SECURITY TARGET

FOR

ASCERTIA ADSS SERVER
SIGNATURE ACTIVATION MODULE (SAM)

Common Criteria version 3.1 revision 5
Assurance Level EAL 4+

Table of Contents

1. ST Introduction

1.1. ST Overview

This Security Target (ST) document defines the security objectives and requirements, as well as the scope of the Common Criteria evaluation of Ascertia ADSS Server Signature Activation Module (SAM) product.

The Target of Evaluation (TOE) is the Ascertia ADSS Server SAM software product, which consists of the following main software components:

- ADSS Server SAM Service – handles user registration and signing requests through its API interface
- ADSS Server SAM Admin Console – allows trusted administrators to configure the product and perform maintenance / housekeeping tasks
- ADSS Server SAM Core – performs automated background tasks

and the following non-software components:

- Operational User Guidance [23]
- Preparation Procedure [24]

The TOE is delivered within a tamper-protected hardware device to ensure a secure execution environment. The TOE provides a remote Qualified Electronic Signatures and Seals (referred to collectively as QES) service according to eIDAS Regulation No 910/2014 [8] at Sole Control Assurance Level 2 (SCAL2) according to EN 419 241-1 [6].

This ST addresses the following advanced security mechanisms: -

- Remote QES according to eIDAS Regulation No 910/2014 [8]; and
- Sole Control Assurance Level 2 (SCAL2) according to sec. 5.4 of EN 419 241-1 [6].

1.2. ST Reference

This ST is identified by the following unique reference: -

ST Title:	ASE: Security Target
ST Version	V18
ST Date:	2018-10-01
ST Author:	Ascertia Limited

1.3. TOE Reference

The TOE is identified by the following unique reference: -

TOE Name	Ascertia ADSS Server Signature Activation Module (SAM) product
TOE short name	ADSS Server SAM
TOE Version	v6.0
Evaluation Criteria	Common Methodology for Information Technology Security Evaluation, Evaluation Methodology; CCMB-2017-04-004, Version 3.1, Revision 5, April 2017
Protection Profile(s)	EN 419 241-2 [5]
Evaluation Assurance Level	EAL 4 augmented by AVA_VAN.5
Developer	Ascertia Limited
Evaluation Sponsor	Ascertia Limited
Evaluation Facility	CCLab Software Laboratory
Certification Body	OCSI
Certification ID	

1.4. TOE Overview

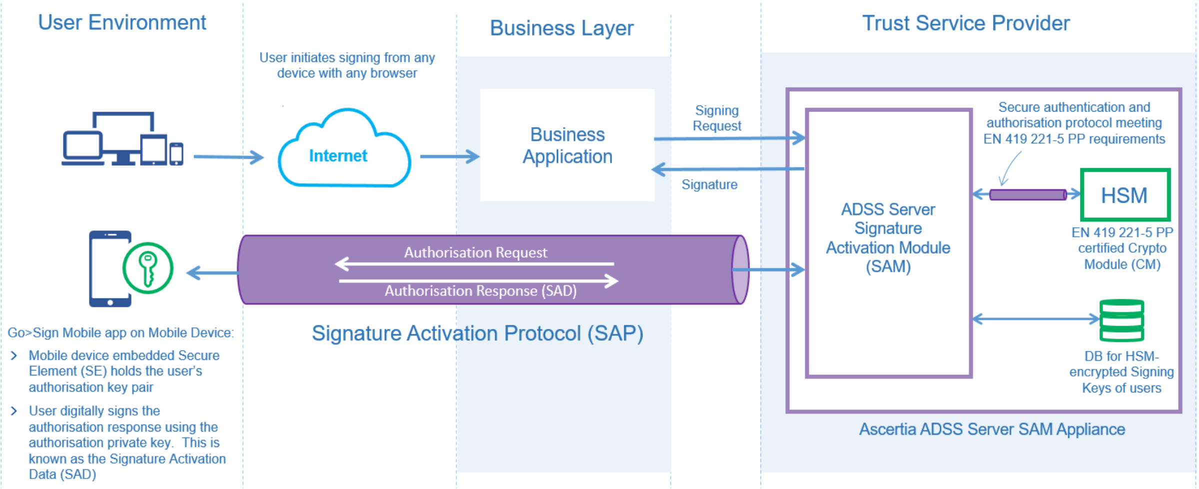
The TOE addressed in this ST is the Ascertia ADSS Server Signature Activation Module (SAM) product.

The Ascertia ADSS Server SAM product is a Trustworthy System Supporting Server Signing (TW4S) that offers remote digital signature services. It ensures that the Signer's signing key or keys are only used under the sole control of the Signer and only used for the intended purpose.

The Ascertia ADSS Server SAM remote signing solution consists of a local and a remote environment as illustrated below.

Signer can initiate a signing transaction by interacting through a Business Application.

Remote Signing eIDAS Compliant Architecture


User Environment
Business Layer
Trust Service Provider
User initiates signing from any device with any browser
Internet
Business Application
Signing Request
Signature
Secure authentication and authorisation protocol meeting EN 419 221-5 PP requirements
ADSS Server Signature Activation Module (SAM)
HSM
EN 419 221-5 PP certified Crypto Module (CM)
Authorisation Request
Authorisation Response (SAD)
Signature Activation Protocol (SAP)
DB for HSM-encrypted Signing Keys of users
Ascertia ADSS Server SAM Appliance
Go>Sign Mobile app on Mobile Device:
Mobile device embedded Secure Element (SE) holds the user's authorisation key pair
User digitally signs the authorisation response using the authorisation private key. This is known as the Signature Activation Data (SAD)

Signers interact with the Business Application normally via a web browser. To sign a document, the Signer has to use a mobile app, named the Go>Sign Mobile app, on a registered mobile device to securely authorise the server-side signing action using a trusted protocol referred to as the Signature Activation Protocol (SAP).

To be able to authorise a remote signature, the mobile device is first registered with the TOE against the user's (Signer) account. The signing authorisation is at Sole Control Assurance Level 2 (SCAL2) according to EN 419 241-1 [6] for qualified signatures.

During a remote signing operation, the Signer will be asked to launch the Go>Sign Mobile app to download the authorisation request for the user to view and authorise.

In order for the Go>Sign Mobile app to sign the authorisation response, the Signer must authenticate to the mobile device in order to access the authorisation key held in the mobile device's embedded Secure Element/Enclave (SE). The user authenticates to the mobile device using iOS/Android biometric techniques (e.g. TouchID or FaceID) or the mobile device passcode.

After the successful authentication of the Signer, the Go>Sign Mobile app digitally signs the authorisation response XML using the Authorisation Key held in the mobile device's SE. The signed authorisation response is referred to as the Signature Activation Data (SAD). The SAD represents 2-factor authentication because it relies on the fact that the user must use their registered mobile device ("something that the user has") and authenticate to the mobile device using either biometric authentication ("something the user is") or the mobile device PIN ("something the user knows").

The signed SAD value cryptographically binds together:

- The Signer's identity and authentication (via the signature on the SAD value)
- The Business Application ID which requested the signature

- The unique transaction ID
- The User’s remote Signing Key ID held within the TOE
- The hash of the data to be signed remotely (DTBS/R)
- Random salt information to prevent replay attacks
- Mobile device ID used to create the SAD
- Validity date/time for the authorisation response

Each of the above elements within the signed SAD are verified by the TOE before allowing the Signer’s key to be used for signing the data identified in the SAD. Note the SAD also contains some additional detailed information not described above.

The TOE is responsible for maintaining the details of: -

- Registered users (signers);
- Authorised mobile devices;
- Authorisation requests/responses; and
- Current statuses of the above described features.

In addition to this, the TOE cryptographically verifies every signed authorisation response XML (SAD) before the remote signing is performed by the Ascertia ADSS Server SAM. The signing key is activated in a Cryptographic Module (CM).

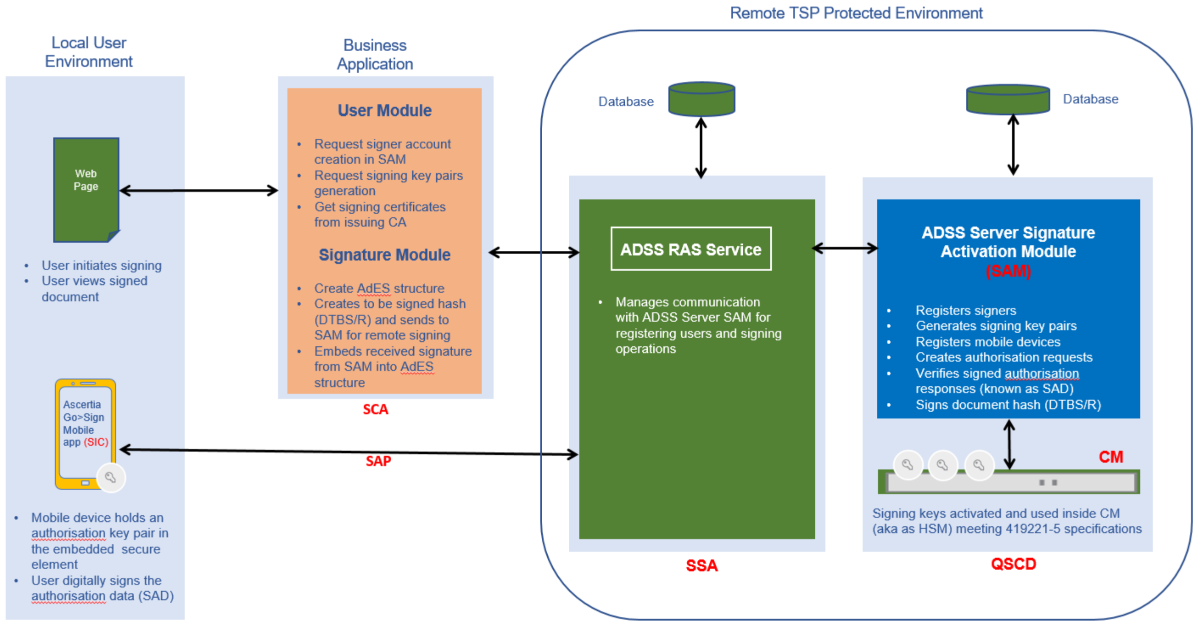
The Ascertia ADSS Server SAM is located within its own tamper protected environment referred to as the ADSS Server SAM appliance. The CM is located inside the appliances tamper protected environment. The TOE is connected to the CM through a trusted channel. The CM is shipped as an embedded module inside the ADSS Server SAM appliance, although even in this case it is important to note that the CM has its own tamper protected environment and the same trusted channel is used for the communication as if it was an external device.

SAD verification means that the ADSS Server SAM checks the following bindings:

- The signature on the SAD can be verified by the authorisation certificate belonging to the Signer for this registered mobile device (note the Signer may multiple devices registered for remote authorisation purposes each with their own authorisation key pair and certificate)
- The Signer’s ID in the SAD is correct
- The Business Application ID is correct
- The unique transaction ID is correct
- The remote Signing Key ID is available and belongs to this Signer
- The hash of the document (DTBS/R) matches that which the Business Application is requesting to sign on behalf of the user
- The random salt information in the SAD matches that provided in the authorisation request message to ensure this is not a replay
- The mobile device ID is recognised as a registered device for this Signer
- The validity period in the response is not expired according to current date/time

Note the TOE also various some additional items from the SAD which are not mentioned above.

The following Figure shows the main components of the ADSS Server SAM solution in more detail.


Local User Environment
Web Page
User initiates signing
User views signed document
Ascertia Go>Sign Mobile app (SIC)
Mobile device holds an authorisation key pair in the embedded secure element
User digitally signs the authorisation data (SAD)
Business Application
User Module
Request signer account creation in SAM
Request signing key pairs generation
Get signing certificates from issuing CA
Signature Module
Create AdES structure
Creates to be signed hash (DTBS/R) and sends to SAM for remote signing
Embeds received signature from SAM into AdES structure
SCA
SAP
Remote TSP Protected Environment
Database
ADSS RAS Service
Manages communication with ADSS Server SAM for registering users and signing operations
SSA
Database
ADSS Server Signature Activation Module (SAM)
Registers signers
Generates signing key pairs
Registers mobile devices
Creates authorisation requests
Verifies signed authorisation responses (known as SAD)
Signs document hash (DTBS/R)
CM
Signing keys activated and used inside CM (aka as HSM) meeting 419221-5 specifications
QSCD

Note: Term "document" refers to a document or transaction

It is important to note that all API interactions with the ADSS Server SAM must be conducted through the ADSS RAS Service. This means from the perspective of the TOE, the ADSS RAS Service is considered as a Business Application. The TOE authenticates the ADSS RAS Service as a Business Application before allowing access to its services. Throughout this document the term Business Application applies to the ADSS RAS Service as well.

1.4.1. TOE Usage & Major Security Features

The Ascertia ADSS Server SAM solution ensures that the remote Signer has sole control of his signing keys at Sole Control Assurance Level 2 (SCAL2) according to EN 419241-1 [6] for qualified signatures. This means that the signature operation needs to be authorised, which is done using the Go>Sign Mobile app.

The TOE functionality includes but is not limited to: -

- Maintain the details of registered signers;
- Maintain the details of registered mobile devices;
- Generate authorisation requests;
- Verify the signed authorisation response XML (SAD);
- Generate the signing key pair in the CM; and
- Activate the signing key within the CM.

The Signer, using the Go>Sign Mobile app, communicates with the TOE via RAS Service to submit the SAD. The SAD binds together the Signer authentication with the signing key and the data to be signed, i.e. DTBS/R.

To avoid direct communication, the Go>Sign Mobile app securely communicate with the RAS Service (c.f. sec. 1.4.3), which then communicates with the TOE during both user registration as well remote signing operations.

The TOE can be used in High Availability by having multiple ADSS Server SAM appliances running in parallel behind a load-balancer. Replication of the cryptographic keys from one ADSS Server SAM appliance to the other is supported (including the replication or the keys managed by the embedded EN 419 221-5 compliant HSM).

1.4.2. TOE Type

The TOE is the Ascertia ADSS Server SAM.

The TOE is a software component, which implements the SAP. The TOE runs in a tamper protected environment that is connected to the CM via a trusted channel.

The TOE uses the SAD from the Signer to activate the corresponding signing key for use in a CM.

The TOE and the CM are together the Qualified Signature Creation Device (QSCD).

1.4.3. Non-TOE Hardware/Software/Firmware

The following hardware, firmware and software are supplied by the IT environment, and are therefore excluded from the TOE boundary: -

1. **ADSS RAS Service**

   The ADSS RAS Service provides

   - the required API interface towards the Business Application to: -
     1. Register the users, i.e. Signers;
     2. Send hash signing requests;
     3. Check the status of pending signing requests; and
     4. Retrieve the signed hash.
   - the required API interface towards the Go>Sign Mobile app to: -
     1. Allow user mobile registration (including authorisation key pair generation and certification)
     2. Allow Signer (user) to download authorisation request XML into the Go>Sign Mobile app;
     3. Send the signed authorisation response XML (SAD) from Go>Sign Mobile app to the TOE.

2. **Go>Sign Mobile app**

   The Go>Sign Mobile app is the Signer's Interaction Component (SIC). It is used locally by the Signer to communicate with the Ascertia ADSS Server SAM solution. The Go>Sign Mobile app authorises the remote signing actions by digitally signing authorisation responses using an authorisation key pair held securely in the mobile device's Secure Hardware Element.

3. **Database**

   The Ascertia ADSS Server SAM uses an RDBMS to store the details of: -

   1. Registered users;
   2. Registered mobile devices;
   3. Authorisation requests;
   4. Authorisation responses; and
   5. Signing transaction logs.

   The integrity of the data stored in any of the Ascertia ADSS Server SAM tables is protected by sequenced HMAC. The HMAC symmetric key is held in a CM.

4. **Cryptographic Module**

   The cryptographic key management and signing cryptographic operations occur in a CM which shall be certified according to EN 419 221-5 [7]. Note the CM is embedded inside the ADSS Server SAM appliance. The ADSS Server SAM authenticates and authorises each signing operation with the CM using a secure protocol which meets the requirements of the standard EN 419 221-5 [7].

5. **Business Application**

   The Business Application can be a server-side or desktop application to which a Signer interacts. The Signer (user) registration process is carried out by the Business Application on behalf of the user. The Business Application initiates different operations in the TOE, for example: -

   1. Create Signers (users);
   2. Create signing key pairs;
   3. Send the hash for signing;
   4. Check status of pending signing requests; and
   5. Retrieve signed hashes.

As explained earlier all Business Applications interactions with the TOE are conducted through the ADSS RAS Service. From the perspective of the TOE the ADSS RAS Service itself is the Business Application.

The ADSS RAS Service is registered as client using the ADSS Server SAM Admin Console. It is identified by a "Client ID" and authenticated using a TLS client certificate. The Business Application provides the document hash, i.e. DTBS/R, in the signing request to the ADSS RAS Service which passes it to the TOE for signature.

1.5. TOE Description

The ADSS Server SAM consists of the three components named:

1. ADSS Server SAM Service

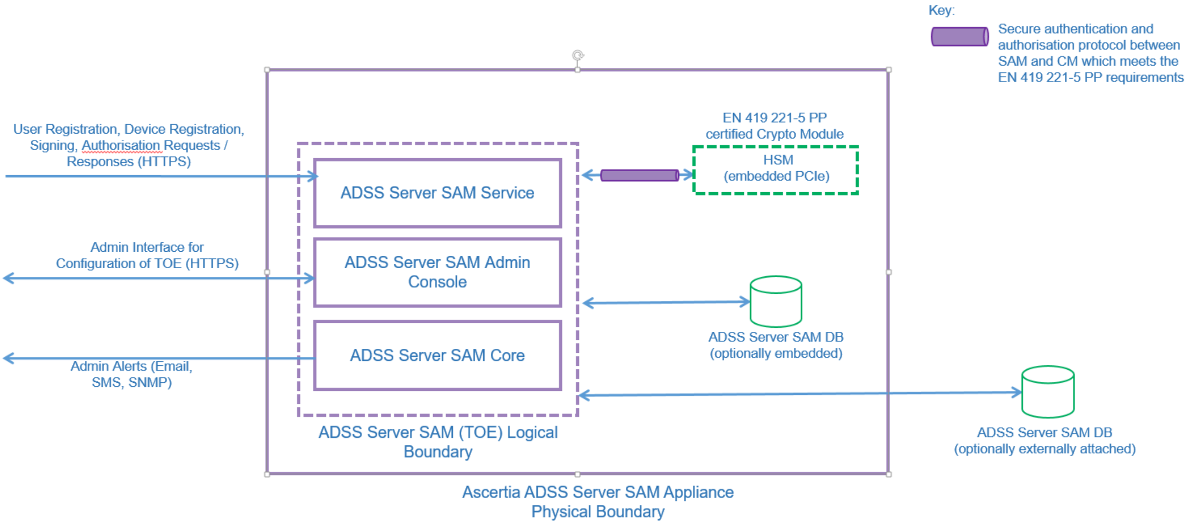
ADSS Server SAM Service provides different web services to perform various operations e.g. User Account Registration, user key enrollment, user mobile device registration, transaction/hash signing etc.

2. ADSS Server SAM Admin Console

ADSS Server SAM Admin Console allows administrators to configure the product, e.g. define access control, signer/user management, signer device management, configuring crypto source i.e. HSM etc.

3. ADSS Server SAM Core

ADSS Server SAM Core performs various background tasks e.g. Logs archiving, DB monitoring, HSM monitoring etc.


Key:
Secure authentication and authorisation protocol between SAM and CM which meets the EN 419 221-5 PP requirements
User Registration, Device Registration, Signing, Authorisation Requests / Responses (HTTPS)
Admin Interface for Configuration of TOE (HTTPS)
Admin Alerts (Email, SMS, SNMP)
ADSS Server SAM Service
ADSS Server SAM Admin Console
ADSS Server SAM Core
ADSS Server SAM (TOE) Logical Boundary
EN 419 221-5 PP certified Crypto Module
HSM (embedded PCIe)
ADSS Server SAM DB (optionally embedded)
ADSS Server SAM DB (optionally externally attached)
Ascertia ADSS Server SAM Appliance Physical Boundary

1.5.1. Physical Scope of the TOE

The physical boundary of the TOE is the tamper protected hardware fulfilling requirements of ISO/IEC 19790 [22] Security Level 3. The tamper protected hardware is also part of the TOE but not everything inside is part of the TOE. The operating system, application server, HSM, database etc. do not belong to the TOE. The TOE consists of the tamper protected hardware and the software components inside ADSS Server SAM logical boundary on the figure above.

The TOE can be configured using the ADSS Server SAM Admin Console. There are multiple possible configuration of ADSS Server SAM but only a specified configuration is certified which uses the Common Criteria certified-mode security settings and a certified CM. The configuration is described in AGD Guidance documents.

1.5.1.1. Delivery of the TOE

1. Ascertia will provide the Ascertia ADSS Server SAM software to its authorised SAM distributor
2. The software is encrypted using zip password-protection and uploaded to a secure FTP area hosted by Ascertia. Credentials for accessing the FTP area will be provided to the authorised SAM distributor securely using encrypted email. The email will also contain the cryptographic checksum of the uploaded software. The same email will also contain details of the customer to whom the ADSS Server SAM Appliance is to be shipped.
3. Ascertia will provide details of how to verify the cryptographic checksum to its SAM distributor
4. The SAM distributor verify the checksum on the ADSS Server SAM software after downloading it from the Ascertia FTP site
5. The SAM distributor will construct the ADSS Server SAM Appliance by installing the required components, i.e.:
    a. operating system,
    b. database,
    c. the approved EN 419 221-5 certified HSM (by contacting the HSM vendor and ensuring its securely delivery according to its defined procedures)
6. Then checksum-verified ADSS Server SAM software will be placed inside the appliance by the SAM distributor.
7. The SAM distributor will ensure that the SAM Appliance is properly sealed/protected and once done securely deliver the SAM appliance to the customer and inform Ascertia and the customer about the status (ETA)
8. The end-customer who receives the appliance will ensure the ADSS Server SAM appliance seals have not been tampered with.
9. The ADSS Server SAM software will then installed by the end-customer following the defined deployment documentation.

1.5.1.2. Roles & Available Functions

The TOE maintains the following roles: -

1) **Privileged Users.** There are two types of Privileged Users who can perform TOE specific operation through either the ADSS Server SAM Admin Console or the externally available ADSS Server SAM API:

    a. **Ascertia ADSS Server SAM Operators (Operators)**: They access the Ascertia ADSS Server SAM Admin Console to perform different TOE specific operations, e.g. configuring communication with the HSM, etc. These trusted Operators are created in ADSS Server SAM Admin Console and each operator is identified by an “Operator ID”.

    b. **Business Applications**: These access the TOE via APIs provided by RAS service to perform different TOE specific operations. On one hand they manage Signers (User Module) and the other, they act as Signature Creation Application (SCA). Business application are identified by their "Client ID".

2) **Unprivileged Users.**

   a. **Signers**: These are able to request remote signing operations by interacting with above Business Applications and then authorise these operations using the Ascertia Go>Sign mobile app to supply the required authorisation data

1.5.1.3. Authentication & Authorisation

The following authentication and authorisation processes occur: -

- **Operators:** They must logon to the Ascertia ADSS Server SAM using TLS client certificates before being allowed to perform any activity on the Ascertia ADSS Server SAM Admin Console. Operator TLS client certificates and associated private keys should be stored on a secure smart card/USB token thereby providing an extra layer of security for the private key plus two-factor authentication of the operator. The revocation status of Ascertia ADSS operator TLS certificates can also be checked at the time of logon by configuring this in Ascertia ADSS Server SAM Admin Console. However, it is recommended that operators' accounts are also immediately updated on Ascertia ADSS Server SAM at the time a certificate is revoked. The Ascertia ADSS Server SAM Admin Console ensures that access to system objects is strictly controlled. Users are first identified and authenticated as explained above, and once this process is complete and the user has successfully logged in, then access to system objects is controlled according to the user's role. Each role has a definition of which system objects it can access, and the type of access, e.g. read only, or edit/create/delete.
- **Business Applications:** They must be authenticated before accessing the Ascertia ADSS Server SAM APIs. The business applications must also authenticate using their respective TLS client certificate because all the communication is via mutually authenticated TLS channel. Note here the term Business Application refers to the ADSS RAS Service through which all business app interactions are conducted with the TOE.
- **Signer:** They are identified by the user ID and authenticated during device registration by two OTPs sent to the user's registered mobile number and email address. During the signing operation, Signers are identified via their user ID and authenticated by the signed authorisation response XML (SAD).

1.5.1.4. Cryptographic Support

The TOE does not perform cryptographic operations for its users (Signers): explicitly it does not generate/store/destroy, export/import, backup/restore, or use user key. The TOE invokes the CM with appropriate parameters whenever a cryptographic operation for the Signer is required, i.e. to authorise usage of the Assigned Key.

The TOE uses different infrastructure keys to protect its stored files and database records, and data transmitted or received via communication channels.

1.5.1.5. Audit

The TOE audits all security related events. All the audit records produced as a result of the TOE operator actions or the API requests from business applications are stored in Ascertia ADSS Server SAM database. The audit records do not include any data which allows the retrieval/decipherment of confidential data.

1.5.1.6. Trusted Communication

The TOE implements and enforces the following trusted communication methods and protocols: -

- **Operators:** Ascertia ADSS Server SAM Operators (Privileged Users) access the Ascertia ADSS Server SAM Admin Console GUI over a mutually authenticated TLS v1.2 channel.
- **ADSS RAS Service:** ADSS RAS Service communicates with TOE using mutually authenticated TLS v1.2 channel.
- **CM:** The TOE communicates with CM using vendor specific APIs. User passwords do not travel on this channel. This communication with the CM is enforced by the CM to be secure, authenticated and protected from replay attacks, i.e. the CM meets the requirements of EN 419 221-5 Protection Profile and is certified to Common Criterial EAL4+ level.
- **Go>Sign Mobile app:** The Go>Sign Mobile app (SIC) does not connect to the TOE but via the ADSS RAS Service. The Go>Sign mobile app connects to the ADSS RAS service over server-authenticated TLS v1.2 channel. The ADSS RAS Service then connects with the TOE as explained above.

The communication between the Signer and the TOE for authenticating and authorising the remote signature (using Go>Sign Mobile app) is conducted through the ADSS RAS Service as explained is based on the Signature Activation Protocol (SAP). The SAP is protected against replay, bypass and forgery attack, using a salt (random value to avoid replay attack), a validity period and the PKCS#1 authorization signature of the Signer. The SAP provides confidentiality for all sensitive transmitted data and integrity protection for all transmitted data, including the authentication and authorisation data and DTBS/R.

1.5.2. TOE Life Cycle

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

1. **Development:** The TOE developer develops the TOE application and its guidance documentation using any appropriate guidance documentation for components working within the TOE physical boundary, including the CM.
2. **Delivery:** The TOE is securely delivered from the TOE developer to the TSP.

3. **Installation and configuration:** The TSP installs and configures the TOE with the appropriate configuration and initialisation data.
4. **Operational phase:** In operation, the TOE can be used by Operators to create other Privileged Users (Operators or Business Applications) and by Business Applications to register Signers. Operators can maintain TOE configuration. Business Applications can manage Signers accounts and generate signature keys for a Signer. Only Business Applications can supply the data to be signed to the TOE and only Signers can authorise signature creation from their registered mobile devices using the Go>Sign Mobile app.

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

2. Conformance Claims

2.1 Common Criteria Conformance Claim

This ST claims conformance to: -

- Common Criteria for Information Technology Security Evaluation Part 1: Introduction and General Model; CCMB-2017-04-001, Version 3.1, Revision 5, April 2017 [1].
- Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Components; CCMB-2017-04-002, Version 3.1, Revision 5, April 2017 [2].
- Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements; CCMB-2017-04-003, Version 3.1, Revision 5, April 2017,[3].

as follows: -

Part 2 extended; and

Part 3 conformant.

The following must be considered: -

- Common Methodology for Information Technology Security Evaluation, Evaluation methodology; CCMB-2017-04-004, Version 3.1, Revision 5, April 2017, [4].

2.2 Package Conformance Claim

This ST claims conformance to: -

- EAL 4 assurance package augmented by AVA_VAN.5 defined in the CC Part 3 [3].

2.3 Protection Profile Conformance Claim

This ST claims strict conformance to: -

- EN 419 241-2: Trustworthy Systems Supporting Server Signing Part 2: Protection Profile (PP) for QSCD for Server Signing [5].

2.4 Protection Profile Conformance Rationale

2.4.1 Security Problem Definition

This ST claims strict conformance to the prEN 419241-2:2017 PP [5]. The parts of the TOE listed in this PP correspond to the ones listed in section 1.4.1 of this ST.

The security problem definition includes the assets, the subjects, the assumptions, the threats and the organizational security policies of the PP.

The following tables demonstrates that this ST contains all assumptions, threats and organisational security policies listed in prEN 419241-2:2017 PP [5].

Table 2-1 Source of Assumptions

Assumptions	419241-2:2017 PP [5]	Added by this ST
A.PRIVILEGED_USER	x	
A.SIGNER_ENROLMENT	x	
A.SIGNER_AUTHENTICATION_DATA_PROTECTION	x	
A.SIGNER_DEVICE	x	
A.CA	x	
A.ACCESS_PROTECTED	x	
A.AUTH_DATA	x	
A.TSP_AUDITED	x	
A.SEC_REQ	x	
A.CERTIFICATION_AUTHORITY		x

Table 2-2 Source of Threats

Threats	419241-2:2017 PP [5]	Added by this ST
ENROLMENT		
T.ENROLMENT_SIGNER_IMPERSONATION	x	
T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_ DISCLOSED	x	
T.SVD_FORGERY	x	
SIGNER MANAGEMENT		
T.ADMIN_IMPERSONATION	x	
T.MAINTENANCE_AUTHENTICATION_DISCLOSE	x	
USAGE		
T.AUTHENTICATION_SIGNER_IMPERSONATION	x	
T.SIGNER_AUTHENTICATION_DATA_MODIFIED	x	
T.SAP_BYPASS	x	
T.SAP_REPLAY	x	
T.SAD_FORGERY	x	
T.SIGNATURE_REQUEST_DISCLOSURE	x	
T.DTBSR_FORGERY	x	

Threats	419241-2:2017 PP [5]	Added by this ST
T.SIGNATURE_FORGERY	x	
SYSTEM		
T.PRIVILEGED_USER_INSERTION	x	
T.REFERENCE_PRIVILEGED_USER_AUTHENTICATION_DATA_MODIFICATION	x	
T.AUTHORISATION_DATA_UPDATE	x	
T. AUTHORISATION_DATA _DISCLOSE	x	
T.CONTEXT_ALTERATION	x	
T.AUDIT_ALTERATION	x	
T.RANDOM	x	

Table 2-3 Source of Organisational Security Policies

Organisational Security Policies	419241-2:2017 PP [5]	Added by this ST
OSP.RANDOM	x	
OSP.CRYPTO	x	

2.4.2 Security Objectives

The security objectives of prEN 419241-2:2017 PP [5] are included in this ST.

No additional security objectives were added by this ST.

Table 2-5 Source of Security objectives

	419241-2:2017 PP [5]	Added by this ST
Security Objectives for the TOE		
OT.SIGNER_PROTECTION	x	
OT.REFERENCE_SIGNER_AUTHENTICATION_DATA	x	
OT.SIGNER_KEY_PAIR_GENERATION	x	
OT.SVD	x	
OT.PRIVILEGED_USER_MANAGEMENT	x	
OT.PRIVILEGED_USER_AUTHENTICATION	x	
OT.PRIVILEGED_USER_PROTECTION	x	
OT.SIGNER_MANAGEMENT	x	
OT.SAD_VERIFICATION	x	
OT.SAP	x	
OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION	x	
OT.DTBSR_INTEGRITY	x	

	419241-2:2017 PP [5]	Added by this ST
OT.SIGNATURE_INTEGRITY	x	
OT.CRYPTO	x	
OT.RANDOM	x	
OT.SYSTEM_PROTECTION	x	
OT.AUDIT_PROTECTION	x	
Security Objectives for the operational environment		
OE.SVD_AUTHENTICITY	x	
OE.CA_REQUEST_CERTIFICATE	x	
OE.CERTIFICATE_VERFICATION	x	
OE.SIGNER_AUTHENTICATION_DATA	x	
OE.DELEGATED_AUTHENTICATION	x	
OE.DEVICE	x	
OE.ENV	x	
OE.CRYPTOMODULE_CERTIFIED	x	
OE.TW4S_CONFORMANT	x	

2.4.3 Security Functional Requirements

The functional requirements described in section 6 of this ST include all SFRs from prEN 419241-2:2017 PP [5].

Iterations and changes to the SFRs, with respect to prEN 419241-2:2017 PP [5], are listed in this table. These changes do not lower TOE security.

Table 2-6 Source of Security functional requirements

Security Functional Requirement	419241-2:2017 PP [5]	Changes by this ST
Security Audit		
FAU_GEN.1.	x	
FAU_GEN.2	x	
Cryptographic Support		
FCS_CKM.1	x	iterated to */RSA, */ECDSA, */AES
FCS_CKM.4	x	
FCS_COP.1	x	iterated to */DIG_SIG_GEN, */DIG_SIG_VER, */HASH, */HMAC, */ENC
FCS_RNG.1	x	

Security Functional Requirement	419241-2:2017 PP [5]	Changes by this ST
User Data Protection		
FDP_ACC.1/Privileged User Creation	x	added refinement
FDP_ACF.1/Privileged User Creation	x	added refinement
FDP_ACC.1/Signer Creation	x	added refinement
FDP_ACF.1/ Signer Creation	x	added refinement
FDP_ACC.1/Signer Key Pair Generation	x	added refinement
FDP_ACF.1/Signer Key Pair Generation	x	added refinement
FDP_ACC.1/Signer Maintenance	x	added refinement
FDP_ACF.1/Signer Maintenance	x	added refinement
FDP_ACC.1/Signer Key Pair Deletion	x	added refinement
FDP_ACF.1/Signer Key Pair Deletion	x	added refinement
FDP_ACC.1/Supply DTBS/R	x	added refinement
FDP_ACF.1/Supply DTBS/R	x	added refinement
FDP_ACC.1/Signing	x	
FDP_ACF.1/Signing	x	
FDP_ACC.1/TOE Maintenance	x	added refinement
FDP_ACF.1/TOE Maintenance	x	added refinement
FDP_ETC.2/Signer	x	
FDP_IFC.1/Signer	x	added refinement
FDP_IFF.1/Signer	x	added refinement
FDP_ETC.2/ Privileged User	x	
FDP_IFC.1/Privileged User	x	added refinement
FDP_IFF.1/Privileged User	x	added refinement
FDP_ITC.2/Signer	x	
FDP_ITC.2/ Privileged User	x	
FDP_UCT.1	x	
FDP_UIT.1	x	
Identification and Authentication		
FIA_AFL.1	x	iterated to */BA, */Signer
FIA_ATD.1	x	
FIA_UAU.1	x	added refinement
FIA_UAU.5/Signer	x	
FIA_UAU.5/Privileged User	x	added refinement
FIA_UID.2	x	
FIA_USB.1	x	added refinement
Security Management		

Security Functional Requirement	419241-2:2017 PP [5]	Changes by this ST
FMT_MSA.1/Signer	x	
FMT_MSA.1/Privileged User	x	
FMT_MSA.2	x	
FMT_MSA.3/Signer	x	added refinement
FMT_MSA.3/Privileged User	x	added refinement
FMT_MTD.1	x	added refinement
FMT_SMF.1	x	
FMT_SMR.2	x	added refinement
Protection of the TSF		
FPT_PHP.1	x	
FPT_PHP.3	x	
FPT_RPL.1	x	
FPT_STM.1	x	
FPT_TDC.1	x	
Trusted Path/Channels		
FTP_TRP.1/SSA	x	added refinement
FTP_TRP.1/SIC	x	added refinement
FTP_ITC.1/CM	x	

2.4.4 Security Assurance Requirements

The minimum package of security assurance requirement allowed for conformance to prEN 419241-2:2017 PP [5] is EAL 4 augmented with AVA_VAN.5.

This ST claims conformance to EAL 4 augmented by AVA_VAN.5. Therefore, the afore-described requirement is met and with respect to prEN 419241-2:2017 PP [5].

3. Security Problem Definition

3.1 Assets

The TOE has the following assets, which are to be protected in integrity and confidentiality as described below. The TOE must ensure that whenever an asset is persisted outside the TOE, the TOE has performed the necessary cryptographic operations to enforce confidentiality and detect if an asset has been modified. Access control to TOE assets outside the TOE are to be 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 R.Signer's sole control. The signing key can only be used by the CM. The TOE uses the asset R.Signing_Key_Id, which identifies a signing key in the CM. The binding of the R.Signing_Key_Id with R.Signer shall be protected in integrity.

Application Note 1 (Application Note 1 from [5], refined by the ST Author)

The integrity and confidentiality of the signing key and the link between the R.Signing_Key_Id and the signing key is the responsibility of the CM. The TOE ensures that only the R.Signer can use the signing key under their sole control. This is achieved by designing the TOE to conform to the requirements of Level 2 sole control described in [6].

R.Authorisation_Data: is data used by the TOE to activate a signing key in the CM. The signing key is identified by R.Signing_Key_Id. In ADSS Server SAM case R.Authorisation_Data is an authorisation key pair hold by the TOE.

It shall be protected in integrity and confidentiality.

Application Note 2 (Application Note 2 from [5], refined by the ST Author)

The R.Authorisation_Data is used by the CM to activate a signing key. The TOE holds an authorisation key pair which acts as the R.Authorisation_Data. To ensure the confidentiality, the authorisation private key is stored in encrypted form in the Ascertia ADSS Server SAM database. HMAC ensures the integrity of the database records.

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

The TOE uses a CM for signing key pair generation. As part of the signing key pair generation, CM provides the TOE with R.Signing_Key_Id and R.SVD. The TOE provides the R.SVD to the ADSS RAS Service (SSA) for further handling for the key pair to be certified. The integrity of the R.SVD is ensured through the use of a digital certificate.

R.DTBS/R: set of data which is transmitted to the TOE for digital signature creation on behalf of the R.Signer. The DTBS/R is transmitted to the TOE. The R.DTBS/R shall be protected in integrity. The transmission of the DTBS/R(s) to the TOE shall require the sending party - Signer or Privileged User - to be authenticated.

Application Note 3 (Application Note 3 from [5])

The confidentiality of the R.DTBS/R is not required by eIDAS Regulation No 910/2014 [8].

R.SAD: SAD is a set of data involved in the SAP, which activates the signature creation data to create a digital signature under the R.Signer's sole control. The R.SAD must combine: -

- The R.Signer's strong authentication as specified in EN 419 241-1 [6].
- If a particular key is not implied (e.g. a default or one-time key) a unique reference to R.Signing_Key_Id.
- A given R.DTBS/R.

The R.SAD shall be protected in integrity and confidentiality.

Application Note 4 (Application Note 4 from [5], refined by the ST Author)

Applied. The authorisation response XML is digitally signed by the Go>Sign Mobile app using an authorisation key held on the mobile device's secure element under the control of the Signer. The authorisation signature value is appended to the authorisation response XML under the <Signature> element. There is no confidential data contained in the authorisation response XML, hence the confidentiality requirement is not applicable. The confidentiality requirement is considered fulfilled.

Application Note 5 (Application Note 5 from [5])

The R.SAD may include some or all authentication factors or evidence from other systems that some or all authentication factors have been verified.

Application Note 6 (Application Note 6 from [5], refined by the ST Author)

The unique reference to R.Signing_Key_Id in the R.SAD is information obtained from the R.Signer's authentication.

R.Signing_Key_Id is <CertificateID> of the certificate assigned to the R.Signer.

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 R.Signing_Key_Id by the CM under the R.Signer's control as part of the SAP. The R.Signature shall be protected in integrity. The R.Signature can be verified outside TOE using R.SVD.

R.Audit: is an audit record that contains a log of events, which require audit. The logs are produced by the TOE and stored externally. The R.Audit shall be 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 shall be protected in integrity and confidentiality.

Application Note 7 (Application Note 7 from [5])

It is only within the TOE the R.Signer needs to be unique. It is not the responsibility of the TOE to establish 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.

Application Note 8 (Application Note 8 from [5])

The R.Signer can include references to zero, one or several R.Signing_Key_Ids and R.SVDs.

Application Note 9 (Application Note 9 from [5], refined by the ST Author)

The R.Signer does not require encrypted data then the confidentiality requirement is considered fulfilled.

R.Reference_Signer_Authentication_Data: is the set of data used by TOE to authenticate the R.Signer. It contains all the data (OTP device serial number, phone numbers, protocol settings, etc.) and keys (device, verification, etc.) used by the TOE to authenticate the R.Signer. This may include a SVD or certificate to verify an assertion provided as a result of delegated authentication. In ADSS Server SAM case the R.Reference_Signer_Authentication_Data is the mobile device id and the public key certificate which can verify the signature on SAD.

The R.Reference_Signer_Authentication_Data shall be protected in integrity and confidentiality.

Application Note 10 (Application Note 10 from [5])

The R.Reference_Signer_Authentication_Data is used by the TOE to authenticate the R.Signer, and the R.Authorisation_Data is used by the TOE to activate a signing key in the CM.

Application Note 11 (Application Note 11 from [5], refined by the ST Author)

The integrity of the records stored in Ascertia ADSS Server SAM database is protected by sequenced HMAC. The HMAC secret key is stored in the CM. There is no sensitive data which is stored in R.Signer’s record. Hence the confidentiality requirement is not applicable. The confidentiality requirement is considered fulfilled.

R.TSF_DATA: is the set of TOE configuration data used to operate the TOE. It shall be protected in integrity.

Application Note 12 (Application Note 12 from [5], refined by the ST Author)

The TOE configuration data includes but not limited to: -

- CM configuration;
- HMAC key to protect the integrity of the records stored in database; and
- TLS certificate configured to allowed secure access to privileged users.

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

R.Reference_Privileged_User_Authentication_Data: is the set of data used by the TOE to authenticate the Privileged User. It shall be protected in integrity and confidentiality. In case of ADSS Server SAM this is a TLS client certificate.

Application Note 13 (Application Note 13 from [5], refined by the ST Author)

The integrity of the records stored in Ascertia ADSS Server SAM database is protected by sequenced HMAC. The HMAC secret key is stored in the CM. There is no sensitive data which is stored in Privileged User’s record. Hence, the confidentiality requirement is not applicable. The confidentiality requirement is considered fulfilled.

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

3.2 Subjects

This following list of subjects interact with the TOE: -

- **Signer**: is the natural or legal person who uses the TOE through the SAP where they provide the SAD and can sign DTBS/R(s) using their signing key in the CM. They are able to perform signing operations (authorising their signing keys in the CM, transmitting the required data, including the unique user ID, two different authentication factors, the key ID, the key Authorisation Data and DTBS/R(s))
- **Privileged User**: which performs the administrative functions of the TOE and is able to provide a DTBS/R(s) to the TOE as part of the signature operation. There are two types of Privileged Users in Ascertia ADSS Server SAM: -
    - **Operators**: They access the Ascertia ADSS Server SAM Admin Console to perform different TOE specific operations, e.g. configure a CM, HMAC Key, etc. The Operators are defined in Ascertia ADSS Server SAM Admin Console and each operator is identified by an “Operator ID”.
    - **Business Applications**: They access the TOE via API provided by the ADSS RAS service to perform different TOE specific operations. On one hand, they manage Signers (User Module) and the other, act as Signature Creation Application (SCA). It’s important to note Business Applications do not directly interact with the TOE instead their requests must be channelled through the ADSS RAS Service. The TOE will only accept API request which originate from the ADSS RAS Service. From the perspective of the TOE the ADSS RAS Service is the Business Application.

Application Note 14 (Application Note 14 from [5], refined by the ST Author)

The list of subjects described in EN 419 241-1 [6] clause 6.2.1.2 SRG M.1.2 contains more roles as it covers the whole T4WS. The ST writer described the specific roles it implements and how these relate to authorisation rules in the SFRs. The TW4S maintains outside of the TOE the following roles: -

- **Privileged Roles**: -
    - **Security Officers**: have overall responsibility for administering the implementation of the security policies, practices and have access to security related information.
    - **System Auditors**: authorized to view archives and audit logs of the TW4S for the purposes of auditing the operations of the system in line with security policy.
- **Non-Privileged Roles**: -
    - **Registration Authority (RA):** authorized to send the public key certificate to the TW4S in response of a certificate signing request.

Application Note 15 (Application Note 15 from [5], refined by the ST Author)

The ADSS RAS Service (acts in the role of the Server Signing Application or SSA) plays a special role as it interacts directly with the TOE. Privileged Users can interact with the TOE directly or via the SSA. If the SSA as a service can perform administrative functions, e.g. creating a Signer, this is in this ST considered as a Privileged User. Operators interact directly with the TOE. Business Applications as Privileged Users interact with the TOE via ADSS RAS Service (SSA).

Application Note 16 (Application Note 16 from [5], refined by the ST Author)

The creation of Signers, management of reference signer authentication data and signing key generation is configured to be carried out together with a RA providing a registration service using the SSA, as specified in e.g. ETSI EN 319 411-1 [11].

3.3 Threats

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.

3.3.1 Enrolment

The threats during enrolment are: -

T.ENROLMENT_SIGNER_IMPERSONATION

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

- Transferring wrong R.Signer to TOE from RA; or
- 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 the 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 the 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 the Signer.

The threats on enrolment are threats on 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 Certification Authority. This results in loss of R.SVD integrity in the binding of R.SVD to the 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 [11] clause 6.3.3 d) then an attacker can forge signatures masquerading as the R.Signer.

Application Note 17 (Application Note 17 from [5], refined by the ST Author)

The keys use by the TOE are generated in the CM. According to the Application Note 17 from [5], if this is the case, this threat can be countered without any specific measures within the TOE. Applied. The remote signing key pair generated in CM and a Certificate Signing Request (CSR) is produced by the CM and delivered to the TOE for transmission to the TSP. The TOE passes it to the ADSS RAS Service (SSA) for transmission to the CA (TSP). It receives a certificate against it from an issuing CA. The CSR is self-signed with remote signing private key so the signature on CSR automatically serves as the Proof of Possession of private key. This threat can be countered without any specific measure within the TOE.

3.3.2 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 the 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.

3.3.3 Usage

This section describes threats for signature operation including authentication.

T.AUTHENTICATION_SIGNER_IMPERSONATION

An attacker impersonates the 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.

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 the 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.

Application Note 18 (from ST author)

This threat is mitigated. See Application Note 3 and 4.

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

Application Note 19 (Application Note 18 from [5])

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

3.3.4 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.

Application Note 20 (Application Note 19 from [5], refined by ST Author)

Applied. The TOE holds an authorisation key pair which acts as the R.Authorisation_Data. To ensure the confidentiality, the authorisation private key is stored in encrypted form in the Ascertia ADSS Server SAM database. HMAC ensures the integrity of the database records.

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 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 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.

The asset R.Random is threatened.

3.4 Relation Between Threats & Assets

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. (The table does not contain information where the confidentiality requirement is considered fulfilled)

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 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
	Confidentiality	T.AUTHENTICATION_SIGNER_IMPERSONATION T.SIGNATURE_REQUEST_DISCLOSURE T.DTBSR_FORGERY 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
R.Privileged_User	Integrity	T.PRIVILEGED_USER_INSERTION T.REFERENCE_PRIVILEGED_USER_AUTHENTICATION_DATA_MODIFICATION
R.Reference_Privileged_User_Authentication_Data	Integrity	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 3-1 Overview of the relationship between asset, associated security properties and threats

3.5 Organisational Security Policies

TOE shall comply 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 is required to generate random numbers that meet a specified quality metric. These random numbers shall be 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 shall only use algorithm, 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.

Application Note 21 (Application Note 20 from [5])

For cryptographic algorithms within the European Union this is as indicated in eIDAS [8] and an exemplary list of algorithms and parameters is given in ETSI TS 119 312 [9] or SOG-IS [10].

3.6 Assumptions

A.CERTIFICATION_AUTHORITY

It is assumed that the certificate for the R.SVD contains the R.SVD.

Application Note 22 (from ST author):

This assumption is not in the 419241-2:2017 PP [5]. Added this because the OE.CERTIFICATE_VERIFICATION from the 419241-2:2017 [5] has a useful purpose now.

A.PRIVILEGED_USER

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

A.SIGNER_ENROLMENT

The Signer shall be enrolled and certificates managed in conformance with the regulations given in eIDAS [8]. Guidance for how to implement an enrolment and certificate management system in conformance with [8] are given in e.g. EN 319 411-1[11] or for qualified certificate in e.g. EN 319 411-2 [12].

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 Signer qualified certificates is compliant with the relevant requirements for qualified TSP's as defined in eIDAS [8].

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 generated by the TOE are only handled by authorised personal in a physical secured environment. The personal that carries these activities should act under established practices.

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.

Application Note 23 (Application Note 21 from [5], refined by ST author)

All the data are stored outside the TOE but managed by the TOE. The HSM is secure since it is certified according to EN 419 221-5 [7]. Data stored in the database are secured with HMAC key which is also generated and kept in the HSM.

A.AUTH_DATA

It is assumed that the SAP is designed in such a way that the activation of the signing key is under 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 TOE is a qualified TSP according to article 3 (20) of eIDAS Regulation (EU) No 910/2014 [8] and audited to be compliant with the requirements for TSP's given by eIDAS [8].

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 [11].

4. Security Objectives

This section identifies and defines the security objectives for the TOE and its operational environment. These security objectives reflect the stated intent, counter the identified threats, and take into account the assumptions.

4.1 Security objectives for the TOE

The following security objectives describe security functions to be provided by the TOE.

4.1.1 Enrolment

OT.SIGNER_PROTECTION

The TOE shall ensure that data associated to R.Signer is protected in integrity and if needed in confidentiality.

OT.REFERENCE_SIGNER_AUTHENTICATION_DATA

The TOE shall be able to securely handle signer authentication data, R.Reference_Signer_ Authentication_Data, as part of R.Signer.

OT.SIGNER_KEY_PAIR_GENERATION

The TOE shall be able to securely use the CM to generate R.Signer signing key pairs and assign R.Signing_Key_Id and R.SVD to R.Signer.

OT.SVD

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

4.1.2 User Management

OT.PRIVILEGED_USER_MANAGEMENT

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

Application Note 24 (Application Note 22 from [5])

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 shall ensure that data associated to R.Privileged_User are protected integrity and if needed in confidentiality.

OT.SIGNER_MANAGEMENT

The TOE shall ensure 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.

4.1.3 Usage

OT.SAD_VERIFICATION

The TOE shall verify the SAD. That is, it shall check there is a link between the SAD elements and ensure the Signer is strongly authenticated.

Application Note 25 (Application Note 23 from [5], refined by the ST Author)

In this case, the TOE derives authorisation data from authentication data in the SAD and uses this to authorise the signing key in the CM. This function does not depend on the controls provided by the CM.

Application Note 26 (Application Note 24 from [5])

Requirements for authentication are described in EN 419 241-1 [11] SRA_SAP.1.1.

OT.SAP

The TOE shall implement the server-side endpoint of a 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; and
- Protection against replay, bypass of one or more steps and forgery.

Application Note 27 (Application Note 25 from [5], refined by the ST Author)

The Signer authentication is assumed to be conducted according to EN 419 241-1 [6] SCAL.2 for qualified signatures. This means Signer authentication can be carried out in the following way: -

- In the case of the current TOE it is done directly by the SAM.

OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION

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

OT.DTBSR_INTEGRITY

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

OT.SIGNATURE_INTEGRITY

The TOE shall ensure that a signature can’t be modified inside the TOE.

OT.CRYPTO

The TOE shall only use algorithm, 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.

4.1.4 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 sufficient entropy.

OT.SYSTEM_PROTECTION

The TOE shall ensure that modification of R.TSF_DATA is authorised by Privileged User and that unauthorised modification can be detected.

OT.AUDIT_PROTECTION

The TOE shall ensure that modifications to R.AUDIT can be detected.

4.2 Security Objectives for the Operational Environment

OE.SVD_AUTHENTICITY

The operational environment shall ensure the SVD integrity during transmission outside the TOE to the CA.

OE.CA_REQUEST_CERTIFICATE

The operational environment shall ensure that the qualified TSP that issues qualified certificates is compliant with the relevant requirements for qualified TSP’s as defined in eIDAS [8].

The operational environment shall use 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_VERFICATION

The operational environment shall verify 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 shall be 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 [11] SRA_SAP.1.1 are met.

In addition, the TSP shall ensure that: -

- The delegated party fulfils all the relevant requirements of this standard and the requirements for registration according to the eIDAS Regulation (EU) No 910/2014 [8]; 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 eIDAS Regulation (EU) No 910/2014 [8].

If the Signer is only authenticated using a delegated party, the TSP shall ensure that the secret key material used to authenticate the delegated party to the TOE shall reside in a certified CM consistent with the requirement as defined in EN 419 241-1 [11] SRG_KM.1.1.

OE.DEVICE

The device, computer/tablet/smart phone containing the SIC and which is used by the Signer to interact with the TOE shall be protected against malicious code. It shall participate using SIC as local part of the SAP and may calculate SAD as described in EN 419 241-1 [11]. It may be used to view the document to be signed.

OE.ENV

The TSP deploying the SSA and TOE shall be a qualified TSP according to article 3 (20) of eIDAS Regulation (EU) No 910/2014 [8] and audited to be compliant with the requirements for TSP's given by eIDAS [8]. The audit of the qualified TSP shall cover the security objectives for the operational environment specified in this clause.

The TOE shall operate in a protected environment that limits physical access to the TOE to authorised privileged users. The TOE software and hardware environment (including client applications) shall be installed and maintained by Privileged Users in a secure state that mitigates against the specific risks applicable to the deployment environment, including (where 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 CM defined in EN 419221-5 [7] and relies on said CM for providing a tamper-protected environment and for cryptographic functionality and random number generation.

Application Note 28 (Application Note 26 from [5], refined by ST author)

Applied.

OE.TW4S_CONFORMANT

The TOE shall be operated by a qualified TSP in an operating environment conformant with EN 419241-1 [6].

4.2.1 Security Problem Definition & Security Objectives

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

	Enrolment	OT.SIGNER_PROTECTION	OT.REFERENCE_SIGNER_AUTHE	OT.SIGNER_KEY_PAIR_GENERATION	OT.SVD
Enrolment					
T.ENROLMENT_SIGNER_IMPERSONATION		X	X		
T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_DISCLOSED		X	X		
T.SVD_FORGERY				X	X
Signer Management					
T.ADMIN_IMPERSONATION					
T.MAINTENANCE_AUTHENTICATION_DISCLOSE			X		
Usage					
T.AUTHENTICATION_SIGNER_IMPERSONATION					
T.SIGNER_AUTHENTICATION_DATA_MODIFIED			X		
T.SAP_BYPASS					
T.SAP_REPLAY					
T.SAD_FORGERY					
T.SIGNATURE_REQUEST_DISCLOSURE					
T.DTBSR_FORGERY					
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					
T.RANDOM					

Table 4-1 TOE Security objectives (Enrolment) and threats

	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
Enrolment									
T.ENROLMENT_SIGNER_IMPERSONATION					X				
T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_DISCLOSED									
T.SVD_FORGERY									
Signer Management									
T.ADMIN_IMPERSONATION			X		X				
T.MAINTENANCE_AUTHENTICATION_DISCLOSE									
Usage									
T.AUTHENTICATION_SIGNER_IMPERSONATION									
T.SIGNER_AUTHENTICATION_DATA_MODIFIED									
T.SAP_BYPASS									
T.SAP_REPLAY									
T.SAD_FORGERY									
T.SIGNATURE_REQUEST_DISCLOSURE									
T.DTBSR_FORGERY									
T.SIGNATURE_FORGERY									
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
T.RANDOM							X		

Table 4-2 TOE Security objectives (Signer Management and System) and threats

	Usage	OT.SAD_VERIFICATION	OT.SAP	OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION	OT.DTBSR_INTEGRITY	OT.SIGNATURE_INTEGRITY	OT.CRYPTO
Enrolment							
T.ENROLMENT_SIGNER_IMPERSONATION							
T.ENROLMENT_SIGNER_AUTHENTICATION_DATA_DISCLOSED							
T.SVD_FORGERY							X
Signer Management							
T.ADMIN_IMPERSONATION							
T.MAINTENANCE_AUTHENTICATION_DISCLOSE							
Usage							
T.AUTHENTICATION_SIGNER_IMPERSONATION		X					
T.SIGNER_AUTHENTICATION_DATA_MODIFIED			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							
T.REFERENCE_PRIVILEGED_USER_AUTHENTICATION_DATA_MODIFICATION							
T.AUTHORISATION_DATA_UPDATE							
T.AUTHORISATION_DATA_DISCLOSE							
T.CONTEXT_ALTERATION							
T.AUDIT_ALTERATION							
T.RANDOM							

Table 4-3 TOE Security objectives (Usage) and threats

	OT.SIGNER_PROTECTION	OT.REFERENCE_SIGNER_AUTHENTICATION_DATA	OT.SIGNER_KEY_PAIR_GENERATION	OT.SVD	OT.RANDOM	OT.CRYPTO
OSP.RANDOM					X	
OSP.CRYPTO						X

Table 4-4 TOE Security Objectives and Organizational Security Policies

	OE.SVD_AUTHENTICITY	OE.CA_REQUEST_CERTIFICATE	OE.CERTIFICATE_VERIFICATION	OE.SIGNER_AUTHENTICATION_DATA	OE.DELEGATED_AUTHENTICATION [1]	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.SIGNATURE_REQUEST_DISCLOSURE									
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									
T.RANDOM									

Table 4-5 Threats and Security Objectives for the environment

[1] No delegated party is involved, only the SAM verifies the Signer’s authentication factor(s). The TOE does not use delegated authentication. Because of this, this OE is not tracing to any security problem.

	OE.SVD_AUTHENTICITY	OE.CA_REQUEST_CERTIFICATE	OE.CERTIFICATE_VERIFICATION	OE.SIGNER_AUTHENTICATION_DATA	OE.DELEGATED_AUTHENTICATION [2]	OE.DEVICE	OE.ENV	OE.CRYPTOMODULE_CERTIFIED	OE.TW4S_CONFORMANT
Organisational Security Policies									
OSP.RANDOM									
OSP.CRYPTO								X	
Assumptions									
A.PRIVILEGED_USER									X
A.SIGNER_ENROLMENT							X		
A.SIGNER_AUTHENTICATION_DATA_PROTECTION				X					
A.SIGNER_DEVICE						X			
A.CA		X							
A.ACCESS_PROTECTED							X		
A.AUTH_DATA						X			
A.TSP_AUDITED							X		
A.SEC_REQ									X
A.CERTIFICATION_AUTHORITY			X						

Table 4-6 Organizational Security Policies and Security Objectives for the environment and Assumptions and Security Objectives for the environment

4.2.2 Rationale for the Security Objectives

This section provides a rationale objective that covers each threat, organizational security policy and assumption. Except the OE. DELEGATED_AUTHENTICATION, because the TOE does not use delegated authentication. Because of this, this OE is not tracing to any security problem.

4.2.2.1 Threats & Objectives

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 R.Signer to be securely created.
It is also covered by OT.REFERENCE_SIGNER_AUTHENTICATION_DATA requiring the TOE to be

[2] No delegated party is involved, only the SAM verifies the Signer's authentication factor(s). The TOE does not use delegated authentication. Because of this, this OE is not tracing to any security problem.

able to assign Signer authentication data to the R.Signer.
It is also covered by OE.TW4S_CONFORMANT as that requires signer enrolment to be handled in accordance with [14] for level at least substantial.

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.

T.SVD_FORGERY is covered by OT.SIGNER_KEY_PAIR_GENERATION requiring a CM to generate R.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 transmit from the TOE to the CA.
It is also covered by OE.CA_REQUEST_CERTIFICATE requiring the certification request to be protected in integrity.

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.

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

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

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.

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.

T.SAP_REPLAY is covered by OT.SAP requiring that the SAP must be able to resist whole or part of it being replayed.
It is also covered by OE.DEVICE requiring the SIC to participate the in SAP.

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

T.DTBSR_FORGERY is covered by OT.DTBSR_INTEGRITY requiring the R.DTBS/R to be protected in integrity during transmit to the TOE.
It is also covered by OE.DEVICE requiring the SIC to participate in the SAP.

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.

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.

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.

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

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

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

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

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

T.SAD_FORGERY is covered by OT.SAP requiring the TOE to be able to detect if the SAD has been modified during transmit to the TOE.
It is also covered by OT.SIGNATURE_AUTHENTICATION_DATA_PROTECTION requiring signature authentication datato be protected during transmit to the TOE.
It is also covered by OE.SIGNER_AUTHENTICATION_DATA requiring the Signer to protect their 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.

4.2.2.2 Organizational Security Policies & Objectives

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

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.

4.2.2.3 Assumptions & Objectives

A.PRIVILEGED_USER is covered by OE.TW4S_CONFORMANT which requires that the system where the TOE operates is compliant with EN 419 241-1 [6] where clause SRG M1.8 requires that administrators are trained.

A.SIGNER_ENROLMENT is covered by OE.ENV requiring the TSP to be audited.

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

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

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

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

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

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

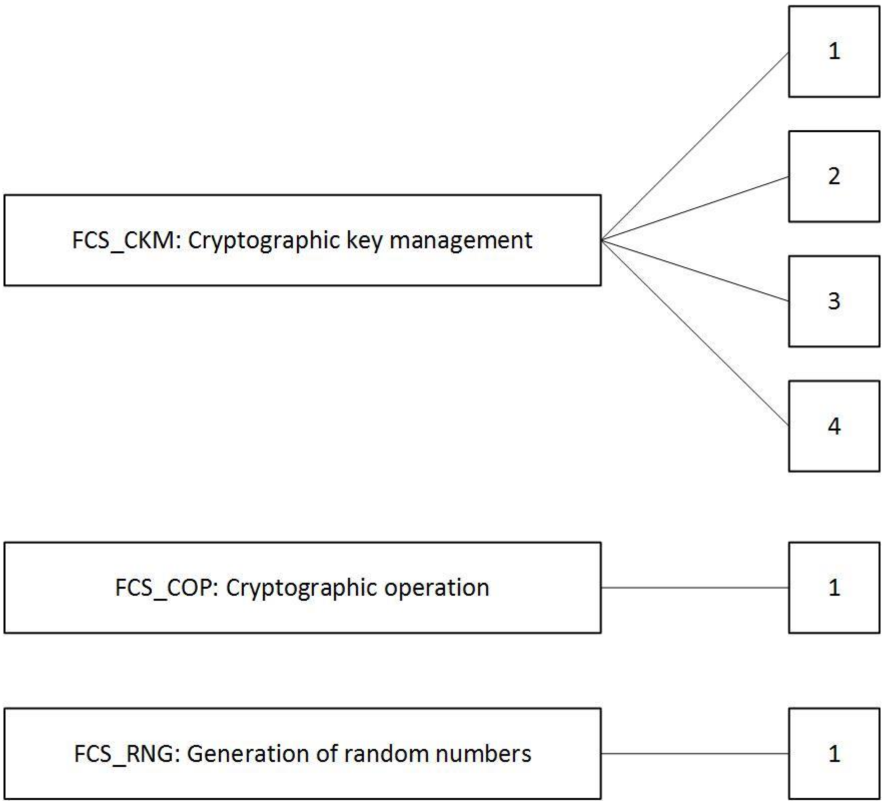
A.SEC_REQ is covered by OE.TW4S_CONFORMANT requiring the system where the TOE operates is compliant with EN 419 241-1 [6].

A.CERTIFICATION_AUTHORITY is covered by OE.CERTIFICATE_VERIFICATION.

5. Extended Components Definitions

5.1 Class FCS: Cryptographic Support

The Class FCS: Cryptographic Support as defined in CCPART2 [2] is extended with a new family: Generation of Random Numbers (FCS_RNG). The family is concerned with generation of random numbers. The following picture illustrates the decomposition of the Class FCS: Cryptographic Support with the added family FCS_RNG:


FCS_CKM: Cryptographic key management
1
2
3
4
FCS_COP: Cryptographic operation
1
FCS_RNG: Generation of random numbers
1

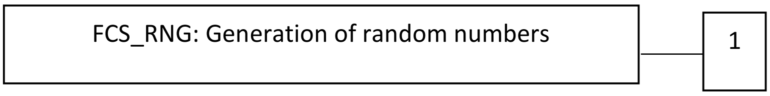
5.1.1 Generation of Random Numbers (FCS_RNG)

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 used 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 29 (Application Note 27 from [5])

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.

6. Security Requirements

6.1 Use of requirement specifications

Common Criteria allows several operations to be performed on functional requirements; refinement, selection, assignment, and iteration. Each of these operations is used in this ST and the underlying PP. The footnotes in this ST indicate the operations of the PP and the ST as well.

The **refinement** operation is used to add detail to a requirement, and thus further restricts a requirement. Refinement of security requirements is denoted by the word "refinement" in **bold text** and/or the added/changed words are in bold text. In cases where words from a CC requirement were deleted, a separate attachment indicates the words that were removed. If the refinement belongs to a selection or to an assignment it is denoted as **underlined bold**. In case of the selection or assignment is filled by the ST author it is denoted **double underlined bold**. In case of the refinement is the PP modification it is denoted as ***underlined italic bold***.

The **selection** operation is used to select one or more options provided by the CC in stating a requirement. Selections that have been made by the PP authors are denoted as underlined text and the original text of the component is given by a footnote. Selections to be filled in by the ST author appear in square brackets with an indication that a selection is to be made, [selection:], and are *italicized*. Selections filled in by the ST author are denoted as double underlined text and a foot note where the selection choices from the PP are listed.

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments that have been made by the PP authors are denoted by showing as underlined text and the original text of the component is given by a footnote. Assignments to be filled in by the ST author appear in square brackets with an indication that an assignment is to be made [assignment:], and are *italicized*. In some cases, the assignment made by the PP authors defines a selection to be performed by the ST author. Thus, this text is underlined and italicized like *this*. Assignments filled in by the ST author are denoted as double underlined text.

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

6.2 Subjects, Objects and Operations

This section describes the subjects, objects and operations support by the TOE.

Subject	Description
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

Table 6-1 Subjects

Object	Description
R.Reference_Privileged_User_Authentication_Data	Data used by the TOE to authenticate a Privileged_User
R.Reference_Signer_Authentication_Data	Data used by the TOE to authenticate a Signer
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 6-2 Objects

Subject	Operation	Object	Description
R.Privileged_User **(Operator)**	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.
R.Privileged_User **(Business Application)**	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 used to object authenticate the newly created signer.
R.Privileged_User **(Business Application)** **~~R.Signer~~**	Generate_Signer_Key_Pair	R.Signer R.SVD R.Signing_Key_Id	A key pair can be generated and assigned to a signer.
R.Privileged_User **(Business Application, Operator)**	Signer_Key_Pair_Deletion	R.Signer R.SVD R.Signing_Key_Id	A key pair can be deleted from a signer.

Subject	Operation	Object	Description
~~**R.Signer**~~			
R.Privileged_User **(Business Application, Operator)**	Signer_Maintenance	R.Signer R.Reference_Signer_Authentication_Data	Maintain the signer's security attributes.
R.Privileged_User **(Business Application)**	Supply_DTBS/R	R.Signer R.DTBS/R	Data to be signed by a signer can be supplied 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 **(Operator)**	TOE_Maintenance	R.TSF_DATA	The TOE configuration can be maintained by a privileged user.

Table 6-3 Subject, Objects and Operations

6.3 SFRs overview

This section gives an overview of how the SFRs are related to handle TOE usage scenarios and Signer object.

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 describes rules for creation, maintaining and usage of the R.Signer object as well as requirements to its values.

Authentication

- FIA_AFL.1/* limit 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 requires 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/* describe 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_Authentication_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.
- FPT_RPL.1 requires detection of replay of the R.SAD and reject signature operation in case of replay detected.
- FPT_STM1 is responsible for reliable time stamps for the signatures.

Privileged User object

- FIA_ATD.1 and FIA_USB.1 requires 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.
- FDP_IFC.1/Privileged user and FDP_IFF.1/Privileged User describes rules accessing any of Privileged User's data for Operator.

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 Maintenance

- FDP_ACC.1/TOE Maintenance using FDP_ACF.1/TOE Maintenance
- FMT_SMF.1, FMT_SMR.2 and FMT_MTD.1 requires the TOE to be able to carry out management functions and maintain users and roles.
- FPT_PHP.1 and FPT_PHP.3 requires the detection of any physical tampering or opening the case that compromises the TOE.

Audit

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

Communication

- FTP_TRP.1/SSA and FTP_TRP.1/SIC requires that either the Privileged User or the Signer initiates the communication.
- FCS_RNG.1 is required to generate random numbers for securing communication channels.
- FTP_ITC.1/CM requires trusted path for communication between SAM and the CM.

6.4 Security Functional Requirements

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

6.4.1 Security Audit (FAU)

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 [3] 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;[4]

k) none [5]

Application Note 30 (Application Note 28 from [5], refined by the ST Author).

Management of R.Privileged User and R.Signer objects include all events, which creates, modifies or deletes the R.Signer or R.Privileged User objects.

No delegated party is involved, only the SAM verifies the Signer's authentication factor(s).

Change of TOE configuration shall include all events, which creates, modifies and deletes the configuration object.

Application Note 31 (Application Note 29 from [5], refined by the ST Author)

Generation of a certification request is usage of the signing key and mandates an audit trail.

Application Note 32 (Application Note 30 from [5], refined by the ST Author)

Applied.

The audit logs will contain the DTBS/R with Base64 encoding and the entire database record will be HMAC protected (using HMACSHA256).

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information: -

[3] [selection: *minimum, basic, detailed, not specified*]

[4] [assignment: *other specifically defined auditable events.*]

[5] [assignment: *other specifically defined auditable events*]

a) Date and time of the event, type of event, subject identity (if applicable), 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,
- unique log ID,
- readable message or status about the operation. [6]

Application Note 33 (Application Note 31 from [5], refined by the ST Author)

Applied.

Audit trail does not include any data which allow to retrieve sensitive 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.

6.4.2 Cryptographic Support (FCS)

FCS_CKM.1/RSA Cryptographic key generation

Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction
FCS_CKM.1.1/ RSA	The TSF shall generate **RSA** cryptographic keys in accordance with a specified cryptographic key generation algorithm RSA [7] and specified cryptographic key sizes 2048, 3072, 4096, 8192 bits [8]

[6] [assignment: *Type of action performed (success or failure), identity of the role which performs the operation]*

[7] [assignment: *cryptographic key generation algorithm*]

[8] [assignment: *cryptographic key sizes*]

that meet the following: TS 119312 [9] PKCS#1 [15] FIPS 186-4 [16]. [9]

Application Note 34 (Application Note 32 from [5])

The TOE is expected to use a CM certified in conformance with EN 419 221-5 [7]. See also OE.CRYPTOMODULE_CERTIFIED for key generation. Although the TSF may not generate keys itself, these FCS SFRs express the requirement for the TSF to invoke the CM with the appropriate parameters whenever key generation is required.

Guidance on cryptographic algorithms can be found in ETSI TS 119 312 [9] or SOG-IS [10].

Application Note 35 (Application Note 33 from [5])

Applied.

The ST writer created iterations of this SFR.

FCS_CKM.1/ECDSA Cryptographic key generation

Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction
FCS_CKM.1.1/ ECDSA	The TSF shall generate **ECDSA** cryptographic keys in accordance with a specified cryptographic key generation algorithm ECDSA [10] and specified cryptographic key sizes 192, 224, 256, 384, 521 bits [11] that meet the following: TS 119312 [9] PKCS#1 [15] FIPS 186-4 [16]. [12]

FCS_CKM.1/AES Cryptographic key generation

Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction
FCS_CKM.1.1/ AES	The TSF shall generate **AES** cryptographic keys in accordance with a specified cryptographic key generation algorithm AES [13] and specified cryptographic key sizes 256, 512 bits [14] that meet the following: SOG-IS [10]. [15]

[9] [assignment: *list of standards*]
[10] [assignment: *cryptographic key generation algorithm*]
[11] [assignment: *cryptographic key sizes*]
[12] [assignment: *list of standards*]
[13] [assignment: *cryptographic key generation algorithm*]
[14] [assignment: *cryptographic key sizes*]
[15] [assignment: *list of standards*]

FCS_CKM.4 Cryptographic key destruction

Hierarchical to: No other components.

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

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method zeroise [16] that meets the following: FIPS 140-2 [17] [17].

Application Note 36 (Application Note 34 from [5], refined by ST author)

Applied.

The TOE is expected to use a CM certified in conformance with EN 419 221-5 [7] for key destruction.

Although the TSF may not destruct keys, this SFR expresses the requirement for the TSF to invoke the CM with the appropriate parameters whenever key destruction is required.

Zeroisation of the keys is the common method of the TOE.

Application Note 37 (Application Note 35 from [5], refined by ST author)

Applied.

Zeroisation of the keys is the common method of the TOE.

FCS_COP.1/DIG_SIG_GEN Cryptographic operation

Hierarchical to: No other components.

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

FCS_COP.1.1/DIG_SIG_GEN The TSF shall perform digital signature – generation [18] in accordance with a specified cryptographic algorithm Table 6-4: Signature generation algorithm [19] and cryptographic key sizes Table 6-4: Key sizes [20] that meet the following: Table 6-4: Applicable standards[21].

[16] [assignment: *cryptographic key destruction method*]
[17] [assignment: *list of standards*]
[18] [assignment: *list of cryptographic operations*]
[19] [assignment: *cryptographic algorithm*]
[20] [assignment: *cryptographic key sizes*]
[21] [assignment: *list of standards*]

Signature generation algorithm	Key sizes	Padding / Short curve name	Hash algorithm	Applicable standards
RSA	2048, 3072, 4096, 8192 bits	RSASSA-PKCS-v1.5 RSASSA-PSS	SHA-224, SHA256, SHA-384, SHA512	TS 119312 [9] RSASSA-PKCS#1v1_5 PKCS#1 [15]
ECDSA	192, 224, 256, 384, 521 bits	NIST P-256, P-384, P-521	SHA-224, SHA256, SHA-384, SHA512	FIPS 186-4 [16], ANSI X9.62 [18]

Table 6-4: Signature generation algorithm

Application Note 38 (Application Note 36 from [5])

The TOE is expected to use a CM certified in conformance with EN 419 221-5 [7] for cryptographic operations.

Application Note 39 (Application Note 37 from [5])

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 eIDAS Regulation No 910/2014 [8] and a list of approved signature and seal formats are given in [13].

Application Note 40 (from ST Author)

Ascertia ADSS Server SAM allows configurations to restrict the use of weaker hash algorithms. The weaker algorithms are supported just for backward compatibility and legacy applications.

The Ascertia ADSS Server SAM configuration manual explains how to configure the product to use the recommended algorithms.

If the TOE uses aforementioned weaker algorithms it is outside of the scope of this ST.

FCS_COP.1/DIG_SIG_VER Cryptographic operation

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/ DIG_SIG_VER

The TSF shall perform digital signature – verification [22] in accordance with a specified cryptographic algorithm Table 6-5: Signature verification algorithm [23] and cryptographic key sizes Table 6-5: Key sizes [24] that meet the following: Table 6-5: Applicable standards [25].

Signature verification algorithm	Key sizes	Padding / Short curve name	Hash algorithm	Applicable standards
RSA	2048, 3072, 4096, 8192 bits	RSASSA-PKCS-v1.5 RSASSA-PSS	SHA224, SHA256, SHA384, SHA512	TS 119312 [9] RSASSA-PKCS#1v1_5 PKCS#1 [15]
ECDSA	192, 224, 256, 384, 521 bits	NIST P-256, P-384, P-521	SHA224, SHA256, SHA384, SHA512	FIPS 186-4 [16], ANSI X9.62 [18]

Table 6-5: Signature verification algorithm

FCS_COP.1/HASH Cryptographic operation

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/ HASH

The TSF shall perform cryptographic hash function [26] in accordance with a specified cryptographic algorithm SHA256, SHA384, SHA512 [27] and cryptographic key sizes none [28] that meet the following: TS 119312 [9], FIPS 186-4 [16] [29].

FCS_COP.1/HMAC Cryptographic operation

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

[22] [assignment: *list of cryptographic operations*]
[23] [assignment: *cryptographic algorithm*]
[24] [assignment: *cryptographic key sizes*]
[25] [assignment: *list of standards*]
[26] [assignment: *list of cryptographic operations*]
[27] [assignment: *cryptographic algorithm*]
[28] [assignment: *cryptographic key sizes*]
[29] [assignment: *list of standards*]

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/ HMAC	The TSF shall perform keyed-hash message authentication code [30] in accordance with a specified cryptographic algorithm HMAC-SHA256 [31] and cryptographic key sizes: 256 bit [32] that meet the following: RFC 2104 [19] [33].

FCS_COP.1/ENC Cryptographic operation

Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1/ ENC	The TSF shall perform key encryption [34] in accordance with a specified cryptographic algorithm AES [35] and cryptographic key sizes 256, 512 bits [36] that meet the following: SOG-IS [10] [37].

The next SFR is relevant when the TOE is deployed in an appliance distinct form the CM.

FCS_RNG.1 Generation of random numbers

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FCS_RNG.1.1	The TSF shall provide a deterministic [38] random number generator that implements: securing communication with CM [39].
FCS_RNG.1.2	The TSF shall provide bits [40] that meet BSI AIS 20 v2.0 [20] or NIST 800-90A [21] [41].

Application Note 41 (Application Note 40 from [5])

[30] [assignment: *list of cryptographic operations*]
[31] [assignment: *cryptographic algorithm*]
[32] [assignment: *cryptographic key sizes*]
[33] [assignment: *list of standards*]
[34] [assignment: *list of cryptographic operations*]
[35] [assignment: *cryptographic algorithm*]
[36] [assignment: *cryptographic key sizes*]
[37] [assignment: *list of standards*]
[38] [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*
[39] [assignment: *list of security capabilities*]
[40] [selection: *bits, octets of bits, numbers [assignment: format of the numbers]*]
[41] [assignment: *a defined quality metric*]

For more information on the selections and assignments, see the SFR definition in section 5.

Application Note 42 (Application Note 41 from [5], refined by ST author)

This SFR is not relevant as the TOE is deployed in the same appliance as the CM.**Application Note 43 (from ST Author)**

The algorithm to be used can be configured by Ascertia ADSS Server SAM operator. The Ascertia ADSS Server SAM operator can choose an RNG algorithm from the following options: -

- HMAC/SHA-256 MAC-based secure random according to NIST SP800-90A; or
- SHA-256 hash-based secure random according to BSI AIS 20 v2.0.

6.4.3 User Data Protection (FDP)

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 <u>*Privileged User Creation SFP*</u>[42] on: - <u>*Subjects: Privileged User*</u> <u>**(Operator)**</u> <u>*Objects: New security attributes for the Privileged User to be created.*</u> <u>*Operations: Create New Privileged User:*</u> <u>*The TOE creates R.Privileged User and R.Reference Privileged User Authentication Data with information transmitted by Privileged User*</u>.[43]

Application Note 44 (Application Note 42 from [5], refined by ST author)

Applied.

When Ascertia ADSS Server SAM is installed, a default Ascertia ADSS Server SAM Operator account is automatically created. The default operator logins to the Ascertia ADSS Server SAM Admin Console and creates Privileged Users (Operators and Business Applications, see sec. 3.2).

Application Note 45 (from ST Author)

The ST author added refinement (in this and the following SFRs) about which type of Privileged User (Operator, Business Application) is the relevant.

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

Hierarchical to: No other components.

[42] [assignment: *access control SFP*]
[43] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1/ Privileged User Creation	The TSF shall enforce the *Privileged User Creation SFP* [44] to objects based on the following: - 1) *whether the subject is a Privileged User* **(Operator)** *authorized to create a new Privileged User.* [45]
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* **(Operator)** *who has been authorised for creation of new users can carry out the Create New Privileged User operation.* [46]
FDP_ACF.1.3/ Privileged User Creation	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *None.* [47]
FDP_ACF.1.4/ Privileged User Creation	The TSF shall explicitly deny access of subjects to objects based on the following additional rule: *None.* [48]

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/ Signer Creation	The TSF shall enforce the *Signer Creation SFP* [49] on: - *Subjects: Privileged User* **(Business Application)** *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* **(Business Application)** [50]

FDP_ACF.1/Signer Creation	Security attribute based access control
Hierarchical to:	No other components.

[44] [assignment: *access control SFP*]
[45] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]
[46] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[47] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]
[48] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]
[49] [assignment: *access control SFP*]
[50] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

Dependencies: FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1/ Signer Creation

The TSF shall enforce the *Signer Creation SFP* [51] to objects based on the following: -

1) *whether the subject is a Privileged User* **(Business Application)** *authorized to create a new Signer.* [52]

FDP_ACF.1.2/ Signer 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* **(Business Application)** *who has been authorised for creation of new users can carry out the Create_New_Signer operation.* [53]

FDP_ACF.1.3/ Signer Creation

The TSF shall explicitly authorise access of subjects to objects based on the following additional rules*: None.* [54]

FDP_ACF.1.4/ Signer Creation

The TSF shall explicitly deny access of subjects to objects based on the following additional rule: *None.* [55]

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* [56] *on:* -

Subjects: Privileged User **(Business Application, Operator)** ~~**and Signer**~~

Objects: The security attributes R.Reference_Signer_Authentication_Data of R.Signer

Operations: Signer_Maintenance:

The Privileged User **(Business Application, Operator)** ~~**or Signer**~~ *instructs the TOE to update R.Reference_Signer_Authentication_Data of R.Signer.*[57]

[51] [assignment: *access control SFP*]

[52] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

[53] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]

[54] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]

[55] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]

[56] [assignment: *access control SFP*]

[57] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

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 initialisation

FDP_ACF.1.1/ Signer Maintenance The TSF shall enforce the *Signer Maintenance SFP* [58] to objects based on the following:

1) *Whether the subject is a Privileged User* **(Business Application, Operator)** ~~*or Signer*~~ *authorised to maintain the Signer security attributes*. [59]

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* **(Business Application, Operator)** ~~*or Signer*~~ *who has been authorised to maintain a Signer can carry out the Signer Maintenance operation*. [60]

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.*[61]

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.*[62]

Application Note 46 (Application Note 43 from [5], refined by ST author)

R.Reference_Signer_Authentication Data can be maintained by only the Privileged User (Business Application and Operator).

FDP_ACC.1/Signer Key Pair Generation Subset access control

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

[58] [assignment: *access control SFP*]
[59] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]
[60] [assignment*: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[61] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]
[62] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]

FDP_ACC.1.1/ Signer Key Pair Generation	The TSF shall enforce the *Signer Key Pair Generation SFP* [63] on: - *Subjects: Privileged User* **(Business Application)** ~~**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* **(Business Application)** ~~**or Signer**~~ *instruct 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.* [64]

Application Note 47 (Application Note 44 from [5], refined by ST author)

Applied.

See Application Note 2 in sec. 3.1 Assets.

Application Note 48 (Application Note 45 from [5], refined by ST author)

Applied.

See R.Authorisation_Data. The user's signing keys can be backed-up from one CM and imported to another CM.

Application Note 49 (Application Note 46 from [5], refined by ST author)

Applied.

The TOE does not use pre-generated keys.

Application Note 50 (Application Note 47 from [5], refined by ST author)

Applied.

The R.SVD is added to the CSR to get the signing certificate from an issuing CA.

Application Note 51 (from ST Author)

The key pair generation request can be immediately sent after the user details are registered or it can be initiated later depending on business application implementation. Therefore, the Signer creation process is two-step process in the TOE and these are covered by FDP_ACC.1/Signer_Creation and FDP_ACC.1/Signer Key Pair Generation.

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 initialisation

[63] [assignment: *access control SFP*]
[64] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

FDP_ACF.1.1/ Signer Key Pair Generation

The TSF shall enforce the *Signer Key Pair Generation SFP* [65] to objects based on the following: -

1) *whether the subject is a Privileged User* **(Business Application)** ~~*or Signer*~~ *authorised to generate a key pair.* [66]

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* **(Business Application)** ~~*or Signer*~~ *who has been authorised to generate the key pair can carry out the Generate Signer Key Pair operation*. [67]

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*. [68]

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*. [69]

Application Note 52 (Application Note 48 from [5], refined by ST author)

Applied.

The TOE does not use pre-generated keys.

Application Note 53 (Application Note 49 from [5])

Owning an 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* [70] *on: - Subjects: Privileged User* **(Business Application, Operator)** ~~**and Signer**~~

[65] [assignment: *access control SFP*]
[66] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]
[67] [assignment*: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[68] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]
[69] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]
[70] [assignment: *access control SFP*]

Objects: The security attributes R.Signing_Key_Id and R.SVD of R.Signer
Operations: Signer_Key_Pair_Deletion:
The Privileged User **(Business Application, Operator)** ~~*or Signer*~~ *instructs the TOE to delete the R.Signing_Key_Id and R.SVD of R.Signer.*[71]

Application Note 54 (Application Note 50 from [5], refined by ST author)

Applied. Not required.

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.

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 initialisation

FDP_ACF.1.1/ Signer Key Pair Deletion

The TSF shall enforce the *Signer Key Pair Deletion SFP* [72] to objects based on the following: -

1) *Whether the subject is a Privileged User* **(Business Application, Operator)** ~~*or Signer*~~ *authorised to delete the Signer security attributes.* [73]

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* **(Business Application, Operator)** ~~*or Signer*~~ *who has been authorised to delete a key pair can carry out the Signer_Key_Pair_Deletion operation.* [74]

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.* [75]

[71] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]
[72] [assignment: *access control SFP*]
[73] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]
[74] [assignment*: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[75] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]

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, the key pair can't be deleted*. [76]

The DTBS/R can be supplied to the TOE either by the Signer as part of the SAP, which is covered by the FDP_ACC.1/Signing or by a Privileged User prior the signature operation. The following SFR handles the case where the Privileged User supplies the DTBS/R.

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* [77] on: -
Subjects: Privileged User **(Business Application)**
Objects: The security attributes R.DTBS/R of R.Signer.
Operations: Supply DTBS/R:
The Privileged User **(Business Application)** *instructs the TOE to link the supplied DTBS/R(s) to the next signature operation for R.Signer.*[78]

Application Note 55 (Application Note 51 from [5], refined by ST author)

Applied. The TOE provides facilities to supply the DTBS/R.

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 initialisation

FDP_ACF.1.1/ Supply DTBS/R

The TSF shall enforce the *Supply DTBS/R SFP* [79] to objects based on the following: -

1) *Whether the subject is a Privileged User* **(Business Application)** *authorised to supply a DTBS/R(s)*. [80]

[76] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]
[77] [assignment: *access control SFP*]
[78] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]
[79] [assignment: *access control SFP*]
[80] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

FDP_ACF.1.2/ Supply DTBS/R

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* **(Business Application)** *who has been authorised to supply a DTBS/R(s) can carry out the Supply DTBS/R operation*. [81]

FDP_ACF.1.3/ Supply DTBS/R

The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *None.* [82]

FDP_ACF.1.4/ Supply DTBS/R

The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *None.* [83]

Application Note 56 (Application Note 52 from [5], refined by ST author)

Applied. The TOE provides facilities to supply the DTBS/R.

FDP_ACC.1/Signing	Subset access control

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/ Signing

The TSF shall enforce the *Signing SFP* [84] 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.* [85]

Application Note 57 (Application Note 53 from [5], refined by ST author)

Applied. See Application Note 2 in sec. 3.1 Assets.

[81] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[82] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]
[83] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]
[84] [assignment: *access control SFP*]
[85] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

Application Note 58 (Application Note 54 from [5], refined by ST author)

Applied.

The Business Application (SCA) provides the document hash, i.e. DTBS/R, to RAS Service (SSA) which passes it to the TOE for signature computation.

Application Note 59 (Application Note 55 from [5])

Signing key deactivation means that the Signer shall authorise any subsequent use of it.

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 initialisation

FDP_ACF.1.1/ Signing The TSF shall enforce the *Signing SFP* [86] to objects based on the following: -

1) *Whether the subject is a Signer authorised to create a signature*. [87]

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 an 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.* [88]

FDP_ACF.1.3/ Signing The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: -

[86] [assignment: *access control SFP*]
[87] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]
[88] [assignment*: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]

1) *The Signer must be the owner of the R.Signer object used to generate the signature.* [89]

FDP_ACF.1.4/ Signing	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 used to create a signature.* [90]

Application Note 60 (Application Note 56 from [5], refined by ST author)

Applied. The R.Signing_Key_Id implied.

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/ Maintenance	The TSF shall enforce the *TOE Maintenance SFP* [91] on: *Subjects: Privileged User* **(Operator)** *Objects: R.TSF_DATA.* *Operations: TOE_Maintenance:* *The Privileged User* **(Operator)** *transmits information to the TOE to manage R.TSF_DATA.*[92]

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 initialisation
FDP_ACF.1.1/ TOE Maintenance	The TSF shall enforce the *TOE Maintenance SFP* [93] to objects based on the following: 1) *Whether the subject is a Privileged User* **(Operator)** *authorised to maintain the TOE configuration data.* [94]
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: -

[89] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]
[90] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]
[91] [assignment: *access control SFP*]
[92] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]
[93] [assignment: *access control SFP*]
[94] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

	1) *Only a Privileged User* **(Operator)** *who has been authorised to maintain the TOE can carry out the TOE Maintenance operation*. [95]
FDP_ACF.1.3/ TOE Maintenance	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *None.* [96]
FDP_ACF.1.4/ TOE Maintenance	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *None.* [97]

The TOE can store data in an external repository to meet requirements on, e.g. capacity and redundancy.

FDP_ETC.2/Signer	Export of user data with security attributes
Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or 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* [98] 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 TSF, 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 TSF: *None.* [99]

Application Note 61 (Application note 57 from [5], refined by ST author)

The TOE does not export user data then this SFR is trivially satisfied.

FDP_IFC.1/Signer	Subset information flow control

[95] [assignment*: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[96] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]
[97] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]
[98] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]
[99] [assignment: *additional exportation control rules*]

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* [100] on *Privileged User* **(Business Application and Operator)** ~~**and Signer**~~ *accessing Signer security attributes for all operations.* [101]

FDP_IFF.1/Signer Simple security attributes

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control, or

FMT_MSA.3 Static attribute initialisation

FDP_IFF.1.1/ Signer The TSF shall enforce the *Signer Flow SFP* [102] based on the following types of subject and information security attributes: -

Privileged User **(Business Application and Operator)** ~~**and Signer**~~ *accessing the Signer security attributes.* [103]

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. [104]

FDP_IFF.1.3/ Signer The TSF shall enforce the: *None.* [105]

FDP_IFF.1.4/ Signer The TSF shall explicitly authorise an information flow based on the following rules: *None.* [106]

[100] [assignment: *information flow control SFP*]
[101] [assignment: *list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP*]
[102] [assignment: *information flow control SFP*]
[103] [assignment: *list of subjects and information controlled under the indicated SFP, and for each, the security attributes*]
[104] [assignment: *for each operation, the security attribute-based relationship that must hold between subject and information security attributes*]
[105] [assignment: *additional information flow control SFP rules*]
[106] [assignment: *rules, based on security attributes, that explicitly authorise information flows*]

FDP_IFF.1.5/ Signer	The TSF shall explicitly deny an information flow based on the following rules: *None.* [107]

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

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_ETC.2.1/ Privileged User	The TSF shall enforce the *Privileged User Creation SFP* [108] when exporting user data, controlled under the SFP(s), outside of the TSF.
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 TSF, 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 TSF: *None.* [109]

Application Note 62 (Application Note 58 from [5], refined by ST author)

The ST writer shall describe which user data that can be exported from the TOE. The following Privileged User data can be exported from the TOE:

- Data of Operator: Id, FirstName, LastName, RoleName, EmailAddress, MobileNo, CertificateRequest, Certificate, CertificateChain, SubjectName, Status, CreatedAt, CreatedBy, LastUpdatedAt, LastUpdatedBy.
- Data of Business Application: Id, FriendlyName, Address, PhoneNo, EmailAddress, Status, AssignedSamProfiles, Certificate, ValidFrom, ValidTo

The Privileged User data can be exported in an XML format just like TOE configuration data. This XML later can be imported to another instance of the TOE.

FDP_IFC.1/Privileged user	Subset information flow control

Hierarchical to:	No other components.
Dependencies:	FDP_IFF.1 Simple security attributes

[107] [assignment: *rules, based on security attributes, that explicitly deny information flows*]
[108] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]
[109] [assignment: *additional exportation control rules*]

FDP_IFC.1.1/ Privileged User — The TSF shall enforce the *Privileged User Flow SFP* [110] on *Privileged User* **(Operator)** *accessing Privileged User security attributes for all operations.* [111]

FDP_IFF.1/Privileged User	Simple security attributes

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control, or

FMT_MSA.3 Static attribute initialisation

FDP_IFF.1.1/ Privileged User — The TSF shall enforce the *Privileged User Flow SFP* [112] based on the following types of subject and information security attributes: -

Privileged User **(Operator)** *accessing the Privileged User security attributes.* [113]

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. [114]

FDP_IFF.1.3/ Privileged User — The TSF shall enforce the: *None.* [115]

FDP_IFF.1.4/ Privileged User — The TSF shall explicitly authorise an information flow based on the following rules: *None.* [116]

FDP_IFF.1.5/ Privileged User — The TSF shall explicitly deny an information flow based on the following rules: *None.* [117]

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

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or

FDP_IFC.1 Subset information flow control]

[110] [assignment: *information flow control SFP*]

[111] [assignment: *list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP*]

[112] [assignment: *information flow control SFP*]

[113] [assignment: *list of subjects and information controlled under the indicated SFP, and for each, the security attributes*]

[114] [assignment: *for each operation, the security attribute-based relationship that must hold between subject and information security attributes*]

[115] [assignment: *additional information flow control SFP rules*]

[116] [assignment: *rules, based on security attributes, that explicitly authorise information flows*]

[117] [assignment: *rules, based on security attributes, that explicitly deny information flows*]

[FTP_ITC.1 Inter-TSF Trusted channel, or

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* [118] 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.* [119]

Application Note 63 (Application Note 59 from [5], refined by ST author)

The TOE does not import user data then this SFR is trivially satisfied.

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

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or

FDP_IFC.1 Subset information flow control]

[FTP_ITC.1 Inter-TSF Trusted channel, or

FTP_TRP.1 Trusted path]

FPT_TDC.1 Inter-TSF basic TSF data consistency

FDP_ITC.2.1/ Privileged User — The TSF shall enforce the *Privileged User Creation SFP* [120] 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.

[118] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]
[119] [assignment: *additional importation control rules*]
[120] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]

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.* [121]

Application Note 64 (Application Note 60 from [5], refined by ST author)

The ST writer shall describe which user data that can be imported to the TOE. The following Privileged User data can be imported to the TOE:

- Data of Operator: Id, FirstName, LastName, RoleName, EmailAddress, MobileNo, CertificateRequest, Certificate, CertificateChain, SubjectName, Status, CreatedAt, CreatedBy, LastUpdatedAt, LastUpdatedBy.
- Data of Business Application: Data of Business Application: Id, FriendlyName, Address, PhoneNo, EmailAddress, Status, AssignedSamProfiles, Certificate, ValidFrom, ValidTo

FDP_UCT.1	Basic data exchange confidentiality
Hierarchical to:	No other components.
Dependencies:	[FTP_ITC.1 Inter-TSF Trusted channel, or FTP_TRP.1 Trusted path] [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_UCT.1.1	The TSF shall enforce the *Signer Flow SFP and Privileged User Flow SFP* [122] to *transmit and receive* [123] user data in a manner protected from unauthorised disclosure.

FDP_UIT.1	Data exchange integrity
Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]

[121] [assignment: *additional importation control rules*]

[122] [assignment: *access control SFP(s) and/or information flow control SFP(s)]*

[123] [selection: *transmit, receive]*

[FTP_ITC.1 Inter-TSF Trusted channel, or

FTP_TRP.1 Trusted path]

FDP_UIT.1.1 The TSF shall enforce the *Signer Flow SFP and Privileged User Flow SFP* [124] to be able to *transmit and receive* [125] user data in a manner protected from *modification and insertion* [126] 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* [127] ***for R.Signer and R.Privileged User and for R.SAD for modification and replay*** has occurred.

Application Note 65 (Application Note 61 from [5])

Insertion of objects would mean that authorised creation of Signer and Privileged User could be possible.

6.4.4 Identification & Authentication (FIA)

FIA_AFL.1/BA	Authentication failure handling

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1 / BA The TSF shall detect when the **fails to authentication to the Ascertia ADSS Server SAM** [128] ~~**unsuccessful authentication attempts occur**~~ related to *Privileged User* **(Business Application)** ~~***and Signer***~~ *authentication*. [129]

FIA_AFL.1.2 / BA When the defined number of unsuccessful authentication attempts has been *met* [130], the TSF shall *suspend the Privileged User* **(Business Application) with marks it inactive** ~~***and when it is a Signer suspend the usage of R.Signing_Key_Id***~~ [131]

Application Note 66 (Application Note 62 from [5], refined by ST author)

Applied. The ST writer extended and iterated the FIA_AFL.1 to introduce the operations.

Application Note 67 (Application Note 63 from [5], refined by ST author)

Applied. The TOE uses direct authentication.

[124] [assignment: *access control SFP(s) and/or information flow control SFP(s)]*

[125] [selection: *transmit, receive*]

[126] [selection: *modification, deletion, insertion, replay*]

[127] [selection: *modification, deletion, insertion, replay*]

[128] [selection: *[assignment: positive integer number], a TOE Maintenance configurable positive within [assignment: range of acceptable values]]*

[129] [assignment: *list of authentication events*]

[130] [selection: *met, surpassed*]

[131] [assignment: *list of actions*]

Application Note 68 (from ST Author)

For operator authentication the TSF relies on the underlying smartcard/token holding the operator's TLS client authentication key/certificate how many unsuccessful login attempt does it allow to an operator before card/token is locked. This limit may vary from device to device. The Operator cannot be identified in case of wrong certificate and it's not possible to count the number of tries. That is why FIA_AFL.1 is not iterated to Operator.

Application Note 69 (from ST Author)

Refined the original description from the [5] to specify the detection of the failing of the Privileged User (Business Application) authentication.

Application Note 70 (from ST Author)

The Privileged User (Operator) can activate the Privileged User (Business Application) when it is inactive.

FIA_AFL.1/Signer	Authentication failure handling

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1 / Signer The TSF shall detect when a TOE Maintenance configurable positive integer defined in the Global Settings of ADSS Server SAM (R.TSF_DATA) [132] unsuccessful authentication attempts occur related to ~~***Privileged User and***~~ *Signer authentication*. [133]

FIA_AFL.1.2 / Signer When the defined number of unsuccessful authentication attempts has been *met* [134], the TSF shall **block the Signer for an amount of period defined in the Global Settings of ADSS Server SAM (R.TSF_DATA)** ~~***suspend the Privileged User and when it is a Signer suspend the usage of R.Signing_Key_Id***~~**.** [135]

Application Note 71 (from ST Author)

Refined the original description from the [5] to specify the detection of the failing of the Signer authentication.

FIA_ATD.1	User attribute definition

Hierarchical to: No other components.

Dependencies: No dependencies.

[132] [selection: *[assignment: positive integer number], a TOE Maintenance configurable positive within [assignment: range of acceptable values]]*

[133] [assignment: *list of authentication events*]

[134] [selection: *met, surpassed*]

[135] [assignment: *list of actions*]

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*. [136]

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

1) Establishing a trusted path between: -
   a. remote Signer and the TOE by means of TSF required by FTP_TRP.1/SIC.
   b. Privileged User **(Operator and Business application)** and the TOE by means of TSF required by FTP_TRP.1/SSA.
2) Identification of the user by means of TSF required by FIA_UID.2 [137]

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 two OTPs (one sent on Signer's mobile number and other on Signer's email) with mobile device authentication using biometric fingerprint/faceID or device PIN authentication when a new mobile device is used [138] to support Signer authentication.

FIA_UAU.5.2/ Signer The TSF shall authenticate any ***Signer's*** claimed identity according to the following: -

- the Signer provides fingerprint (inherence authentication factor); and

[136] [assignment: *list of security attributes*]
[137] [assignment: *list of TSF mediated actions*]
[138] [selection: *[assignment: list of direct authentication mechanisms conformant to EN 419 241-1 [6] SRA_SAP.1.1, [assignment: list of delegated authentication mechanisms conformant to EN 419 241-1 [6] SRA_SAP.1.1]*]

- the Signer provides OTPs (possession-based authentication factor) when a new mobile device is used. [139]

Application Note 72 (Application Note 64 from [5], refined by ST author)

Applied. This SFR only applies to Signer authentication for signing (FDP_ACC.1/Signing).

The TOE does not use delegated authentication.

Successful authentication gives Signer access to the relevant R.Signer object as the owner.

The above mentioned authentication mechanisms are used to register a device. After the registration process has successfully finished the Signer will authenticate himself accessing the mobiles secure element and signing SAD with his authorisation key.

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: -

1) The authentication is done using TLS client certificate
2) The certificate is read from a card which is PIN protected

to support **Privileged User (Operator)** authentication;

1) The authentication is done using TLS client certificate [140]

to support **Privileged User (Business Application)** authentication.

FIA_UAU.5.2/ Privileged User

The TSF shall authenticate any **Privileged User's** claimed identity according to the following: Privileged User uses its TLS authentication certificate. [141]

Application Note 73 (from ST Author)

The ST author refined this SFR to separate the two types of Privileged Users.

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.

[139] [selection: [assignment:*the rules describing how delegated authentication is verified by the TSF], [assignment: the rules describing how direct authentication mechanisms provide authentication]*]

[140] [assignment: *list of authentication mechanisms*]

[141] [assignment: *rules describing how the multiple authentication mechanisms provide authentication*]

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; and*
5) *R. Authorisation_Data* [142]

to Signer: -

1) *R.Reference_Priviliged_User_Authentication_Data; and*
2) R.Privileged_User [143]

to Privileged User. [144]

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* **(Business Application)** *authorized to create a new Signer.*
2) *Whether the subject is a Privileged User* **(Operator)** *authorized to create a new Privileged User* **(Business Application or Operator)**.
3) None [145] [146]

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* **(Business Application, Operator)** *authorized to modify an R.Signer object.*
2) ~~***Whether the subject is a Signer authorized to modify his own R.Signer object.***~~

[142] [assignment: *list of user security attributes*]
[143] [assignment: *list of user security attributes*]
[144] [assignment: *list of user security attributes*]
[145] [assignment: *rules for the initial association of attributes]*
[146] [assignment: *rules for the initial association of attributes*]

3) None [147] [148]

Application Note 74 (Application Note 65 from [5])

In FIA_USB.1.2 several attributes including R.Signing_Key_ID, R.SVD and R.DTBS/R may initially be empty.

Application Note 75 (Application Note 66 from [5], refined by ST author)

Applied. Detailed information about R.Authorisation_Data in Application Note 2 in sec. 3.1.

Application Note 76 (Application Note 67 from [5], refined by ST author)

Applied. The Business Application (SCA) provides the document hash (DTBS/R) to RAS Service (SSA) which passes it to the TOE for signature computation.

6.4.5 Security Management (FMT)

FMT_MSA.1/Signer	Management of security attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or

FDP_IFC.1 Subset information flow control]

FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_MSA.1.1/ Signer The TSF shall enforce the: -

1) *Signer Creation SFP* [149] to restrict the ability to *create* [150] the security attributes *listed in FIA_USB.1 for Signer* [151] to *authorised Privileged User* **(Business Application)** [152].

2) *Generate Signer Key Pair SFP* [153] to restrict the ability to *generate* [154] the security attributes *R.SVD and R.Signing_Key_Id* **as part of R.Signer** [155] to *authorised Privileged User* **(Business Application)** ~~**and Signer**~~ [156].

[147] [assignment: *rules for the changing of attributes*]
[148] [assignment: *rules for the changing of attributes*]
[149] [assignment: *access control SFP(s), information flow control SFP(s)]*
[150] [selection: *change_default, query, modify, delete, [assignment: other operations]]*
[151] [assignment: *list of security attributes*]
[152] [assignment: *the authorised identified roles*]
[153] [assignment: *access control SFP(s), information flow control SFP(s)]*
[154] [selection: *change_default, query, modify, delete, [assignment: other operations]]*
[155] [assignment: *list of security attributes*]
[156] [assignment: *the authorised identified roles*]

3) *Signer Key Pair Deletion SFP* [157] to restrict the ability to *destruct* [158] the security attribute *R.SVD and R.Signing Key Id as part of R.Signer* [159] to *authorised* **Privileged User (Business Application, Operator)** ~~***Signer***~~. [160]

4) *Supply DTBS/R SFP* [161] to restrict the ability to *create* [162] the security attribute *R.DTBS/R as part of R.Signer* [163] to *authorised Privileged User* **(Business Application)** [164].

5) *Signing SFP* [165] to restrict the ability to *create* [166] the security attribute *R.DTBS/R as part of R.Signer* [167] to *authorised Signer.* [168]

6) *Signing SFP* [169] to restrict the ability to *query* [170] the security attributes *as listed in FIA_USB.1* [171] to *authorised Signer.* [172]

7) *Signer Maintenance SFP* [173] to restrict the ability to *change* [174] the security attributes *R.Reference Signer Authentication Data* [175] to *authorised Privileged User* **(Privileged User)** ~~***and Signer***~~. [176]

FMT_MSA.1/Privileged User	Management of security attributes
Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]

[157] [assignment: *access control SFP(s), information flow control SFP(s)]*
[158] [selection: *change_default, query, modify, delete, [assignment: other operations]]*
[159] [assignment: *list of security attributes*]
[160] [assignment: *the authorised identified roles*]
[161] [assignment: *access control SFP(s), information flow control SFP(s)]*
[162] [selection: *change_default, query, modify, delete, [assignment: other operations]]*
[163] [assignment: *list of security attributes*]
[164] [assignment: *the authorised identified roles*]
[165] [assignment: *access control SFP(s), information flow control SFP(s)]*
[166] [selection: *change_default, query, modify, delete, [assignment: other operations]]*
[167] [assignment: *list of security attributes*]
[168] [assignment: *the authorised identified roles*]
[169] [assignment: *access control SFP(s), information flow control SFP(s)]*
[170] [selection: *change_default, query, modify, delete, [assignment: other operations]]*
[171] [assignment: *list of security attributes*]
[172] [assignment: *the authorised identified roles*]
[173] [assignment: *access control SFP(s), information flow control SFP(s)]*
[174] [selection: *change_default, query, modify, delete, [assignment: other operations]]*
[175] [assignment: *list of security attributes*]
[176] [assignment: *the authorised identified roles*]

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* [177] to restrict the ability to *create and query* [178] the security attributes *listed in FIA_USB.1 for Privileged User* [179] *to authorised Privileged User*. [180]

FMT_MSA.2 Secure security attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or

FDP_IFC.1 Subset information flow control]

FMT_MSA.1 Management of security attributes

FMT_SMR.1 Security roles

FMT_MSA.2.1 The TSF shall ensure that only secure values are accepted for *all security attributes listed in FIA_USB.1.* [181]

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* [182] to provide *restrictive* [183] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/ Signer The TSF shall allow the *Privileged User* **(Business Application)** [184] to specify alternative initial values to override the default values when an object or information is created.

FMT_MSA.3/Privileged User Static attribute initialisation

[177] [assignment: *access control SFP(s), information flow control SFP(s)]*

[178] [selection: *change_default, query, modify, delete, [assignment: other operations]]*

[179] [assignment: *list of security attributes*]

[180] [assignment: *the authorised identified roles*]

[181] [assignment: *list of security attributes*]

[182] [assignment: *access control SFP, information flow control SFP*]

[183] [*selection, choose one of: restrictive, permissive, [assignment: other property]]*

[184] [assignment: *the authorised identified roles*]

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* [185] to provide *restrictive* [186] 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* **(Operator)** [187] to specify alternative initial values to override the default values when an object or information is created.

FMT_MTD.1	Management of TSF data
Hierarchical to:	No other components.
Dependencies:	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MTD.1.1	The TSF shall restrict the ability to: - 1) *modify* [188] the *R.TSF_DATA data* [189] to *Privileged User* **(Operator)**. [190]

Application Note 77 (Application Note 68 from [5])

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;* *3) Configuration management; and* 4) None. [191] [192]

[185] [assignment: *access control SFP, information flow control SFP*]
[186] [*selection, choose one of: restrictive, permissive, [assignment: other property]]*
[187] [assignment: *the authorised identified roles*]
[188] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]]*
[189] [assignment: *list of TSF data*]
[190] [assignment: *the authorised identified roles*]
[191] [assignment: *additional list of management functions to be provided by the TSF*]
[192] [assignment: *list of management functions to be provided by the TSF*]

FMT_SMR.2	Restrictions on security roles

Hierarchical to: FMT_SMR.1 Security roles

Dependencies: FIA_UTD.1 Timing of identification

FMT_SMR.2.1 The TSF shall maintain the roles: *Signer and Privileged User* **(Operator and Business Application)**[193], none. [194]

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 can’t be a Privileged User* are satisfied.

Application Note 78 (Application Note 69 from [5], refined by ST author)

Applied. The ST writer described which roles are defined in the TOE and which operations the role can perform.

6.4.6 Protection of the TSF (FPT)

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 79 (Application Note 70 from [5])

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 [22] for Security Level 3.

FPT_PHP.3	Resistance to physical attack

Hierarchical to: No other components.

[193] [assignment: *authorised identified roles*]

[194] [assignment: *other authorised identified roles*]

Dependencies: No dependencies.

FPT_PHP.3.1 TSF shall resist opening the case [195] to the cover [196] by responding automatically such that the SFRs are always enforced.

Application Note 80 (Application Note 69 from [5], refined by ST author)

Applied. The TOE is not implemented as a local application with the same physical boundary as the CM. To make it more clear: the CM has its own tamper protection but the TOE is not implemented inside the CM. Both the CM and the TOE are put inside an appliance and the appliances protection is responsible for the protection of the TOE. If the appliance detects tamper it notifies and starts the CMs tamper protection procedures as well.

Application Note 81 (Application Note 70 from [5])

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 [22] 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 [22] 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*. [197]

FPT_RPL.1.2 The TSF shall perform *reject the signature operation* [198] 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 82 (Application Note 71 from [5], refined by the ST Author)

The system clock is the time source. One or more NTP Servers can be configured in TOE to ensure its system clock is synched with a trusted clock. In case of time deviation is detected

[195] [assignment: *physical tampering scenarios*]
[196] [assignment: *list of TSF devices/elements*]
[197] [assignment: *list of identified entities*]
[198] [assignment: *list of specific actions*]

by TOE, it automatically notifies and suspends its operations, i.e. Signer creation, signature creation, etc.

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; and*
7) *R.Reference_Privileged_User_Authentication_Data*
8) *R.TSF_DATA.* [199]

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* [200] when interpreting the TSF data from another trusted IT product.

Application Note 83 (Application Note 72 from [5], refined by ST author)

Applied. The SFR is used to handle the situation where the whole or part of the above data are stored outside the TOE.

6.4.7 Trusted Paths/Channels (FTP)

FTP_TRP.1/SSA	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*** **(Operator and Business Application)** ***through SSA*** [201]

[199] [assignment: *list of TSF data types*]
[200] [assignment: *list of interpretation rules to be applied by the TSF*]
[201] [selection: *remote, local]*

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.* [202]

FTP_TRP.1.2/ SSA The TSF shall permit: ***Privileged User* (Operator and Business Application) *through SSA***. [203] 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. [204] [205]

Application Note 84 (Application Note 73 from [5])

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 the SIC*** [206] 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.* [207]

FTP_TRP.1.2/ SIC The TSF shall permit: ***Remote Signer through the SIC*** [208] 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***~~

[202] [selection: *modification, disclosure, [assignment: other services for which trusted path is required]]*
[203] [selection: *the TSF, local users, remote users*]
[204] [assignment: *other services for which trusted path is required*]
[205] [selection: *initial user authentication, [assignment: other services for which trusted path is required]*]
[206] [selection: *remote, local]*
[207] [selection: *modification, disclosure, [assignment: other services for which trusted path is required]]*
[208] [selection: *the TSF, local users, remote users*]

3) ~~*FDP_ACC.1/Signer Key Pair Deletion*~~
4) *FDP_ACC.1/Signing*
5) None [209] [210]

Application Note 85 (Application Note 74 from [5], refined by ST author)

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 ST writer shall describe if the SAP can be used to transmit sensitive data and how these are protected in confidentiality.

The TOE is not expected to verify the SIC as a communication end-point and it may rely on the Signer authentication.

Application Note 86 (from ST Author)

The communication path is the following: SIC – SAP – SSA – TOE. From the SIC the communication is done over TLS 1.2 authenticated channel.

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 ***CM certified according to EN 419 221-5 [7]*** that is logically distinct from other communication paths and provides assured authentication of its end points and protection of the communicated data from modification or disclosure.

FTP_ITC.1.2/CM The TSF shall permit the ***TSF and a CM certified according to EN 419 221-5 [7]*** [211] to initiate communication via the trusted channel.

FTP_ITC.1.3/CM The TSF shall initiate communication via the trusted channel for all functions which need a CM. [212]

Application Note 87 (Application Note 75 from [5], refined by ST author)

Applied. Communication with the CM is through a secure channel using vendor specific APIs commands in any case.

[209] [assignment: *other services for which trusted path is required*]
[210] [selection: *initial user authentication, [assignment: other services for which trusted path is required]*]
[211] [selection: *the TSF, another trusted IT product*]
[212] [assignment: *list of functions for which a trusted channel is required*]

6.5 Security Requirements Rationale

6.5.1 Security Requirements Coverage

The following table is used to demonstrate that every SFR is used to cover an objective and that every objective is covered by an SFR.

	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/RSA			X													X	
FCS_CKM.1/ECDSA			X													X	
FCS_CKM.1/AES			X													X	
FCS_CKM.4			X														
FCS_COP.1/ DIG_SIG_GEN			X												X	X	
FCS_COP.1 /DIG_SIG_VER			X												X	X	
FCS_COP.1/HASH			X												X	X	
FCS_COP.1/HMAC			X												X	X	
FCS_COP.1/ENC			X												X	X	
FCS_RNG.1			X														X
User Data Protection																	

	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_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_ACF.1/Signer Maintenance		X															
FDP_ACC.1/Signer Maintenance		X															
FDP_ACC.1/Signer Key Pair Generation			X	X													
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		

	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/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/BA						X					X						
FIA_AFL.1/Signer											X						
FIA_ATD.1	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_UAU.1						X					X						
FIA_UAU.5/Signer											X						
FIA_UAU.5/Privileged User						X											
FIA_UID.2					X		X	X									
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							

	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
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		

Table 6-6: Security requirements coverage

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 is described in FIA_ATD.1 and FIA_USB.1.

OT.REFERENCE_SIGNER AUTHENTICATION_DATA is 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_Authentication_Data.

OT.SIGNER_KEY_PAIR_GENERATION is handled by the requirements for key generation and cryptographic algorithms in FCS_CKM.1/RSA, FCS_CKM.1/ECDSA, FCS_CKM.1/AES, FCS_COP.1/DIG_SIG_GEN, FCS_COP.1/DIG_SIG_VER, FCS_COP.1/HASH, FCS_COP.1/HMAC, FCS_COP.1/ENC. 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 CM.

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 is 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/BA, FIA_UAU.1 and FIA_UAU.5/Privileged User.

OT.PRIVILEGED_USER_PROTECTION is handled by requirements export and import of 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, FIA_UID.2. The actual description of the data is described in FIA_ATD.1 and FIA_USB.1

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 Key Pair Deletionand FDP_ACF.1/Signer Key Pair Deletion. 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 a 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/BA, FIA_AFL.1/Signer, 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 FT_TRP.1/SSA and FTP_TRP.1/SIC requiring data transmission to be protected in integrity. Also covered by access control rules FDP_ACC.1/Supply DTBS/R and FDP_ACF.1/Supply DTBS/R for transmitting DTBS/R to the TSF.

OT.SIGNATURE_INTEGRITY is handled by FCS_COP.1/DIG_SIG_GEN, FCS_COP.1/DIG_SIG_VER, FCS_COP.1/HASH, FCS_COP.1/HMAC, FCS_COP.1/ENC, which describes requirements for algorithms. FTP_ITC.1/CM may be used to transmit data securely between the TOE and the CM. 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/RSA, FCS_CKM.1/ECDSA, FCS_CKM.1/AES, FCS_COP.1/DIG_SIG_GEN, FCS_COP.1/DIG_SIG_VER, FCS_COP.1/HASH, FCS_COP.1/HMAC, FCS_COP.1/ENC, which describes requirements for key generation and algorithms.

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

6.6 SFR Dependencies

The dependencies between SFRs are addressed as shown in.

Requirement	Dependencies	Fulfilled by
FAU_GEN.1	FPT_STM.1	FPT_STM.1
FAU_GEN.2	FAU_GEN.1 FIA_UID.1	FAU_GEN.1 FIA_UID.2
FCS_CKM.1/RSA	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4	FCS_COP.1/DIG_SIG_GEN, FCS_COP.1/DIG_SIG_VER, FCS_COP.1/HASH and FCS_CKM.4
FCS_CKM.1/ECDSA	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4	FCS_COP.1/DIG_SIG_GEN, FCS_COP.1/DIG_SIG_VER, FCS_COP.1/HASH and FCS_CKM.4
FCS_CKM.1/AES	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4	FCS_COP.1/DIG_SIG_GEN, FCS_COP.1/DIG_SIG_VER, FCS_COP.1/HASH and FCS_CKM.4
FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1/*
FCS_COP.1/DIG_SIG_GEN	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/RSA, FCS_CKM.1/ECDSA and FCS_CKM.4
FCS_COP.1/DIG_SIG_VER	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/RSA, FCS_CKM.1/ECDSA and FCS_CKM.4
FCS_COP.1/HASH	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/RSA, FCS_CKM.1/ECDSA and FCS_CKM.4
FCS_COP.1/HMAC	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/RSA, FCS_CKM.1/ECDSA and FCS_CKM.4
FCS_COP.1/ENC	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/AES and FCS_CKM.4

Requirement	Dependencies	Fulfilled by
FCS_RNG.1	None	No dependents
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 Key Pair Generation	FDP_ACF.1	FDP_ACF.1/Signer Key Pair Generation
FDP_ACC.1/Signer Maintenance	FDP_ACF.1	FDP_ACF.1/Signer Maintenance
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 Key Pair Generation	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Signer Key Pair Generation FMT_MSA.3/Signer
FDP_ACF.1/Signer Maintenance	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Signer Maintenance 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

Requirement	Dependencies	Fulfilled by
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] [FTP_ITC.1 or FTP_TRP.1]	FDP_IFC.1/Signer FDP_IFC.1/Privileged User FTP_TRP.1/SSA and FTP_TRP.1/SIC
FIA_AFL.1/BA	FIA_UAU.1	FIA_UAU.1
FIA_AFL.1/Signer	FIA_UAU.1	FIA_UAU.1
FIA_ATD.1	None	No dependents
FIA_UAU.1	FIA_UID.1	FIA_UID.2
FIA_UAU.5/Signer	None	No dependents
FIA_UAU.5/Privileged User	None	No dependents
FIA_UID.2	None	No dependents
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

Requirement	Dependencies	Fulfilled by
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 FMT_SMR.2
FMT_MTD.1	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 FMT_SMF.1
FMT_SMF.1	None	No dependents
FMT_SMR.2	FIA_UID.1	FIA_UID.2
FPT_PHP.1	None	No dependents
FPT_PHP.3	None	No dependents
FPT_RPL.1	None	No dependents
FPT_STM.1	None	No dependents
FPT_TDC.1	None	No dependents
FTP_TRP.1/SSA	None	No dependents
FTP_TRP.1/SIC	None	No dependents
FTP_ITC.1/CM	None	No dependents

Table 6-7: Dependencies

6.7 Security Assurance Requirements

The security assurance requirement level is EAL 4 augmented with AVA_VAN.5. The assurance components are identified in the table below with the augmented item in bold.

Since the TOE is operated in a physically protected environment as described in OE.ENV an evaluation against this ST will probably not include physical attacks.

Assurance Class	Assurance components
ADV: Development	ADV_ARC.1 Security architecture description
	ADV_FSP.4 Complete functional specifications
	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

Assurance Class	Assurance components
	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_INT.1 ST introduction
	ASE_CCL.1 Conformance claims
	ASE_SPD.1 Security problem definition
	ASE_OBJ.2 Security objectives
	ASE_ECD.1 Extended components definition
	ASE_REQ.2 Derived security requirements
	ASE_TSS.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**

Table 6-8 Security Assurance Requirements: EAL 4 augmented with AVA_VAN.5

EAL 4 permits a developer to gain maximum assurance from positive security engineering based on good commercial development practises which, through rigorous, do not require substantial specialist knowledge, skills, and other resources. EAL 4 is the highest level at which it is likely to be economically feasible to retrofit to an existing product line.

As the TOE manages SCD 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.

EAL 4 is therefore augmented with AVA_VAN.5.

7. TOE Summary Specification

To fulfil the Security Functional Requirements, the TOE comprises the following Security Functions (TSF): -

1. User Roles and Authentication (TSF_AUTH);
2. Key Security (TSF_CRYPTO);
3. Access and information flow control (TSF_CTRL);
4. Data protection (TSF_DP);
5. Audit (TSF_AUDIT); and
6. Communication protection (TSF_COMM).

Each of the TOE security functions is described in the following sections in detail.

7.1 TOE Security Functions

7.1.1 User Roles & Authentication (TSF_AUTH)

#FMT_SMR.2 Restrictions on security roles

The TOE maintains Privileged User (Operator or Business Application) and Unprivileged User (Signer) roles and associates users with roles. The TOE identifies users by means of a unique user identifier. The TOE ensures that each user has only one role, consequently a Signer can't be a Privileged User. These users are stored and maintained in different subsystems and identified with different IDs (Operator ID, Client ID, User ID).

All the Operators and Business Application configurations are stored in Ascertia ADSS Server SAM database. The integrity of the records stored in Ascertia ADSS Server SAM database is protected by sequenced HMAC. The HMAC secret key is stored in the CM.

#FIA_UAU.5/Privileged User Multiple authentication mechanisms

Privileged Users authenticate via TLS channel to the TOE. Business Application authenticates with its clientId and certificate. Operator authenticates himself with his certificate stored on a PIN protected card.

#FIA_UID.2 User identification before any action
#FIA_UAU.1 Timing of authentication

The Signer's registered mobile device holds an authorization key pair in the phone's secure element. The key pair is created during the device registration process with the help of two OTPs. Signers must identify and authenticate themselves using two OTPs (sent on registered mobile number and email) i.e. Login to the Go>Sign Mobile app before doing anything to the TOE.

#FIA_UAU.5/Signer Multiple authentication mechanisms

Fingerprint authentication is used inside the Go>Sign Mobile app (if the Signer is already logged in) to authorize a signing request, i.e. create the SAD.

#FIA_AFL.1 Authentication failure handling

The TOE handles authentication failures in a separate way for each role.

Privileged User (Ascertia ADSS Server SAM Operator): Operator's TLS client authentication key/certificate should be held on a smartcard/token. At the time of login using TLS client certificate authentication mechanism, the first check is the PIN/password entry for the release of the TLS client private key. If this check fails then it is possible for the smart card/token to be locked, e.g. after three incorrect attempts.

Privileged User (Business Application): If a business application fails to authenticate, e.g. using a wrong TLS client certificate to connect then Ascertia ADSS Server SAM marks the business application as inactive and an Ascertia ADSS Server SAM Operator is then required to activate it.

Signer: Signer login retries limit and blockage interval after performing number of retries are defined in SAM Advanced Settings.

#FIA_ATD.1 User attribute definition
#FIA_USB.1 User-subject binding
#FMT_MSA.2
The TOE maintains accounts (with different security attributes) belonging to individual users. TOE validates the values assigned for these attributes.

7.1.2 Key Security (TSF_CRYPTO)

#FCS_CKM.1
#FCS_CKM.4
#FCS_COP.1
#FCS_RNG.1

The TOE calls with appropriate parameters a CM certified in conformance with EN 419 221-5 [7] for any key management or cryptographic operations, random number generation.

7.1.3 Access and information flow control (TSF_CTRL)

#FDP_ACC.1/Privileged User Creation
#FDP_ACF.1/Privileged User Creation
#FMT_MSA.1/Privileged User
#FMT_MSA.3/Privileged User
#FMT_SMF.1

When Ascertia ADSS Server SAM is installed, a default Ascertia ADSS Server SAM Operator account is automatically created with a default Operator's certificate. The default operator logs in to the Ascertia ADSS Server SAM Admin Console and creates Privileged Users, i.e. Ascertia ADSS Server SAM Operators and Client Applications. When Operator logs in to Ascertia ADSS Server SAM Admin Console using the default operator's certificate, then a warning dialog is shown to the Operator to instruct how to change the default certificate with new Operator certificate. If Operator doesn't setup new Operator certificate within 7 (seven) days, then operator access to Ascertia ADSS Server SAM Admin Console is blocked and complete reinstallation of Ascertia ADSS Server SAM would be required. Only Operators can manage Privileged Users after successful authentication.

#FDP_ACC.1/Signer Creation
#FDP_ACF.1/Signer Creation
#FMT_MSA.1/Signer
#FMT_MSA.3/Signer
#FMT_MSA.2
#FMT_SMF.1

The TOE guarantees that only a Business Application as a Privileged User can create new Signer and initiate key pair generation on behalf of the Signer. A typical Signer registration process involves registering Signer details and generating remote signing key pair and digital certificate. The Signer visits a Business Application web page and provides their registration details for the Business Application. The registration details include: -

- User Name (full name of the Signer);
- User ID (a unique identifier for the Signer);
- Password (optionally used to authenticate the Signer in the Business Application at the time of signing if provided);
- Email Address (used for subsequent communication purposes and to send the OTP during mobile device registration); and
- Mobile Number (to send the OTP during mobile device registration).

The Business Application receives the registration details from the Signer and creates a user registration request for RAS Service. The Business Application includes Application ID (clientId) in the registration request for RAS Service. The RAS Service internally requests the TOE to store the Signer registration details. The TOE maintains and validates all the Signer registration and security data in Ascertia ADSS Server SAM database. In a different scenario, the Signer may be registered by a RA officer where Signer details are entered in the system by RA officer after manually verifying the Signer's identity. In both cases, i.e. whether the Signer is directly registering with the business application or RA officer registers the user, the RAS Service would require same details about the Signer.

#FDP_ACC.1/Signer Key Pair Generation
#FDP_ACF.1/Signer Key Pair Generation
#FDP_IFC.1/Signer
#FDP_IFF.1/Signer

Once the Signer details are registered, the Business Application requests the ADSS RAS Service to generate the signing key pair for the Signer. The TOE does not use pre-generated keys. To generate a signing key pair for a registered Signer, the Business Application creates a request for RAS Service which then forwards the request to SAM. The request includes: -

- User ID (a unique identifier for the Signer).
- Key ID (a unique ID for the signing key pair).

- Subject DN (the Subject DN to be used in the CSR).
- SAM Profile ID (SAM Profile defines the key type, e.g. RSA or ECDSA, key length, e.g. 2048, Signature Padding Scheme, etc.).

The RAS Service requests the SAM to generate the Signer's signing key pair. SAM uses the CM to generate and securely store the signing key pair for the Signer. The signing key is generated in the HSM then its encrypted with a key held in the HSM. The encrypted key is then backed up securely to the database.

Once the signing key pair has been generated, the SAM transforms the R.SVD (the public key) to a CSR and gets it signed with the remote signing private key, i.e. self-signed. In response to signing key pair generation request, the RAS Service returns the CSR which the Business Application can send to an issuing CA to get a signing certificate. Once the Business Application has obtained the signing certificate from the issuing CA, the Signer registration process is considered complete.

The TOE guarantees that the Signer is the owner of the R.Signer object.

#FDP_ACC.1/Signer Maintenance
#FDP_ACF.1/Signer Maintenance

The Business Application and the Operator can modify the R.Signer's security attributes according to the R.Reference_Signer_Authentication_Data.

#FDP_ACC.1/Signer Key Pair Deletion
#FDP_ACF.1/Signer Key Pair Deletion

If the Business Application or the Operator requests the deletion of R.Signing_Key_Id then the TOE deletes the R.Signing Key Id and R.SVD from R.Signer in Ascertia ADSS Server SAM database. It also deletes the encrypted signing key from database.

#FDP_ACC.1/Supply DTBS/R
#FDP_ACF.1/Supply DTBS/R

To sign a PDF document, e.g. create a PAdES Part-4 LTV signature, the Business Application computes the PDF document hash and then puts it into the signed attributes of CAdES signature structure. The Business Application may also define the signature appearance attributes and meta information before the PDF document signed. Once the CAdES signature structure is ready then Business Application computes the hash (R.DTBS/R) over the CAdES signed attributes and requests the RAS Service to sign this hash with the Signer's signing key.

The TOE queues this authorisation request in the Ascertia ADSS Server SAM database against the Signer details. When Signer later logs in to Go>Sign Mobile app, the authorisation request is downloaded and shown to the Signer before Signer signs the authorisation response to authorise the signing of R.DTBS/R with the signing key.

At this stage, as the Signer is required to sign the authorisation response XML via Go>Sign Mobile app the RAS Service marks the signing transaction status as pending and responds to the Business Application accordingly.

After receiving the response that Signer's signing transaction is pending, the Business Application shows a message to the Signer that it should launch Go>Sign Mobile app on the mobile device and authorise the signing transaction.

#FDP_ACC.1/Signing
#FDP_ACF.1/Signing

The Signer has sole control over its remote signing keys by a dynamic authorization mechanism.

If not already installed, Signer would be required to first download and install the Go>Sign Mobile app from a store depending on the mobile platform in use.

As soon as the Signer launches the Go>Sign Mobile app on the mobile device, the Signer is asked to login (assuming not already signed in) by providing their User ID and two OTPs sent to the users registered email address and phone number. Signer is presented the Go>Sign Mobile app dashboard screen after successful authentication.

On first successful login, the Go>Sign Mobile app generates an authorisation key pair in the mobile device's secure element/enclave. To access the mobile device's secure element/enclave, the Go>Sign Mobile app authenticates the Signer via fingerprint/faceID biometric authentication or device PIN.

The authentication using OTP values and authorisation key pair generation only happens when the Signer logs in to Go>Sign Mobile app on a mobile device. On subsequent launches of the Go>Sign app if the previous session was continued (i.e. user did not logout), the OTP authentication step is automatically skipped.

After the authorisation key pair has been generated in the mobile device's secure element/enclave, the Go>Sign Mobile app generates a CSR having the User ID as the Subject DN and the authorisation public key is set in the CSR. This CSR along with the unique Device ID is sent to RAS Service for registration. The RAS Service communicates with the CA (TSP) to get certificate for this CSR. The certificate is received back from the CA and then the RAS Service provides this certificate to the ADSS Server SAM (TOE). The TOE associates the signer's authorisation certificate and Device ID the registered Signer's details.

If the same Signer tries to login to Go>Sign Mobile app using another mobile device, they would be authenticated again using OTP values and a new authorisation key pair would be generated followed by new mobile device registration with the TOE. This would allow the Signer to authorise the signing transactions using any of their registered mobile devices.

Once the Signer's authorisation key pair has been generated and the mobile device has been registered with the SAM, the Go>Sign Mobile app requests the RAS Service to provide the pending authorisation requests. If there is an authorisation request waiting to be signed by Signer, it is downloaded and signing transaction details are shown to the Signer.

Once Signer has seen the details of the signing transaction, he authorises it by signing the authorisation response. As signing the authorisation response requires accessing the authorisation private key held in the mobile device's secure element/enclave, so it requires Signer to authenticate via their fingerprint. After successful fingerprint authentication, Go>Sign Mobile app signs the authorisation response with the authorisation private key.

The authorisation request is composed of an XML structure. Go>Sign Mobile app puts the Device ID into the XML and then signs this – this data structure is referred to as the authorisation response message. This authorisation response message is sent to RAS Service for verification. The RAS Service in turn gets the SAM to verify the authorisation response message using the authorisation public key certificate already available in the Signer's details. The SAM also verifies all the other details inside the authorisation response message e.g. the mobile device ID, salt information, validity dates etc. The SAM records the authorisation response message in its database log records, and then communicates with the CM over a trusted channel to activate the user's signing key. The returned signature value from the CM is returned back to the RAS Service by the SAM.

SAM authorises itself against the CM before it activates a remote signing key in the CM. The key authorisation is achieved by invoking the "AuthoriseKey" function of the CM. The "AuthoriseKey" function uses the public key part of the user's authorisation key pair. The CM generates a challenge which is signed by the SAM using user's authorisation private key. If the signer is authorised and SAM is authorised to CM, it imports the Signer's encrypted signing key from the database, decrypts it, create the signature then deletes the decrypted signing key.

The TOE ensures that Signer shall authorise any subsequent use of the signing key.

#FDP_ACC.1/TOE Maintenance
#FDP_ACF.1/TOE Maintenance
#FMT_MTD.1
#FMT_SMF.1
#FDP_IFC.1/Privileged user
#FDP_IFF.1/Privileged user

Only an authorised Ascertia ADSS Server SAM Operator can maintain the TOE configuration data via Ascertia ADSS Server SAM Admin Console. There is an Access Control module in Ascertia ADSS Server SAM Admin Console which allows creating the Ascertia ADSS Server SAM Operator Roles. At the time of creating a new operator, one of the available role is assigned to the operator. Each role defines what an operator is allowed to do, e.g. can it create other Operators, register Business Applications, create infrastructure keys/certificates, create RAS Profiles, etc.. Roles can even restrict read/write/delete access on the allowed modules.

#FDP_ETC.2/Signer
#FDP_ITC.2/Signer
#FDP_ETC.2/Privileged user
#FDP_ITC.2/Privileged user

The TOE does not import or export Signer data. Privileged User data can be exported and imported.

7.1.4 Data protection (TSF_DP)

#FPT_PHP.1 Passive detection of physical attack

The TOE implements security functionality against physical tamper.

#FPT_PHP.3 Resistance to physical attack

The TOE detects when the enclosure of the TOE is opened and zeroises sensitive data, and terminates main power. This ensures that the integrity and confidentiality of the assets are preserved. During tamper state, all functionality of the TOE is stopped and no service is provided (both signatory ones and administrative ones) even if the TOE is hardware restarted. When the TOE is hardware restarted it will maintain the tamper state such that the previous tamper condition can be reported.

#FPT_RPL.1 Replay detection

The TOE rejects the signature operation if a SAD is being used more than once. The communication between RAS Service and Signer are based on a proprietary SAP. The SAP is protected against replay, bypass and forgery attack, using a salt (random value to avoid replay attack), a validity period and the PKCS#1 authorization signature of the Signer.

#FPT_STM.1 Reliable time stamps

The source of date and time is the system clock where TOE is deployed. To make sure TOE uses reliable time, one or more NTP Servers can be configured to ensure its clock is synched with a trusted clock. In case of time deviation is detected by the TOE, it suspends its operations, i.e. R.Signer creation, signature creation, etc.

#FPT_TDC.1 Inter-TSF basic TSF data consistency

Whenever the TOE exchanges sensitive data with other components outside of the TOE boundary uses data integrity either on the data or on the communication channel when interpreting the data. The TOE guarantees that the interpretation of main resources will remain consistent. However, some of the main resources of the TOE are stored outside of the TOE. These resources are handled appropriately concerning integrity and confidentiality.

TOE ensures integrity of resources stored in Ascertia ADSS Server SAM database. The integrity of the records stored in Ascertia ADSS Server SAM database is protected by sequenced HMAC. The HMAC secret key is stored in the CM. Confidential data is stored in the CM or stored in Ascertia ADSS Server SAM database encrypted by AES 256-bit symmetric keys stored in the CM.

7.1.5 Audit (TSF_AUDIT)

#FAU_GEN.1 Audit Generation
#FAU_GEN.2 User identity association

The TOE uses an audit Database outside the TOE boundaries. The TOE logs every security related events into the Database. Each audit record contains date and time of the event (using reliable timestamp), type of event, subject identity (the identity of the user that caused the event if applicable, i.e. an identified user initiated the event), and the outcome (success or failure) of the event. The audit trail does not include any data which allows the retrieval of sensitive data.

The integrity of the data stored in any of the tables of the Database is protected by sequenced HMAC approach. The HMAC symmetric key is securely held in the CM.

7.1.6 Communication protection (TSF_COMM)

#FDP_UCT.1 Basic data exchange confidentiality
#FDP_UIT.1 Data exchange integrity
#FTP_TRP.1/SSA FTP_TRP.1/SIC Trusted path
#FTP_ITC.1/CM Inter-TSF trusted channel

The TOE provides protection of user data while in transit. It ensures both confidentiality and integrity. The SIC securely communicates with the RAS Service module, the SCA with the SSA and the SSA with the TOE over TLS v1.2 channel. Communication with the CM is through a secure channel using vendor specific APIs commands. Ascertia ADSS Server SAM Operators (as Privileged Users) access the Ascertia ADSS Server SAM Admin Console GUI over a mutually authenticated TLS v1.2 channel.

7.2 Fulfilment of the SFRs

This section shows that the TSFs are appropriate to fulfil the TOE Security Functional Requirements as specified in chapter 6.3.

The mapping of SFRs and TSFs is given in table 7.1. Table SFR - TSF relationship.

SFR	TSF
FAU_GEN.1	TSF_AUDIT
FAU_GEN.2	TSF_AUDIT
FCS_CKM.1/*	TSF_CRYPTO
FCS_CKM.4	TSF_CRYPTO
FCS_COP.1/*	TSF_CRYPTO
FCS_RNG.1	TSF_CRYPTO
FDP_ACC.1/Privileged User Creation	TSF_CTRL
FDP_ACF.1/Privileged User Creation	TSF_CTRL
FDP_ACC.1/Signer Creation	TSF_CTRL
FDP_ACF.1/Signer Creation	TSF_CTRL
FDP_ACC.1/Signer Key Pair Generation	TSF_CTRL, TSF_DP
FDP_ACF.1/Signer Key Pair Generation	TSF_CTRL
FDP_ACC.1/Signer Maintenance	TSF_CTRL
FDP_ACF.1/Signer Maintenance	TSF_CTRL
FDP_ACC.1/Signer Key Pair Deletion	TSF_CTRL
FDP_ACF.1/Signer Key Pair Deletion	TSF_CTRL
FDP_ACC.1/Supply DTBS/R	TSF_CTRL
FDP_ACF.1/Supply DTBS/R	TSF_CTRL
FDP_ACC.1/Signing	TSF_CTRL
FDP_ACF.1/Signing	TSF_CTRL
FDP_ACC.1/TOE Maintenance	TSF_CTRL
FDP_ACF.1/TOE Maintenance	TSF_CTRL

SFR	TSF
FDP_ETC.2/Signer	[213]
FDP_IFC.1/Signer	TSF_CTRL
FDP_IFF.1/Signer	TSF_CTRL
FDP_ETC.2/Privileged User	TSF_CTRL
FDP_IFC.1/Privileged user	TSF_CTRL
FDP_IFF.1/Privileged User	TSF_CTRL
FDP_ITC.2/Signer	[214]
FDP_ITC.2/Privileged User	TSF_CTRL
FDP_UCT.1	TSF_COMM
FDP_UIT.1	TSF_COMM
FIA_AFL.1/*	TSF_AUTH
FIA_ATD.1	TSF_AUTH
FIA_UAU.1	TSF_AUTH
FIA_UAU.5/Signer	TSF_AUTH
FIA_UAU.5/Privileged User	TSF_AUTH
FIA_UID.2	TSF_AUTH
FIA_USB.1	TSF_AUTH
FMT_MSA.1/Signer	TSF_CTRL
FMT_MSA.1/Privileged User	TSF_CTRL
FMT_MSA.2	TSF_AUTH, TSF_CTRL
FMT_MSA.3/Signer	TSF_CTRL
FMT_MSA.3/Privileged User	TSF_CTRL
FMT_MTD.1	TSF_CTRL
FMT_SMF.1	TSF_CTRL
FMT_SMR.2	TSF_AUTH
FPT_PHP.1	TSF_DP
FPT_PHP.3	TSF_DP
FPT_RPL.1	TSF_DP
FPT_STM.1	TSF_AUDIT
FPT_TDC.1	TSF_DP
FPT_TRP.1/SSA	TSF_COMM
FPT_TRP.1/SIC	TSF_COMM
FPT_ITC.1/CM	TSF_COMM

Table 7-1: SFR – TSF relationship

7.2.1 Security Requirements Coverage

Each TOE Security Functional Requirement is implemented by at least one Security Function (see 7.1. Table SFR - TSF relationship)

[213] Since the TOE does not export user data then FDP_ETC.2/Signer is trivially satisfied.

[214] Since the TOE does not export user data then FDP_ITC.2/Signer is trivially satisfied

8. Glossary and Acronyms

8.1 Acronyms

AC	Access Control
API	Application Programming Interface
CA	Certification Authority
CC	Common Criteria, ISO/IEC 15408, Evaluation criteria for IT security
CM	Cryptography Module certified according to prEN 419 221-5:2016
CSR	Certificate Signing Request
Certificate	Certificate for electronic signature as defined in eIDAS article 3.
DTBS/R	Data To Be Signed Representation
EAL	Evaluation Assurance Level
HMAC	Hash-based Message Authentication Code
HSM	Hardware Security Module
KEK	Key Encryption Key
OTP	One-Time Password
RDBMS	Relational database management system
SAD	Signature Activation Data
SAM	Signature Activation Module
SAP	Signature Activation Protocol
SCA	Signature Creation Application
SIC	Signer's Interaction Component
SSA	Server Signing Application
SVD	Signature Verification Data
ST	Security Target
TOE	Target of Evaluation
TSP	Trust Service Provider
TW4S	Trustworthy System Supporting Server Signing
QSCD	Qualified Electronic Signature (or Electronic Seal) Creation Device as defined in the eIDAS Regulation [8]

9. Bibliography

[1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2017-04-001, Version 3.1, Revision 5, April 2017

[2] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2017-04-002, Version 3.1, Revision 5, April 2017

[3] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; CCMB-2017-04-003, Version 3.1, Revision 5, April 2017

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

[5] prEN 419241-2: 2017, Trustworthy Systems Supporting Server Signing - Part 2: Protection Profile for QSCD for Server Signing, May 2018, v.016

[6] prEN 419241-1:2017, Trustworthy Systems Supporting Server Signing Part 1: General System Security Requirements

[7] prEN 419221-5:2016, Protection profiles for Trust Service Provider Cryptographic modules - Part 5: Cryptographic Module for Trust Services

[8] 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

[9] ETSI TS 119 312 v1.2.1 (2017-05) Electronic Signature and Infrastructures (ESU); Cryptographic Suites

[10] SOG-IS, Crypto Evaluation Scheme, Agreed Cryptographic Mechanisms, version 1.0, 2016

[11] ETSI EN 319 411-1 v1.1.1 (2016-02) Electronic Signatures and Infrastuctures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 1: General requirements

[12] ETSI EN 319 411-2 v2.1.1 (2016-02) Electronic Signatures and Infrastuctures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 2: Requirements for trust service providers issuing EU qualified certificates

[13] 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.

[14] 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.

[15] RSA Laboratories, PKCS #1: RSA Encryption Standard, Version v2.2, October 27, 2012

[16] FIPS PUB 186-4, Federal Information Processing Standards Publication, Digital Signature Standard (DSS), July 2013

[17] FIPS PUB 140-2, Federal Information Processing Standards Publication, Security Requirements for Cryptographic Modules, May 25, 2001

[18] X9.62-1998, Public Key Cryptography For The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA), September 20, 1998

[19] RFC 2104 – HMAC: Keyed-Hashing for Message Authentication, February 1997

[20] BSI AIS 20 / AIS 31, Functionality classes for random number generators Version 2.0, 18 September 2011

[21] NIST Special Publication 800-90A Recommendation for Random Number Generation Using Deterministic Random Bit Generators, June 2015

[22] ISO/IEC 19790:2012 Information technology – Security techniques – security requirements for cryptographic modules

[23] AGD_OPE: Operational User Guidance

[24] AGD_PRE: Preparation Procedure