MSA.3/Privileged User. FDP_ACC.1/Privileged User Creation and FDP_ACF.1/Privileged User Creation describes access controls for creating Privileged Users.

OT.PRIVILEGED_USER_AUTHENTICATION is handled by FIA_AFL.1, FIA_UAU.1 and FIA_UAU.5/Privileged User.

OT.PRIVILEGED_USER_PROTECTION is handled by requirements for export and import of R.Privileged User in a secure way (FDP_ETC.2/Privileged User, FDP_IFC.1/Privileged User, FDP_IFF.1/Privileged User, FDP_ITC.2/Privileged User and FPT_TDC.1). The actual description of the data is described in FIA_ATD.1 and FIA_USB.1. FIA_UID.2 ensures that Privileged Users are authenticated they can carry out any operation.

OT.SIGNER_MANAGEMENT is handled by the requirements for access control in FDP_ACC.1/Signer Creation, FDP_ACF.1/Signer Creation, FDP_ACC.1/Signer Maintenance and FDP_ACF.1/Signer Maintenance. Authentication of Signers and Privileged Users are handled by FIA_UID.2, FMT_MSA.1/Signer, FMT_MSA.1/Privileged User, FMT_MSA.2, FMT_MSA.3/Signer and FMT _MSA.3 /Privileged User.

OT.SYSTEM_PROTECTION is handled by FMT_MTD.1, FMT_SMF.1 and FMT_SMR.2. FDP_ACC.1/TOE Maintenance and FDP_ACF.1/TOE Maintenance describes access control rules for managing TSF data. FPT_PHP.1 and FPT_PHP.3 describes requirements for TSF protection.

FTP_TRP.1/SSA describes that only a Privileged User can maintain the TOE.

OT.AUDIT_PROTECTION is handled by the requirements for audit record generation FAU_GEN.1 and FAU_GEN.2 using reliable time stamps in FPT_STM.1.

OT.SAD_VERIFICATION is handled by the FIA_AFL.1, FIA_UAU.1 and FIA_UAU.5/Signer. FDP_ACC.1/Signing and FDP_ACF.1/Signing describes access control rules for the signature operation and well as for SAP verification.

OT.SAP is covered by the requirements FTP_TRP.1/SIC and FPT_RPL.1 the protocol between the SIC and TSF.

OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION is covered by FTP_TRP.1/SIC, which describes the requirements for data transmitted to the TOE, is protected in integrity.

OT.DTBSR_INTEGRITY is covered by FTP_TRP.1/SSA and FTP_TRP.1/SIC requiring data transmission to be protected in integrity.

OT.SIGNATURE_INTEGRITY is handled by FCS_COP.1, which describes requirements and the algorithms. FTP_ITC.1/CM may be used to transmit data securely between the TOE and the Cryptographic Module. Access control for the signature operation is ensured by FDP_ACC.1/Signing and FDP_ACF.1/Signing.

OT.CRYPTO is covered by FCS_CKM.1 and FCS_COP.1, which describes requirements for key generation and algorithms.

OT.RANDOM is handled by FCS_RNG.1, which describes requirement on the random number generation.

9 TOE SUMMARY SPECIFICATION (ASE_TSS)

This section describes how the TOE meets each SFR by providing, for each SFR from the statement of security requirements, a description of the function behaviours. This section provides potential consumers of the TOE with a high-level view of how each SFR is satisfied.

9.1 TOE Security Functions Specification

This section describes the security functions provided by the TOE to meet the security functional requirements specified for the TOE in Section 8.

List of security functions

- Security Audit
- Cryptographic support
- User Data Protection
- Identification and Authentication
- Security Management
- TSF Protection
- Trusted Path-Channels

9.1.1 Security Audit

The TOE provides a capability to generate audit events protected in integrity. The TOE security function Security Audit meets the audit requirements **FAU_GEN.1** and **FAU_GEN.2**.

Security Function Requirement	Implementation
FAU_GEN.1 Audit Generation	The Audit Function produces an audit record for each security relevant event within the TOE. Each audit record contains information about the task performed, when it was performed and who performed it. These records are stored externally to the TOE and protected in integrity. The Audit log begins at the start of the TOE and it is stopped at the shutdown. Subsystems of the TOE generate audit records which are transferred outside the TOE for storage purposes. Each record is signed in order to guarantee integrity.
FAU_GEN.2 User identity association	Each Audit events record includes the subject that caused the event.

9.1.2 Cryptographic Support

The TOE is a composite product including both the cryptographic module and the software application running inside its CPU. Cryptographic operations are predisposed by the software application and carried out by the CM. The sensitive key material is never shared unencrypted outside of the secure perimeter of the CM, not even with the application. The communication regarding the operations to execute and the data to elaborate occur by the usage of cryptographic libraries provided by the CM vendor, which are integrated in the software application.

Cryptographic support meets the requirements as per the following Table.

Security Function Requirement	Implementation
FCS_CKM.1 Cryptographic key generation	The TOE implements its key generation function by relying on the cryptographic module certified in conformance with [5]. The TOE invokes the cryptographic module with the appropriate parameters whenever the key generation is required, through the usage of cryptographic libraries provided by the CM vendor and integrated into the software application. The TOE shall use RSA keys or ECC keys for the digital signature service. The keys generated have size as recommended by [12][13].
FCS_CKM.4 Cryptographic key destruction	The TOE provides a mechanism for key destruction. The TOE implements its key destruction function by relying on the cryptographic module certified in conformance with [5], which carries out the effective key destruction by the zeroisation method. The TOE invokes the cryptographic module with the appropriate parameters whenever key destruction is required, through the usage of cryptographic libraries provided by the CM vendor and integrated into the software application.
FCS_COP.1 Cryptographic operation	The TOE implements its digital signature creation function by relying on the cryptographic module certified in conformance with [5]. The TOE invokes the cryptographic module with the appropriate parameters whenever digital signature creation is required, through the usage of cryptographic libraries provided by the CM vendor and integrated into the software application. The TOE can use RSA keys or ECC keys for the digital signature service. The TOE performs digital signature creation using strong cryptographic algorithms that fully reflect the state of the art in cryptography, providing an adequate level of security against all presently known or conjectured threats even considering the generally expected increases in computing power.

Security Function Requirement	Implementation
FCS_RNG.1 Generation of random numbers	The TOE has the same physical boundary as the cryptographic module certified in conformance with [5]. The TOE relies on the cryptographic module for the generation of random numbers, through the usage of cryptographic libraries provided by the CM vendor and integrated into the software application. The CM provides a physical random number generator (see specifics in [20]).

9.1.3 User Data Protection

The TOE ensures user data protection and maintains full control over the information transmitted. To do this, all operations are regulated by authorizations and compliance with TSP policies. The authorization takes the form of a SAD, an object signed asymmetrically by the EAS or delegated party with a private key, leveraging on the security given by HSM devices. The SAD is thus verified in integrity and it binds together the Signer authentication, the material to be signed and the identifier of the signing key.

The Privileged User is the only subject allowed to create other Privileged Users and Signers, provided he/she is given the authorization to do so. He/she is responsible for transmitting information to the TOE for managing roles and configurations. The Signer is created within the TOE and the maintenance of both Privileged Users and Signers is guaranteed in integrity and confidentiality.

Privileged Users can obtain the authorization to generate a key pair on behalf of a Signer on the Cryptographic module. Security is ensured also on signature operations: when signing, a key belonging to an authorized user is loaded, used for signing by cryptographic module and then unloaded. It is destroyed when deemed useless, i.e. after expiration date.

The TOE supports access operation with the subject having attributes as administrator roles and user roles. The TOE security function User Data Protection meets the protection of user data requirements as per following Table.

Security Function Requirement	Implementation

Security Function Requirement	Implementation
FDP_ACC.1/Privileged User Creation Subset access control	The TOE allows only the Privileged Users with specific role to create other Privileged Users, stored in **R.TSF_DATA**. The Privileged Users authorized for this, create other Privileged Users and associate some attributes to them, including **R.Reference_Privileged_User_Authentication_Data** and roles. In the initialisation phase, an initial Privileged User accesses the TOE for establishing other Privileged Users and starting-up the TOE, thus unlocking it for regular operation.
FDP_ACF.1/Privileged User Creation Security attribute-based access control	Only a Privileged User with specific roles can create other Privileged Users.
FDP_ACC.1/Signer Creation Subset access control	Only the Privileged User with specific role can create a Signer. At creation time, the only attributes associated with the Signer are **R.Signer**, key lock policy and authenticator, the latter identifying the External Authentication Service that will provide authorizations to the Signer. The **R.Reference_Signer_Authentication_Data** is the SVD of the defined External Authentication Service and it is bound to the R.Signer(s) of such domain.
FDP_ACF.1/ Signer Creation Security attribute-based access control	Only the Privileged User with specific role can bring the operation of creating a Signer to a successful conclusion.
FDP_ACC.1/Signer Maintenance Subset access control	The Privileged User can instruct the TOE to update **R.Reference_Signer_Authentication_Data.** The old and new **R.Reference_Signer_Authentication_Data** can coexist for a limited period that is until the expiration date of the older. Signer cannot instruct the TOE to update **R.Reference_Signer_Authentication_Data** since the indirect schema is adopted.
FDP_ACF.1/Signer Maintenance Security attribute-based access control	The Privileged User can add and remove the public keys of the delegated authenticator referred as the **R.Reference_Signer_Authentication_Data**. Different public keys can exist for the same delegated authenticator.

Security Function Requirement	Implementation
FDP_ACC.1/Signer Key Pair Generation Subset access control	The Privileged User can instruct the TOE to create a Signer keypair through SSA. The TOE requests an association of **R.SVD** and **R.Signing_Key_Id** to the Cryptographic Module and binds them to **R.Signer**. The **R.Authorisation_Data** is provided in input by the Privileged User on behalf of the Signer. The signing keys can be used by several cryptographic modules. Outside the module, the signing keys are protected by encryption by means of cryptographic keys securely stored inside the cryptographic modules themselves.
FDP_ACF.1/Signer Key Pair Generation Security attribute-based access control	A Privileged User can request the creation of a keypair on behalf of a Signer. The Cryptographic Module creates **R.SVD** and **R.Signing_Key_Id**, and the TOE binds them to **R.Signer**. The Signer is not given the possibility to request the keypair creation for other Signers or himself. No pre-generated keys are used by the TOE.
FDP_ACC.1/Signer Key Pair Deletion Subset access control	Both a Privileged User and a Signer can request the keypair deletion. The Signer can request only its own keypair deletion. The Privileged User can request the deletion of a keypair associated with any Signer only if he/she is in possess of a specific role. The **R.Signing_Key_Id** and **R.SVD** associated with **R.Signer** are destroyed from the Cryptographic Module and from the Signer data, along with the certificate and its related data.
FDP_ACF.1/Signer Key Pair Deletion Security attribute-based access control	An authorised Privileged User can carry out the keypair deletion. The Signer can carry out the deletion only of its own keypair.
FDP_ACC.1/Supply DTBS/R Subset access control	The Privileged User cannot supply a R.DTBS/R on behalf of Signer.
FDP_ACF.1/Supply DTBS/R Security attribute-based access control	No Privileged User has a role for supplying a R.DTBS/R on behalf of a Signer.

Security Function Requirement	Implementation
FDP_ACC.1/Signing Subset access control	The Signer calls the signing method by providing **R.SAD**, which contains the identifier **R.Signer** as part of R.SAD. If the TOE accepts the signing request, the TOE performs the following steps for signing: 1. Retrieves the **R.Signing_Key_Id** associated with **R.Signer** 2. Open the session with the CM 3. Loads the slot identified by **R.Signing_Key_Id** with **R.Authorisation_Data** and the **R.DTBS/R** 4. Instructs CM to sign **R.DTBS/R** 5. Close the session with the CM
FDP_ACF.1/Signing Security attribute-based access control	The **R.SAD** integrity and confidentiality is guaranteed by the **R.Reference_Signer_Authentication_Data** bound to the **R.Signer** used to perform the requested operations. The Signer is authenticated by using the indirect authentication schema which verifies the **R.Reference_Authentication_Data**. In case of multiple signatures, a mechanism for creating **R.DTBS/R** signature at each iteration is provided. This is implemented by using a refresh **R.SAD** based on OAuth2 protocol flow with authorization code grant mechanism, described in [14], paragraph 1.5, figure 2. The new **R.SAD** is used to request another signature session containing the new **R.DTBS/R**. **R.DTBS/R** is always contained into **R.SAD** to ensure its integrity. A Signer without a **R.Signer** object does not sign a **R.DTBS/R**.
FDP_ACC.1/TOE Maintenance Subset access control	Only the Privileged Users with a specific role are allowed to modify **R.TSF_DATA**.
FDP_ACF.1/TOE Maintenance Security attribute-based access control	Only Privileged Users with specific role can maintain **R.TSF_DATA**. No other subjects or Privileged Users without role are authorized to maintain **R.TSF_DATA**.

Security Function Requirement	Implementation
FDP_ETC.2/Signer Export of user data with security attributes	Signer can export his/her own keys and certificate, which contains the security attribute **R.SVD**, bound to **R.Signer**. He/she also exports **R.Signature**.
FDP_IFC.1/Signer Subset information flow control	An internal Signer flow is established to permit the TOE to operate in a controlled environment, whereas subjects, information and operations are executed under a predefined flow.
FDP_IFF.1/Signer Simple security attributes	A Signer Flow SFP is defined on Privileged User and Signer accessing Signer security attributes. The macro-steps are the following: - Privileged User initializes the TOE before starting it in running condition. The **R.TSF_DATA** is configured with all mandatory data. - Privileged User creates Signer in the TOE so that only controlled subjects can use the TOE. Only after keypair generation and certificate import, the Signer can request a signature. The above functions are the logical sequence to set up a Signer completely. Afterwards, all the remaining operations can be executed according to Signer's needs: Signer maintenance and signing. Of course, the TOE will respond depending on Signer's configuration, i.e. signing will not be allowed if Signer is locked or the Privileged User has deleted the keypair on behalf of the Signer.
FDP_ETC.2/Privileged User Export of user data with security attributes	Privileged User can export **R.Reference_Privileged_User_Authentication_Data**, **R.TSF_DATA** and Signer certificate which contains the security attribute **R.SVD**, bound to **R.Signer**.
FDP_IFC.1/Privileged User Subset information flow control	An internal Privileged User flow is established to permit the TOE to operate in a controlled environment, whereas subjects, information and operations are executed under a predefined flow.

Security Function Requirement	Implementation
FDP_IFF.1/Privileged User Simple security attributes	A Privileged User Flow SFP is defined on Privileged User accessing Privileged User security attributes. The macro-step is the following: - Privileged User initializes the TOE before starting it in running condition. The **R.TSF_DATA** is configured with all the mandatory data.
FDP_ITC.2/Signer Import of user data with security attributes	The Signer imports his/her own keys and certificate data containing the security attribute **R.SVD** bound to **R.Signer**. The TOE checks the integrity of these data, which have a signature attached, by verifying the signature with its integrity key.
FDP_ITC.2/Privileged User Import of user data with security attributes	Privileged User imports **R.Reference_Privileged_User_Authentication_Data**, **R.TSF_DATA**, **R.Signer** and the Signer certificate, which contains the security attribute **R.SVD**.
FDP_UCT.1 Basic data exchange confidentiality	The communication channel between the TOE and SSA and vice versa is protected by TLS-protocol and the tamper-protected environment. Thus, the data transmitted in input are protected in confidentiality. Audit logs don't require confidentiality, they are protected in integrity by the TOE signature. The TOE controls the validity of the data in transit by verifying if **R.Reference_Signer_Authentication_Data** has a valid signature for Signer requests, and by verifying **R.Reference_Privileged_User_Authentication_Data** and roles for the **R.Privileged_User** requests.
FDP_UIT.1 Data exchange integrity	The **R.Reference_Signer_Authentication_Data** guarantees the integrity of the data contained into **R.SAD**, so that malicious accesses and/or alteration of data are prevented. The verification with EAS SVD is the mechanism used to verify integrity. For Privileged Users, the authenticity and integrity of the data in transit is guaranteed by the signature made with a Privileged User private key. The signature is verified with the corresponding public key stored into **R.TSF_DATA**.

9.1.4 Identification and Authentication

The Signer authentication is carried out applying indirect method. The delegated system of identification and authentication for a Signer ensures the protection of the resources within the

TOE from a malicious access. It consists of a signed assertion, issued by the authorization service, which contains R.Signer, the authorization data and the material to be signed. The verification is performed by validating the signature.

The Privileged user authentication is carried out applying indirect method. Privileged User provides requests to the TOE which are signed by his/her own private key. The TOE authenticates the Privileged User by retrieving his/her data from configuration, according to his/her claimed identity, and using the public key in configuration to verify the provided request signature. A higher level of security is given by the environment in which the TOE resides. Rules are defined for the other Privileged Users creation and his/her initial association of security attributes and roles. Roles define also whether a Privileged User is enabled to create a Signer, another Privileged User or both.

The TOE security function Identification and Authentication meets the requirements as per the following Table.

Security Function Requirement	Implementation
FIA_AFL.1/Authentication failure handling	Both Signers and Privileged Users authenticate adopting an indirect authentication mechanism. Therefore, this SFR is not applicable to the TOE.
FIA_ATD.1 User attribute definition	The TOE stores and maintains the security attributes mentioned in **FIA_USB.1** into an external database. Related to the Signer: • **R.Reference_Signer_Authentication_Data** (the delegated authorization assertion). It is maintained by the Privileged User, • **R.Signing_Key_Id** (the Signer reference private key identifier), • **R.SVD** (the Signer public key), • **R.Signer** (the identifier of the Signer chosen as the primary key). Once defined it cannot be changed, • **R.Authorisation_Data** (provided by Signer when required and never kept in database or elsewhere). Related to the Privileged User: • **R.Reference_Privileged_User_Authentication_Data** (the Privileged User signed assertion), • **R.Privileged_User** (the unique identifier of the Privileged User).

Security Function Requirement	Implementation
FIA_UAU.1 Timing of authentication	The TOE does not permit any action without authentication. Both Signer and Privileged User must be authenticated before accessing any resource, the Signer with **R.Reference_Signer_Authentication_Data** and the Privileged User with **R.Reference_Privileged_User_Authentication_Data**.
FIA_UAU.5/Signer Multiple authentication mechanisms	The delegated authentication consists of a signed assertion, **R.Reference_Signer_Authentication_Data**, issued by the external authorization service and signed according to the requirement [4] **SRA_SAP.1.1**. The verification is performed on the **R.Reference_Signer_Authentication_Data** signature by using one of the public keys allowed by the TOE.
FIA_UAU.5/Privileged User Multiple authentication mechanisms	The TOE provides an authentication mechanism consisting of a signed assertion with **R.Reference_Privileged_User_Authentication_Data** to Privileged Users. The TOE verifies the provided **R.Reference_Privileged_User_Authentication_Data** by querying **R.TSF_DATA** to check if **R.Privileged_User** is present and by retrieving the corresponding public key to validate the assertion signature. If the signature is verified, the authentication succeeds.
FIA_UID.2 User identification before any action	The Signer is identified by **R.Signer**. The TOE retrieves the **R.Signer** contained in the **R.SAD** from the Signer data provided in input and protected in integrity. The Privileged User is identified with **R.Privileged_User** and **R.Reference_Privileged_User_Authentication_Data** and roles stored into **R.TSF_DATA**.

Security Function Requirement	Implementation
FIA_USB.1 User-subject binding	The Privileged User creates a new **R.Signer** and bound it to **R.Reference_Signer_Authentication_Data.** At this stage **R.SVD**, **R.Signing_Key_ID** and **R.Authorisation_Data** are not created yet. The Privileged User will create and bind them later, during the enrollment process. **R.Authorisation_Data** is passed to the TOE by the Privileged User on behalf of the Signer. The Privileged User cannot modify **R.Signer** object once it is created. The Signer can modify his own **R.Authorisation_Data** object only if he/she is the owner. Privileged Users can create other Privileged Users only if they have a specific role. These subjects can decide the roles of a Privileged User.

9.1.5 Security Management

The security management function deals with the management of Signers, Privileged Users, configurations and imposes restrictions on the operations that can be performed by Signers and Privileged Users. The management of security attributes and roles are restricted for Signers and for those Privileged Users who are not authorized.

Signers and Privileged Users are created with restricted default security attributes. These two distinct figures cannot collide, they have different defined security roles and user-roles associations.

The TOE security function Security Management meets the requirements as per the following Table.

Security Function Requirement	Implementation

Security Function Requirement	Implementation
FMT_MSA.1/Signer Management of security attribute	The authorized Privileged User creates a Signer as specified in FIA_USB.1 and generates the key pair on behalf of the Signer. SSA passes the keypair generation request to the TOE and, according to its data, the TOE carries out the keypair generation in the CM. The TOE also gives the possibility of deleting a Signer's keypair. A Privileged User can destroy a Signer's **R.SVD** and **R.Signing_Key_Id** after successful authentication and authorization. This request needs to be done by providing **R.Reference_Privileged_User_Authentication_Data.** The Privileged User cannot supply a DTBS/R on behalf of the Signer. The **R.DTBS/R** (the hashes to be signed) are bound to other data included in the **R.SAD**. The creation of the **R.DTBS/R** is external to the TOE. The **R.Reference_Signer_Authentication_Data** cannot be changed by the Signer, only an authorized Privileged User can do it.
FMT_MSA.1/Privileged User Management of security attributes	Only a Privileged User with specific role can create the security attributes mentioned in FIA_USB.1 for Privileged Users and query them.

<table>
<tr><th>Security Function Requirement</th><th>Implementation</th></tr>
<tr><td>FMT_MSA.2 Secure security attributes</td><td>
The TOE performs security validation checks on the data received in input, while SSA performs the formal validation by matching data against pre-defined regular expressions and constraints set in SSA. SSA informs the TOE that these checks have been executed:

- R.Signer contains a value made of letters and numbers with an established length;
- R.Authorization_Data is made of digits of an established length. The TOE does not control its correctness before executing a real signature using the signing key, it’s only a formal check. The validity check will be done during the signature process by the cryptographic module.

The validation of values is carried out for R.Signer and R.Reference_Signer_Authentication_Data.
For R.Reference_Signer_Authentication_Data the signature is verified by the TOE against the public key of the external authorization system stored into R.TSF_DATA.
R.SVD and R.Signing_Key_Id are TOE assets generated by the cryptographic module which don’t require validation.
Summing up, for Signer’s security attribute the following checks are applied:

<table>
<tr><th>Data</th><th>Formal check</th><th>Value check</th><th>Coherence check</th><th>Internal data</th></tr>
<tr><td>R.Reference_Signer_Authentication_Data</td><td></td><td>X</td><td>X</td><td></td></tr>
<tr><td>R.Signing_Key_Id</td><td></td><td></td><td></td><td>X</td></tr>
<tr><td>R.SVD</td><td></td><td></td><td></td><td>X</td></tr>
<tr><td>R.Signer</td><td>X</td><td>X</td><td>X</td><td></td></tr>
<tr><td>R.Authorisation_Data</td><td>X</td><td>X</td><td></td><td>X</td></tr>
</table>

For Privileged User’s security attributes, security validation is carried out by the TOE against the data contained into R.TSF_DATA.
</td></tr>
</table>

Security Function Requirement	Implementation
FMT_MSA.3/Signer Static attribute initialisation	The TOE accepts and verifies that when a Signer is created, some data are associated to it in the corresponding Signer data protected in integrity. These are: • **R.Signer**, which is randomly generated by the TOE • User Status, which is initially set to CREATED • Key policy, which defines how many signature wrong attempts are tolerated before blocking temporarily and definitively the keys • Authenticator, which is the external authorization system associated to the Signer that verifies the delegated authorization The Signer Creation SFP allows **R.Signer** to be created according to the initial default values. No alternative initial values to override the default values are allowed.
FMT_MSA.3/Privileged User Static attribute initialisation	The Privileged User Creation SFP establishes that a Privileged User is created with one or more roles according to the [Table 14 - Roles vs Operations].
FMT_MTD.1 Management of TSF data	The TOE permits any changes to **R.TSF_DATA** by Privileged User having specific role as specified in [Table 14 - Roles vs Operations].
FMT_SMF.1 Specification of Management Functions	A Privileged User can modify Signers' and/or Privileged Users' security attributes only if it has specific roles. Signer management can be carried out by the Signer or the Privileged User according to related data and roles as figured out in the above clauses. Signer creation, keypair generation and keypair deletion are carried out by the Privileged User. Privileged Users' and configuration's management can be carried out by Privileged Users with specific role.

Security Function Requirement	Implementation
FMT_SMR.2 Restrictions on security roles	The TOE maintains the association between a Privileged User and its roles, as well as Signers' data integrity. The Signer can have only one role: SIGNER. Privileged User can have one or more roles as defined in [Table 14 - Roles vs Operations] and the Privileged User with specific role can associate different roles to other Privileged Users. Signers and Privileged Users are distinct types of users with distinct authentication systems and roles. A Signer cannot be associated with roles related to Privileged Users.

9.1.6 Protection of the TSF (FPT)

The TOE is a composite product including both the cryptographic module and the software application running inside its CPU. The TOE physical boundary coincides with the one of the CM, which is certified both Common Criteria EAL4+ (see paragraph 2.3.4) and FIPS 140-2 Level 3. Therefore, the physical protection of the TOE relies on the physical protection provided by the cryptographic module.

The TOE security function FPT Protection meets the requirements as per the following Table.

Security Function Requirement	Implementation
FPT_PHP.1 Passive detection of physical attack	The TOE subsystems rely on the physical protection given by the CM, as the TOE includes the CM. The CM is capable to determine whether physical tampering with its TSF's devices or TSF's elements has occurred. Whether a detection of physical tampering occurs, the CM (and therefore the TOE) stops its normal operation.
FPT_PHP.3 Resistance to physical attack	As the TOE includes the CM, it relies on the physical boundary protection of the HSM. The CM (and therefore the TOE) is protected by physical attack such as opening the appliance. The CM is capable to resist physical penetration attempts to the hard opaque potted enclosure by automatically responding such that the SFRs are always enforced.

Security Function Requirement	Implementation
FPT_RPL.1 Replay detection	The TOE identifies and rejects replayed requests of R.SAD.
FPT_STM.1 Reliable time stamps	The reliable time stamps are provided to the TOE by the real-time internal clock of the CM.
FPT_TDC.1 Inter-TSF basic TSF data consistency	The TOE interprets the data before using them by checking data format and/or data consistency or value whenever it is applicable. The integrity of data exchanged with other trusted IT products is provided by verifying the data signature made with a TOE integrity key.

9.1.7 Trusted Path-Channels

Data transmitted from the TSF to another trusted IT product is protected by the Operational Environment from unauthorized disclosure during transmission.

The TOE meets the protection of the TSF requirements as per the following Table.

Security Function Requirement	Implementation
FTP_TRP.1/SSA Trusted path	The TOE communication with SSA occurs inside the same tamper-protected environment. The communication path for Privileged Users is logically distinct from the Signers' one and provides protection of the communicated data from disclosure and tampering. All the communications between the SSA and the TOE are protected in integrity using a symmetric encryption algorithm with a TOE integrity key. All the listed functions in FTP_TRP.1.3/SSA are permitted.

Security Function Requirement	Implementation
FTP_TRP.1/SIC Trusted path	The SIC channel is implemented in two parts: from SIC to SSA, and from SSA to the TOE. SIC doesn't have a single channel which directly ends into a TOE endpoint, as the SSA acts as an intermediary. Between SIC and SSA there's one of these two communication protocols: • mutual TLS • HTTPS connection + basic authentication credentials Between SSA components outside the physical protected environment and the ones inside, there's mutual TLS. From SSA components inside the same physical protected environment of the TOE and the TOE there's no need of mutual TLS because the security relies on the secure environment.
FTP_ITC.1/CM trusted channel	The TOE operates in the same physical boundary of the cryptographic module, relying on the local nature of the communication to ensure integrity and confidentiality protection. No additional authentication or cryptographic protection are required.

9.2 SFRs to Security Functions Coverage

Functional Requirements	Security Audit	Cryptographic Support	User Data Protection	Identification Authentication	Security Management	Protection of the TSF	Trusted Path/Channels
Security Audit (FAU)							
FAU_GEN.1 Audit Generation	X						
FAU_GEN.2 User identity association	X						
Cryptographic Support (FCS)							
FCS_CKM.1 Cryptographic key generation		X					
FCS_CKM.4 Cryptographic key destruction		X					
FCS_COP.1 Cryptographic operation		X					
User Data Protection (FDP)							
FDP_ACC.1/Privileged User Creation Subset access control			X				
FDP_ACF.1/Privileged User Creation Security attribute based access control			X				
FDP_ACC.1/Signer Creation Subset access control			X				
FDP_ACF.1/ Signer Creation Security attribute based access control			X				
FDP_ACC.1/Signer Maintenance Subset access control			X				
FDP_ACF.1/Signer Maintenance Security attribute based access control			X				
FDP_ACC.1/Signer Key Pair Generation Subset access control			X				
FDP_ACF.1/Signer Key Pair Generation Security attribute based access control			X				
FDP_ACC.1/Signer Key Pair Deletion Subset access control			X				

Functional Requirements	Security Audit	Cryptographic Support	User Data Protection	Identification Authentication	Security Management	Protection of the TSF	Trusted Path/Channels
FDP_ACF.1/Signer Key Pair Deletion Security attribute based access control			X				
FDP_ACC.1/Supply DTBS/R Subset access control			X				
FDP_ACF.1/Supply DTBS/R Security attribute based access control			X				
FDP_ACC.1/Signing Subset access control			X				
FDP_ACF.1/Signing Security attribute based access control			X				
FDP_ACC.1/TOE Maintenance Subset access control			X				
FDP_ACF.1/TOE Maintenance Security attribute based access control			X				
FDP_ETC.2/Signer Export of user data with security attributes			X				
FDP_IFC.1/Signer Subset information flow control			X				
FDP_IFF.1/Signer Simple security attributes			X				
FDP_ETC.2/ Privileged User Export of user data with security attributes			X				
FDP_IFC.1/Privileged User Subset information flow control			X				
FDP_IFF.1/Privileged User Simple security attributes			X				
FDP_ITC.2/Signer			X				
FDP_ITC.2/ Privileged User Import of user data with security attributes			X				
FDP_UCT.1 Basic data exchange confidentiality			X				
FDP_UIT.1 Data exchange integrity			X				
Identification and Authentication (FIA)							
FIA_AFL.1 Authentication failure handling				X			
FIA_ATD.1 User attribute definition				X			

Functional Requirements	Security Audit	Cryptographic Support	User Data Protection	Identification Authentication	Security Management	Protection of the TSF	Trusted Path/Channels
FIA_UAU.1 Timing of authentication				X			
FIA_UAU.5/Signer Multiple authentication mechanisms				X			
FIA_UAU.5/Privileged User Multiple authentication mechanisms				X			
FIA_UID.2 User identification before any action				X			
FIA_USB.1 User-subject binding				X			
Security Management (FMT)							
FMT_MSA.1/Signer Management of security attributes					X		
FMT_MSA.1/Privileged User Management of security attributes					X		
FMT_MSA.2 Secure security attributes					X		
FMT_MSA.3/Signer Static attribute initialisation					X		
FMT_MSA.3/Privileged User Static attribute initialisation					X		
FMT_MTD.1 Management of TSF data					X		
FMT_SMF.1 Specification of Management Functions					X		
FMT_SMR.2 Restrictions on security roles					X		
Protection of the TSF (FPT)							
FPT_PHP.1 Passive						X	
FPT_PHP.3 Resistance						X	
FPT_RPL.1 Replay detection						X	
FPT_STM.1 Reliable time stamps						X	
FPT_TDC.1 Inter-TSF basic TSF data consistency						X	

Functional Requirements	Security Audit	Cryptographic Support	User Data Protection	Identification Authentication	Security Management	Protection of the TSF	Trusted Path/Channels
Trusted Paths/Channels (FTP)							
FTP_TRP.1/SSA Inter-TSF Trusted path							**X**
FTP_TRP.1/SIC Inter-TSF Trusted path							**X**
FTP_ITC.1/CM Inter-TSF trusted channel							**X**