swissbit®

Swissbit Cloud CSP-L - Common Criteria Security Target

Version 1.2.6 (d2dacdec38e278adc011f1af19660d13b12c4a22), 2024-11-06

Table of Contents

1. ST Introduction

The Fiscal Code of Germany [FCG] requires that records and accounts for an electronic record-keeping system (ERS) have to be protected by a certified technical security device. The Federal Office for Information Security defines requirements for the components of such a system, i.e. for the security module using Common Criteria Protection Profiles. The security module consists of a controller, executing the security module application (referenced as Client Remote Entity) and the cryptographic service provider (CSP-L). Therefore, this Security Target defines the security requirements that apply to the cryptographic service provider (CSP-L) Swissbit Cloud CSP-L in version 1.0.7 produced by Swissbit AG.

The requirements are defined in the Protection Profile "Cryptographic Service Provider Light" [PP-CSPL-V1.0].

1.1. ST and TOE Reference

Document Type

Security Target

Document based on commit

d2dacdec38e278adc011f1af19660d13b12c4a22

Document status

DRAFT

ST Reference

Swissbit Cloud CSPL - Common Criteria Security Target

Document Version

1.2.6

Sponsor

Swissbit AG, Industriestraße 4, 9552 Bronschhofen Schweiz

CC version

version 3.1 revision 5

Assurance Level

EAL 2 augmented with ALC_CMS.3, ALC_LCD.1

Certification ID

BSI-DSZ-CC-1238

TOE identification

Swissbit Cloud CSP-L

TOE version

1.0.7

1.2. TOE Overview

1.2.1. TOE type

The Swissbit Cloud CSP-L is a Cryptographic Service Provider Light (CSPLight) and is a pure software implementation, dedicated to provide cryptographic services for the protection of the confidentiality and the integrity of user data, and for entity authentication. Parts of these features are supported by the underlying hardware/software platform. Hereby, the TOE is operated in a secure data centre with a sufficiently high physical and organizational security level, to fulfill existing requirements for the operation in Technical Security Device scenarios.

The TOE does not provide an interface for direct end-user interaction. Instead, the TOE provides its services in form of a defined API to be consumed by other applications.

1.2.2. TOE description

The TOE is defined as a software component, i.e. a cryptographic library. The TOE is installed on and runs on a dedicated hardware platform. The hardware platform is not part of the TOE, but it is expected that the TOE adheres to the platform guidance and the TOE relies on functionality provided by the operating system.

The TOE security functionality (TSF) is logically defined by a common set of security services for users and security mechanisms for internal use. The cryptographic services for users/entities comprise:

- authentication of users/entities,
- authentication and attestation of the TOE to entities,
- data authentication and non-repudiation including time stamps,
- encryption and decryption of user data,
- trusted channel functionality including mutual authentication of the communicating entities, encryption and message authentication proof for the sent data, decryption and message

authentication verification for received data,

- management of cryptographic keys with security attributes including key generation, key derivation and key agreement, internal storage of keys, import and export of keys with protection of their confidentiality and integrity,
- generation of random bits which may be used for security services outside the TOE,
- clustering and synchronization of cryptographic keys and user entities between multiple TOE instances,
- generation of audit logs,
- generation of time stamps,
- methods for time synchronization with client applications

1.2.3. Major Security Feature of the TOE

The Swissbit Cloud CSP-L is a pure software TOE. Its main purpose is providing cryptographic operations and security services for security module applications such as the Security Module Application for Electronic Record-Keeping Systems (SMAERS) defined by BSI-CC-PP0105-V2-2020 (cf. **[PP-SMAERS]**). The main functions are:

- Cryptographic Operations used for
  - TSF data import incl. certificates and cryptographic keys,
  - Confidentiality protection of stored user data and TSF data.
- User authentication,
- Access control on cryptographic TSF and keys,
- TSF protection,
- Secure communication channel (establishment) with defined entities,
- Authentication & verification of update packages,
- Clustering for performance scalability and high availability and
- Timestamping and audit.

1.2.4. TOE Life-Cycle

The Life-Cycle of the released TOE is separated in the following phases:

Delivery

This phase represents the delivery to the customer. It is the phase, in which the TOE leaves the secure premises of the Swissbit AG.

Initialization

Before put into operation, the TOE must be initialized/personalized. Here Administrator configures the clustering behaviour (each master is assigend exacty one slave), imports attestation keys and trust anchors, imports or generates a chip authentication key pair, generates a signature from the CA for it, and imports this into the TOE. In addition, trust anchors for secure NTP have to be configured and the port on which the TOE listens is set.

Operations

After the initialization process, the TOE starts the actual operational phase. All security functionality of the TOE is operating as specified within this Security Target.

Update

In case of bug fixes or addition of new features to the TOE, Swissbit AG will create updated releases of the TOE.[1] The update will be installed manually by the administrator of the Swissbit Cloud CSP-L.

End-of-Life

This phase is entered intentionally if the operational phase of a TOE should be ended permanently. Here, the Administrator has to modify the clustering behaviour of the remaining instances. Especially, if the master of the cluster enters this phase, the slave has to be assigned as new master and all other instances have to be configured appropriately. Finally, the cryptographic key material has to be cleared from the TOE's platform accordingly.

1.2.5. Method of Use

The TOE is intended to be used with different applications. The TOE security services are logically separated and provided through well-defined external interfaces towards these applications. The operational environment can not affect the security and correctness of the TSF, but it supports the availability of the TSF.

Next to those defined in **[PP-CSPL-V1.0]**, the Swissbit Cloud CSP-L also provides time service and time stamp service (cmp. **[PP-CSPL-TS-AU-V1.0]**) and clustering (cmp. **[PP-CSPL-Cluster-V1.0]**) as additional methods of use. In addition the TOE allows applications to retrieve random bits from a physical random generator to enable these applications to seed their pseudo random number generators.

1.2.6. Physical Scope of the TOE

The TOE is a pure software implementation, preinstalled an a non-TOE secure platform (cmp. section **Non-TOE Hardware**), updates are provided via Update Code Packages which get installed by an administrator. Hence, the TOE as a software application does not have any direct physical boundaries, since the hardware and software platform are not part of the TOE. Nevertheless, this Security Target (Swissbit Cloud CSPL - Common Criteria Security Target in version 1.2.6), as well as the guidance documentation of the Swissbit Cloud CSP-L (**[CSPL-AGD]**) are part of the TOE.

As desired, the guidance documents and the TOE are delivered to the customer via a personal delivery, encrypted and signed email or a secure download portal.

1.2.7. Logical Scope of the TOE

The TOE provides the following security features (SF), listed in the order the SFRs arising in the **[PP-CSPL-V1.0]**, **[PP-CSPL-TS-AU-V1.0]** and **[PP-CSPL-Cluster-V1.0]**:

- **Key Management**
  - Access Control to key management functionalities,
  - Management of key security attributes,
  - Attribute initialization,
  - Hash Generation,
  - Key Generation/ Derivation/ Agreement/ Destruction/ & Wrapping,
  - Certificate Management, and
  - Random Number Generation.
- **Cryptographic operations for encryption & decryption**
- **Hybrid encryption for user data**
- **Data integrity mechanisms**
  - HMAC
  - CMAC
- **Authentication & attestation of the TOE, trusted channel**
  - Digital Signature Service (ECDSA & RSA)
  - Trusted Channel Establishment (PACE, Terminal authentication, Chip authentication)
  - Verification of authenticity of the TOE

- **User identification & authentication**
  - User attribute and reference data definition, handling & management
  - Authentication failure handling
  - User-subject binding
  - Authentication timing restrictions
  - Multiple authentication mechanisms
  - Re-Authentication rules
- **Access control**
- **Security Management**
- **Protection of the TSF**
  - Secure State
  - Testing of the TSF
- **Secure Update**
- **Time Stamp**
  - Time synchronization with a trusted time service
- **Security Audit**
  - Generation of audit logs of auditable events acc. [PP-CSPL-TS-AU-V1.0] and [PP-CSPL-Cluster-V1.0]
- **Clustering**, scalability of
  - performance,
  - availability,
- Retrieval of random bits from a physical random number generator to seed pseudo random number generators of authenticated clients, and
- methods for time synchronization with client applications.

1.2.8. Non-TOE components available to the TOE

The Swissbit Cloud CSP-L requires the provision of an operating environment consisting of software and hardware components as defined below.

Non-TOE Software

The Swissbit Cloud CSP-L is a software. As such it requires the existence of an Operating System (Ubuntu Linux 24.04). As the Swissbit Cloud CSP-L has been developed in Java, it also requires a Java Runtime Environment (JRE) of version 17. In order to access the TPRNG (True Physical Random Number Generator) of the platform, a daemon process is required to run on the OS. This daemon is developed and provided by the manufacturer of the hardware platform and ensures that the random from the TPRNG is provided to the Swissbit Cloud CSP-L via the device file /dev/random-enforcer

Non-TOE Hardware

The Swissbit Cloud CSP-L is run within a dedicated hardware platform together with the operating system mentioned in **Non-TOE Software** that supports execution of the CSPLight.

The hardware platform has to be certified and the Swissbit Cloud CSP-L is the only software that is executed on the platform, both as required by **[PP-SMAERS]** “Appendix: Operational Requirements for CSPLight”. The platform specifically provides physical protection and a TPRNG to the TOE.

The Swissbit Cloud CSP-L utilizes the Enforcer R2 (by Private Machines) (hardware version: R2-XD1736NT, firmware version: 2.1.2 with libucl version: 2.7.1 and libdrbg Version: 2.0.0) for its operation. This platform complies with the aforementioned requirements.

1.2.9. Operational Environment of the TOE

The TOE is operated in a secure environment which is certified according to ISO/IEC 27001 and operates in conformance to an Information Security Management System (ISMS) with security level 'high' according to the Assumptions (A.SecComm) in **[PP-CSPL-V1.0]** and the Appendix: "Operational Requirements for CSPLight" in **[PP-SMAERS]**.

[1] Please note that (concerning **[PP-CSPL-V1.0]**) the update can also be seen as the end of life of the previous TOE as it will not longer be existing.

2. Conformance Claims

2.1. Common Criteria Conformance Claims

This ST claims conformance to CC version 3.1 revision 5.

Particularly, the conformance to CC Part 2 (security functional requirements) [CC2] is CC Part 2 extended, the conformance to CC Part 3 (security assurance requirements) [CC3] is CC Part 3 conformant.

2.2. Protection Profile Claims

This ST claims strict conformance to the following Protection Profiles:

- Base-PP: Protection Profile Cryptographic Service Provider Light (CSPLight) Version 1.0, BSI-CC-PP-0111-2019 [PP-CSPL-V1.0],
- PP-Module: CSPLight Time Stamp Service and Audit (PPM-TS-Au), Version 1.0, BSI-CC-PP-0112-2020 [PP-CSPL-TS-AU-V1.0], and
- PP-Module: CSPLight Clustering (PPM-Cl), Version 1.0, BSI-CC-PP-0113-2020 [PP-CSPL-Cluster-V1.0].

2.3. Package claim

This ST claims package-augmented conformance to EAL2 augmented with ALC_CMS.3 and ALC_LCD.1

2.4. Conformance Rationale

The TOE described in this Security Target provides cryptographic services for the protection of the confidentiality and the integrity of user data, and for entity authentication, which is defined as the TOE Type by [PP-CSPL-V1.0]. The two PP-Modules [PP-CSPL-TS-AU-V1.0]and [PP-CSPL-Cluster-V1.0] are used in this Security Target, which are consistent for the PP-Configuration.

All PPs listed above require strict conformance which is claimed by this Security Target.

3. Security Problem Definition

3.1. Introduction

3.1.1. Assets

The assets of the TOE are:

- Concerning **[PP-CSPL-V1.0]**:
  - user data, which integrity and confidentiality shall be protected,
  - cryptographic services and keys which shall be protected against unauthorized use or misuse, and which integrity shall be protected,
  - update code packages (UCP), which integrity and confidentiality shall be protected,
  - additional TSF-data (e.g. security flags), which integrity and/or confidentiality shall be protected, and
  - other TOE resources, which unauthorized use and misuse shall be prevented.
- Additional concerning **[PP-CSPL-TS-AU-V1.0]**
  - user data and time stamps, which integrity shall be protected, and
  - time services which time base shall be protected against manipulation.
- Additional concerning **[PP-CSPL-Cluster-V1.0]**
  - TSF data, the security attributes of the known users and the cryptographic keys with their security attributes transferred between master and slave.

3.1.2. Users

The TOE knows the external entities (users) as follows:

- Concerning **[PP-CSPL-V1.0]**:
  - human user communicating with the TOE for security management of the TOE,
  - application component using the cryptographic and other security services of the TOE and supporting the communication with remote entities (e. g. by providing certificates), and
  - remote entity exchanging user data and TSF data with the TOE over insecure media.
- Additional concerning **[PP-CSPL-TS-AU-V1.0]**
  - none

- Additional concerning [PP-CSPL-Cluster-V1.0]
  - cluster-CSPLight being another TOE instance in a cluster with the TOE.

3.1.3. Subjects

The TOE communicates with:

- Concerning [PP-CSPL-V1.0]:
  - human user through a secure channel,
  - application component through a secure channel, and
  - remote entities over a trusted channel using cryptographic mechanisms including mutual authentication.
- Additional concerning [PP-CSPL-TS-AU-V1.0]
  - none
- Additional concerning [PP-CSPL-Cluster-V1.0]
  - cluster-CSPLight in encrypted and integrity protected form.

The subjects as active entities in the TOE perform operations on objects. Objects obtain their associated security attributes from the authenticated users, or the security attributes are defined by default values.

3.1.4. Objects

The TSF operates user data objects and TSF data objects (i. e. passive entities, that contain or receive information, and upon which subjects perform operations) . User data objects are imported, used in cryptographic operation, temporarily stored, exported and destroyed after use. The update code packages are user data objects that are imported and stored in the TOE until they are used to create an updated version of the CSP. TSF data objects are created, temporarily or permanently stored, imported, exported and destroyed as objects of the security management. They may contain e. g. cryptographic keys with their security attributes, certificates, or authentication data records with authentication reference data of a user. Cryptographic keys are objects of the key management.

3.1.5. Security attributes

A Role is a set of certain access rights and permissions. By defining roles, and associating users with roles (“a user or a subject takes a role”) it is immediately clear, what access rights and permissions this user is granted.

The security attributes of users known to the TOE are stored in Authentication Data Records

containing

- *User identity (User-ID)*,
- *Authentication reference data*, and
- *Role*.

Passwords as Authentication Reference Data have the security attributes:

- *status*: values *initial password, operational password*, and
- *number of unsuccessful authentication attempts*.

Certificates contain security attributes of users including User Identity, a public key and security attributes of the key. If certificates are used as authentication reference data for cryptographic entity authentication mechanisms they may contain the Role of the entity.

The TOE knows at least the following roles that can be taken by a user or a subject[1]:

- *Unidentified User*: this role is associated with any user not (successfully) identified by the TOE. This role is assumed after start-up of the TOE. The TSF associated actions allowed for the Unidentified User are defined in SFR FIA_UID.1.
- *Unauthenticated User*: this role is associated with an identified user but not (successfully) authenticated user. The TSF associated actions allowed for the Unauthenticated User are defined in SFR FIA_UAU.1.
- *Administrator*: a successful authenticated user in this role is allowed to access the TOE in order to perform management functions. It is taken by a human user or a subject acting on behalf of a human user after successful authentication as an Administrator. The Administrator role is split into more detailed roles[2]:
    - *Crypto-Officer*: a role that is allowed to access the TOE in order to manage a cryptographic TSF.
    - *User Administrator*: a role that is allowed to access the TOE in order to manage users.
    - *Update Agent*: a role that is allowed to import and install update code packages.
    - *Auditor Manager*: role that is allowed to configure the audit functionality and read system audit logs (based on **[PP-CSPL-TS-AU-V1.0]**),
    - *Audit Log Receiver*: role that is allowed to read audit logs associated to their own keys (based on **[PP-CSPL-TS-AU-V1.0]**), and
    - *Timekeeper*: role that is allowed to adjust the internal time (based on **[PP-CSPL-TS-AU-V1.0]**).

- *Key Owner*: successful authenticated user allowed to perform cryptographic operation with their own keys. This role may be claimed by human user or an entity.
- *Application Component*: subjects in this role are allowed to use assigned security services of the TOE without being authenticated as a human user (e. g. exporting and importing of wrapped keys). This role may be assigned to an entity communicating through a physically separated secure channel or through a trusted channel (which requires assured identification of its end points).
- *Cluster-CSPLight*: another TOE sample in a cluster with the TOE with security attribute
  - *Master*, or
  - *Slave*.

  This role is bound to the communication through the trusted channel between cluster CSPLights established by the administrator.

The user uses authentication verification data to prove its identity to the TOE. The TSF uses Authentication reference data to verify the claimed identity of a user. The TSF supports

- human user authentication by knowledge, where the authentication verification data is a password and the authentication reference data is a password or an image of the password e. g. a salted hash value or a derived cryptographic key,
- human user authentication by possession of a token, or as user of a terminal by implementing user authentication by cryptographic entity authentication mechanisms, and
- cryptographic entity authentication mechanisms where the authentication verification data is a secret or private key and the authentication reference data is a secret or public key.

A human user may authenticate himself to the TOE, and the TOE authenticates itself to an external entity in charge of the authenticated authorized user.

The TOE is delivered with initial Authentication Data Records for Unidentified User, Unauthenticated User and administrator role(s). The Authentication Data Records for Unidentified User and Unauthenticated User have no Authentication Reference Data. The roles are not exclusive, i. e. a user or subject may be in more then one role, e. g. a human user may claim the Crypto-Officer and Key Owner role at the same time. The SFR may define limitation on roles (especially combinations of roles) a user may be associated with.

Cryptographic keys **have** at least the security attributes:

- *Key identity*, i.e. an attribute that uniquely identifies the key,
- *Key Owner*, i. e. the identity of the owner this key is assigned to,

- *Key type*, i. e. whether the key is as secret key, a private key, or a public key,
- *Key usage type*, an attribute that identifies the cryptographic mechanism or services the key can be used for. For example, a private signature key may be used by a digital signature-creation mechanism (cf. FCS_COP.1/CDS-ECDSA or FCS_COP.1/CDS-RSA); and depending on the corresponding certificate (cf. FDP_DAU.2/Sig) be used for signing data, or for device-attestation.
- *Key access control attributes*, i. e. a list of combinations of the identity of the user, the role for which the user is authenticated, and the allowed key management functions or cryptographic operations. This includes that

  - the *import* of the key is allowed or forbidden,
  - the *export* of the key is allowed or forbidden, and
  - *clustering*: transfer of the key in a cluster of TOE samples (i.e. export by TOE as Master-CSPLight and import by TOE as Slave-CSPLight) is allowed or forbidden,

  and **may** have the security attributes:

- *key validity time period*, i. e. the time period for operational use of the key: The key must not be used before or after a defined time slot. Note that exceptions could be required: For example it might be required that an expired root certificate can be updated with a valid link certificate to a new valid root certificate.
- *key usage counter*, i. e. the number of operations performed with this key – for example the current number of signatures created with a private signature key.

Update Code Packages (UCPs) have at least the security attributes;

- *issuer* of the UCP, and
- *version number* of the UCP.

3.2. Threats

All listed threats are derived from **[PP-CSPL-V1.0]**.

T.DataCompr: Compromise of communication data

An unauthorized entity gets knowledge of information that are stored on media controlled by the TSF, or an unauthorized entity gets knowledge of information that are transferred between the TOE and an authenticated external entity.

T.DataMani: Unauthorized generation or manipulation of communication data

An unauthorized entity generates or manipulates user data that are stored on media controlled by the

TSF or transferred between the TOE and an authenticated external entity, and manipulates such data so that they are accepted as valid by the recipient.

T.Masqu: Masquerade authorized user

A threat agent masquerades as an authorized entity in order to gain unauthorized access to user data, TSF data, or TOE resources.

T.ServAcc: Unauthorized access to TOE security services

An attacker gets unauthorized access to security services of the TOE.

T.PhysAttack: Physical attacks

An attacker gets physical access to the underlying hardware platform that the TOE is running on and may (1) disclose or manipulate user data under TSF control and TSF data, and (2) affect TSF by (a) physical probing and manipulation, (b) applying environmental stress or (c) exploiting information leakage from the TOE.

T.FaUpD: Faulty Update Code Package

An unauthorized entity provides and installs a faulty update code package. Thus attacks against the integrity of the TSF implementation, and against the confidentiality and integrity of user data and TSF data becomes possible.

3.3. Organizational Security Policies

OSPs based on [PP-CSPL-V1.0]

OSP.SecCryM: Secure cryptographic mechanisms

The TOE uses only secure cryptographic mechanisms as confirmed by the certification body for the specified TSF, the assurance security requirements and the operational environment.

OSP.SecService: Security services of the TOE

The TOE provides security services to the authorized users for encryption and decryption of user data, authentication prove and verification of user data, entity authentication to external entities including attestation, trusted channels and random bit generation.

OSP.KeyMan: Key Management

The key management ensures the integrity of all cryptographic keys and the confidentiality of all secret or private keys over the whole life cycle. The life-cycle comprises key generation, storage, distribution, application, archival and deletion. The cryptographic keys and cryptographic key components shall be generated, operated and managed by secure cryptographic mechanisms, assigned to the secure cryptographic mechanisms they are intended to be used with, and to the entities authorized for their

use.

OSP.TC: Trust centre

Trust centres provide secure certificates for trustworthy certificate holders with correct security attributes. The TOE uses certificates for identification and authentication of users, access control and secure use of security services of the TOE. In particular, this includes key management and attestation.

OSP.Update: Authorized Update Code Packages

Update Code Packages are delivered in encrypted form, and are signed by the authorized issuer. The TOE verifies the authenticity of the received Update Code Package using the CSP before storing any update data in the TOE. The TOE restricts the storage of authentic Update Code Package to authorized users.

Additional OSPs based on [PP-CSPL-TS-AU-V1.0]

OSP.Audit: Audit for key management and cryptographic operations

The TOE provides security auditing related to activities controlled by the TSF and security critical events. The security auditing provides evidence to make users responsible for actions they are authorized for and to protect users against unwarranted accusation. The Administrator is allowed to select auditable events.

OSP.TimeService: Time Service and Time stamp service

The TOE provides non-cryptographic time service and cryptographic time stamp service for user data and TSF data. The time stamp service provides evidence that user data were presented to the TSF and exported audit data were generated at certain point in time and in a verifiable sequence.

Additional OSPs based on [PP-CSPL-Cluster-V1.0]

OSP.Cluster: Cluster of TOE samples

The administrator establishes and manages a cluster of multiple TOE samples for secure transfer of the security attributes of the known users and the cryptographic keys as necessary for scalability of performance and availability of security services.

3.4. Assumptions

Assumptions based on [PP-CSPL-V1.0]

A.SecComm: Secure communication

Remote entities support trusted channels by cryptographic mechanisms. The operational environment shall protect the local communication channels by trusted channels using cryptographic mechanisms, or

by secure channels using non-cryptographic security measures. The operational environment must be subject to a security audit that verifies that the communication between the TOE and the application is indeed protected.

Assumptions based on [PP-CSPL-TS-AU-V1.0]

none

Assumptions based on [PP-CSPL-Cluster-V1.0]

A.ClusterAppl: Cluster management by application

The application using the security services of the TOE transfers security attributes of the known users and cryptographic keys with their security attributes between Master-CSPLight and Slave-CSPLights as necessary for scalability of performance and availability of security services.

[1] All roles listed are derived from [PP-CSPL-V1.0]. Exceptions are marked inline

[2] The SFRs uses the general term Administrator or a selection between Administrator role and these detailed roles in case they are supported by the TOE and separation of duties is appropriate.

4. Security Objectives

4.1. Security Objectives of the TOE

SOs based on [PP-CSPL-V1.0]

O.AuthentTOE: Authentication of the TOE to external entities

The TOE authenticates itself in charge of authorized users to external entities by means of secure cryptographic entity authentication and attestation.

O.Enc: Confidentiality of user data by encryption and decryption

The TOE provides secure encryption and decryption as security services for the users to protect the confidentiality of exported or imported user data, or user data stored on media that is within the scope of control of the TSF.

O.DataAuth: Data authentication by cryptographic mechanisms

The TOE provides secure symmetric and asymmetric data authentication mechanisms as security services for the users to protect the integrity and authenticity of user data.

O.RBGS: Random bit generation service

The TOE provide cryptographically secure random bit generation for the users.

O.TChann: Trusted channel

The TSF provides trusted channel functionality using secure cryptographic mechanisms for the communication between the TSF and external entities. The TOE provides authentication of all communication end points, and ensures the confidentiality and integrity of the communication data that are exchanged through the trusted channel.

Note that the TSF can establish the trusted channel by means of secure cryptographic mechanisms only if the other external entity supports these secure cryptographic mechanisms as well. If the trusted channel cannot be established by means of secure cryptographic mechanisms – i.e. due to missing security functionality on the user side – then the operational environment shall provide a secure channel that protects the communication by non-cryptographic security mechanisms, cf. **A.SecComm** and **OE.SecComm**.

O.I&A: Identification and authentication of users

The TOE shall uniquely identify users and verify the claimed identity of the user before providing access to any controlled resources; The TOE shall authenticate IT entities using secure cryptographic mechanisms.

O.AccCtrl: Access control

The TOE provides access control of security services, operations on user data, and management of TSF and TSF data.

O.SecMan: Security management

The TOE provides security management of users, TSF, TSF data and cryptographic keys by means of secure cryptographic mechanisms and certificates. The TSF generates, derives, agrees, imports and exports cryptographic keys as a security service for users and for internal use. The TSF shall destruct unprotected secret or private keys in such a way that any previous information content of the resource is made unavailable.

O.TST: Self-test

The TSF performs self-tests during initial start-up, and after power-on. The TSF enters a secure state if the self-test fails or if attacks are detected. It relies on the underlying hardware platform and operating system (cf. OE.SecPlatform) to implement this functionality.

O.SecUpCP: Secure import of Update Code Packages

The TSF verifies the authenticity of a received encrypted Update Code Package, decrypts the Update Code Package if it is verified to be authentic, and installs it after verifying that it is suitable for the TOE and does not downgrade the TOE's firmware to a previous version.

Additional SOs based on **[PP-CSPL-TS-AU-V1.0]**

O.Audit: Audit

The TSF provides security auditing of selected user activities controlled by the TSF and security critical events. The Administrator is allowed to select auditable events, to manage the audit functionality and the export of audit records.

O.TimeService: Time services

The TOE provide an internal time service and time stamp service for the user.

Additional SOs based on **[PP-CSPL-Cluster-V1.0]**

O.Cluster: Cluster

The TSF supports cluster of TOE samples by secure transfer of the security attributes of the known users and the cryptographic keys with their security attributes between Master-CSPLight and Slave-CSPLights in encrypted and integrity protected form. **In addition, the TSF also transfers audit logs and system logs and additional attributes of users (i.e. their roles).**

ST Application Note 1 The scope of O.Cluster has been extended by the ST author. This allows to ensure that a slave TOE will be able to replace a master TOE without needing further information.

4.2. Security Objectives of the Operational Environment

SOsEnv based on [PP-CSPL-V1.0]

OE.CommInf: Communication infrastructure

The operational environment shall provide a public key infrastructure for entities in the relevant communication networks. Trust centres must generate secure certificates for trustworthy certificate holders with correct security attributes. They must distribute their certificate signing public key securely such that a verification of the digital signature of the generated certificates is possible. Trust centres should further operate a directory service for dissemination of certificates and provision of revocation status information of certificates.

OE.AppComp: Support of the Application component

The Application component supports the TOE for communication with users and trust centres.

OE.SecManag: Security management

The operational environment shall implement appropriate security management functionality for secure use of the TOE. This includes user management as well as key management. It ensures secure key management outside of the TOE and uses the trust centre's services to determine the validity of certificates. Cryptographic keys and cryptographic key components shall be assigned to the secure cryptographic mechanisms they are intended to be used with, and to the entities authorized for their use.

OE.SecComm: Protection of communication channel

Remote entities shall support establishing trusted channels with the TOE by using cryptographic mechanisms. The operational environment shall protect the local communication channels by trusted channels using cryptographic mechanisms, or by secure channels using non-cryptographic security measures. In the latter case, the operational environment must be subject to a security audit that verifies that the communication between the TOE and the application is indeed protected.

OE.SUCP: Signed Update Code Packages

The secure Update Code Package is delivered in encrypted form and signed by the authorized issuer together with its security attributes.

OE.SecPlatform: Secure Hardware Platform

The TOE runs on a secure hardware platform. The hardware platform and its operating system support the implementation of the TSF; this in particular includes the protection of the confidentiality and integrity of user data, TSF data and its correct operation against physical attacks and environmental stress.

SOsEnv based on [PP-CSPL-TS-AU-V1.0]

OE.Audit: Review and availability of audit records

The Administrator shall ensure the regular audit review and the availability of exported audit records.

OE.TimeSource: External time source

The operational environment provides reliable external time source for the adjustment of the TOE internal time source.

SOsEnv based on [PP-CSPL-Cluster-V1.0]

OE.ClusterCtrl: Control of the cluster

The administrator establishes and manages a cluster only of trustworthy samples of the TOE as necessary for scalability of performance and availability of security services.

OE.TSFdataTrans: Transfer of TSF data within the CSPLight cluster

The administrator and the application using the security services of the TOE, transfer the security attributes of the known users and the cryptographic keys with their security attributes between Master-CSPLight and SlaveCSPLights as necessary for scalability of performance and availability of security services.

4.3. Security Objectives Rationale

The following table traces the security objectives for the TOE back to threats countered by that security objective and OSPs enforced by that security objective, and the security objective for the operational environment back to threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective.

4.3.1. Security Objectives Rational based on PP-CSPL-V1.0

Table 1. Security Objectives Rationale based on [PP-CSPL-V1.0]

	T.DataCompr	T.DataManip	T.Masqu	T.ServAcc	T.PhysAttack	T.FaUpD	OSP.SecCryM	OSP.SecService	OSP.KeyMan	OSP.TC	OSP.Update	A.SecComm
O.AccCtrl				X								
O.AuthentTOE							X	X				
O.DataAuth		X					X	X				
O.Enc	X						X	X				
O.I&A			X	X			X	X				
O.RBGS							X	X				
O.SecMan			X				X		X	X		
O.SecUpCP						X						
O.TChann	X	X	X	X			X	X				
O.TST					X							
OE.AppComp	X	X		X				X	X	X		
OE.SecComm	X	X		X								X
OE.SecManag			X					X	X			
OE.SUCP						X					X	
OE.SecPlatform					X							

The following part of the chapter demonstrate that the security objectives counter all threats and enforce all OSPs, and the security objectives for the operational environment uphold all assumptions.

The threat T.DataCompr**: "Compromise of communication data"** is countered by the following security objectives for the TOE and the operational environment:

- O.Enc requires the TOE to provide encryption and decryption as a security service for the users to protect the confidentiality of user data,
- O.TChann requires the TOE to support establishing a trusted channel between the TSF and the application component, between the TSF and other users, and between the application component and other users. The trusted channel ensures authentication of all communication end points, and protected communication for the confidentiality and integrity of the communication and to prevent misuse of sessions of authorized users.
- OE.AppComp requires the application component to support the TOE for communication with

users and trust centres.

- **OE.CommInf** requires the operational environment to provide a communication infrastructure; especially w.r.t. trust centre services.
- **OE.SecComm** requires the operational environment to protect the confidentiality and integrity of communication over local communication channels by physical security measures, and requires remote entities to support trusted channels by means of cryptographic mechanisms. If a trusted channel cannot be established due to missing security functionality of the application component, the operational environment shall protect the communication, cf. **A.SecComm** and **OE.SecComm**. Note that **OE.SecComm** requires measures that the operational environment must be subject to a security audit that verifies that the communication between the TOE and the application is indeed protected.

The threat **T.DataMani**: **"Unauthorized generation or manipulation of communication data"** is countered by the security objectives for the TOE and the operational environment:

- **O.DataAuth** requires the TOE to provide symmetric and asymmetric data authentication mechanisms as a security service for the users to protect the integrity and authenticity of user data.
- **O.TChann** requires the TOE to support trusted channels for the authentication of all communication end points, for the protected communication with the application component, and for other users. This ensures the confidentiality and integrity of the communication between the TOE and the other parties and prevents misuse of sessions of authorized users.
- **OE.AppComp** requires the application component to support the TOE for communication with users and trust centres.
- **OE.CommInf** requires the operational environment to provide trust centre services and securely distribute root public keys.
- **OE.SecComm** requires the operational environment to protect the confidentiality and integrity of communication with the TOE. Remote entities shall support trusted channels with the TOE using cryptographic mechanisms. The operational environment shall protect local communication channels by trusted channels using cryptographic mechanisms, or by secure channels using non-cryptographic security measures.

The threat **T.Masqu**: **"Masquerade authorized user"** is countered by the security objectives for the TOE and the operational environment:

- **O.I&A** requires the TSF to identify uniquely users and verify the claimed identity of the user before providing access to any controlled resources.
- **O.TChann** requires the TSF to provide authentication of all communication end points of the

trusted channel.

- **O.SecMan** requires the TSF to provides security management of users, TSF, TSF data and cryptographic keys by means of secure cryptographic mechanisms and certificates.
- **OE.SecManag** requires the operational environment to implement appropriate security management functionality for the secure use of the TOE. This includes user management.

The threat **T.ServAcc: "Unauthorized access to TOE security services"** is countered by the security objectives for the TOE and the operational environment:

- **O.I&A** requires the TSF to uniquely identify users and to authenticate users before providing access to any controlled resources.
- **O.AccCtrl** requires the TSF to control access of security services, operations on user data, and management of TSF and TSF data.
- **O.TChann** requires mutual authentication of the external entity and the TOE, and the authentication of communicated data between them to prevent misuse of the communication with external entities. The operational environment is required by **OE.SecComm** to ensure that a secure channel is available if a trusted channel cannot be established.
- The operational environment **OE.CommInf** requires the provision of a public key infrastructure for entity authentication. **OE.AppComp** requires the application to support the communication with trust centres.

The threat **T.PhysAttack: "Physical attacks"** is countered by the next security objectives:

- **OE.SecPlatform** ensures that the TOE runs on a secure hardware platform and operating system that provides protection against physical attacks.
- As means to ensure robustness against perturbation **O.TST** requires the TSF to perform self-tests and to enter a secure state if the self-test fails or attacks are detected.

The threat **T.FaUpD: "Faulty Update Code Package"** is directly countered by the security objective **O.SecUpCP** verifying the authenticity of UCP under the condition that trustworthy UCPs are signed as required by **OE.SUCP**

- **O.SecUpCP**: "Secure import of Update Code Package" requires the TOE to verify the authenticity of received encrypted Update Code Packages before decrypting and storing an authentic Update Code Package.
- **OE.SUCP**: "Signed Update Code Packages" requires the Issuer to sign both the secure Update Code packages as well as its security attributes.

The organizational security policy **OSP.SecCryM: "Secure cryptographic mechanisms"** is

implemented by means of secure cryptographic mechanisms required in

- **O.I&A**: "Identification and authentication of users" and **O.AuthentTOE** "Authentication of the TOE to external entities" which require secure entity authentication of users and the TOE,
- **O.Enc**: "Confidentiality of user data by means of encryption and decryption" and O.DataAuth "Data authentication by cryptographic mechanisms" require secure cryptographic mechanisms for protection of the confidentiality and integrity of user data,
- **O.TChann**: "Trusted channel" require secure cryptographic mechanisms for entity authentication of users and the TOE, and the protection of confidentiality and integrity of communication data.
- **O.RBGS**: "Random bit generation service" requires the TOE to provide a cryptographically secure random bit generation service for the users.
- **O.SecMan**: "Security management" requires secure management of TSF data and cryptographic keys by means of secure cryptographic mechanisms and certificates.

The organizational security policy **OSP.SecService: "Security services of the TOE"** is directly implemented by security objectives for the TOE **O.Enc**: "Confidentiality of user data by means of encryption and decryption", **O.DataAuth**: "Data authentication by cryptographic mechanisms", **O.I&A**: "Identification and authentication of users", **O.AuthentTOE**: "Authentication of the TOE to external entities", **O.TChann**: "Trusted channel" and **O.RBGS**: "Random bit generation service", which require the TSF to provide cryptographic security services for the user. The **OSP.SecService** is supported by **OE.CommInf**: "Communication infrastructure" and **OE.SecManag**: "Security management" which provide the necessary measures for the secure use of these services.

The organizational security policy **OSP.KeyMan: "Key Management"** is directly implemented by **O.SecMan**: "Security management" and supported by trust centre services according to **OE.CommInf**: "Communication infrastructure" and **OE.SecManag**: "Security management".

The organizational security policy **OSP.TC: "Trust centre"** is implemented by security objectives for the TOE and the operational environment:

- **O.SecMan**: "Security management" uses certificates for secure management of users, TSF, TSF data and cryptographic keys.
- **OE.CommInf**: "Communication infrastructure" requires trust centres to generate secure certificates for trustworthy certificate holders with correct security attributes, and to distribute certificates and revocation status information.
- **OE.AppComp**: "Support of the Application component" requires the Application component to support the TOE for the communication with trust centres.

The organizational security policy OSP.Update: **“Authorized Update Code Packages”** is implemented directly by the security objectives for the TOE O.SecUpCP and the operational environment OE.SUCP.

The assumption A.SecComm: **“Secure communication”** assumes that the operational environment protects the confidentiality and integrity of communication data and ensures reliable identification of its end points. The security objective for the operational environment OE.SecComm require the operational environment to protect local communication physically or via trusted channel, and remote entities to support trusted channels using cryptographic mechanisms.

4.3.2. Security Objectives Rational based on PP-CSPL-TS-AU-V1.0

The following table traces the security objectives for the TOE back to threats countered by that security objective and OSPs enforced by that security objective, and the security objective for the operational environment back to threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective.

Table 2. Security Objectives Rationale based on [PP-CSPL-TS-AU-V1.0]

	OSP.Audit	OSP.TimeService
O.Audit	X	
O.TimeService		X
OE.Audit	X	
OE.TimeSource		X

The following part of the chapter demonstrates that the security objectives counter all threats and enforce all OSPs, and the security objectives for the operational environment uphold all assumptions.

The organizational security policy OSP.Audit: **“Audit for key management and cryptographic operations”** is directly implemented by

- The security objective for the TOE O.Audit requiring security auditing and
- The security objective for the operational environment OE.Audit requiring the regular audit review and the availability of exported audit records.

The organizational security policy OSP.TimeService: **“Time Service and Time stamp service”** is directly implemented by

- The security objective for the TOE O.TimeService: “Time services ” requiring the TOE to

provide an internal time service and time stamp service for the user, and

- The security objective for the operational environment **OE.TimeSource**: "External time source" requiring the operational environment to provide reliable external time stamps for adjustment of TOE internal time source.

4.3.3. Security Objectives Rational based on PP-CSPL-Cluster-V1.0

The following table traces the security objectives for the TOE back the OSPs enforced by that security objective, and the security objective for the operational environment back OSPs enforced by that security objective, and assumptions upheld by that security objective. Note the **OSP.SecCryM**: "Secure cryptographic mechanisms" defined in the **[PP-CSPL-V1.0]**.

Table 3. Security Objectives Rationale based on **[PP-CSPL-Cluster-V1.0]**

	OSP.SecCryM	OSP.Cluster	A.ClusterAppl
O.Cluster	X	X	
OE.ClusterCtrl		X	
OE.TSFdataTrans		X	X

The following part of the chapter demonstrate that the security objectives enforce all OSPs, and the security objectives for the operational environment uphold all assumptions.

The organizational security policy **OSP.SecCryM: "Secure cryptographic mechanisms"** defined in the **[PP-CSPL-V1.0]** is implemented by means of secure cryptographic mechanisms required in

- **O.Cluster**: "Cluster" requiring secure transfer in encrypted and integrity protected form of the security attributes of the known users and the cryptographic keys with their security attributes between MasterCSPLight and Slave-CSPLights.

The organizational security policy **OSP.Cluster: "Cluster of TOE samples"** is implemented by security objectives for the TOE and the operational environment:

- **O.Cluster** requiring support for cluster of TOE samples as CSPLights with distribution of Authentication Data Records and cryptographic keys between Master-CSPLight and Slave-CSPLights through a trusted channel keeping the confidentiality and integrity of the security attributes of the known users and of the cryptographic keys with their security attributes.
- **OE.ClusterCtrl** requiring administrator to build a cluster only of trustworthy samples of the TOE as needed for scalability of performance and availability of security services.
- **OE.TSFdataTrans** requires the administrator and the application using the security services of

the TOE transfer security attributes of the known users and cryptographic keys with their security attributes between Master-CSPLight and Slave-CSPLights as necessary for scalability of performance and availability of security services.

The assumption **A.ClusterAppl** is directly ensured by **OE.TSFdataTrans**.

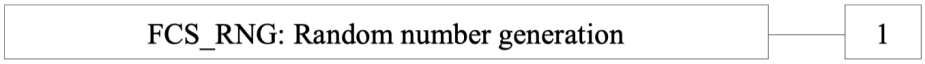
5. Extended Component Definition

5.1. Generation of random numbers (FCS_RNG)

5.1.1. Family Behaviour

This family defines quality requirements for the generation of random numbers that are intended to be used for cryptographic purposes.

5.1.2. Component levelling



FCS_RNG.1 Generation of random numbers, requires that the random number generator implements defined security capabilities and that the random numbers meet a defined quality metric.

5.1.3. Management: FCS_RNG.1

There are no management activities foreseen.

5.1.4. Audit: FCS_RNG.1

There are no auditable events foreseen.

5.1.5. FCS_RNG.1: Random number generation

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 random numbers that meet [assignment: *a defined quality metric*].

5.2. Cryptographic key derivation (FCS_CKM.5)

This chapter describes a component of the family Cryptographic key management (FCS_CKM) for key derivation as process by which one or more keys are calculated from either a pre-shared key or a shared

secret and other information. Key derivation is the deterministic repeatable process by which one or more keys are calculated from both a pre-shared key or shared secret, and other information, while key generation required by FCS_CKM.1 uses internal random numbers.

The component FCS_CKM.5 is on the same level as the other components of the family FCS_CKM.

5.2.1. Management: FCS_CKM.5

There are no management activities foreseen.

5.2.2. Audit: FCS_CKM.5

The following actions should be auditable if FAU_GEN "Security audit data generation" is included in the ST:

a. Minimal: Success and failure of the activity.

b. Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys).

FCS_CKM.5 Requires the TOE to provide key derivation.

5.2.3. FCS_CKM.5: Cryptographic key derivation

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.5.1	The TSF shall derive cryptographic keys [assignment: *key type*] from [assignment: *input parameters*] in accordance with a specified cryptographic key derivation algorithm [assignment: *cryptographic key derivation algorithm*] and specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].

5.3. Authentication Proof of Identity (FIA_API)

To describe the IT security functional requirements of the TOE a sensitive family (FIA_API) of the Class FIA (Identification and authentication) is defined here. This family describes the functional requirements for the proof of the claimed identity for the authentication verification by an external entity where the other families of the class FIA address the verification of the identity of an external entity.

5.3.1. Family Behavior

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

5.3.2. Component levelling

FIA_API.1 Authentication Proof of Identity, provides prove of the identity of the TOE to an external entity.

5.3.3. Management: FIA_API.1

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

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

5.3.4. Audit: FIA_API.1

There are no auditable events foreseen.

5.3.5. FIA_APL.1 Authentication Proof of Identity

Hierarchical to	No other components.
Dependencies	No dependencies.
FIA_APL.1.1	The TSF shall provide a [assignment: *authentication mechanism*] to prove the identity of the [assignment: *object, authorized user or role*] to an external entity.

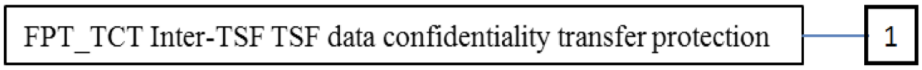
5.4. Inter-TSF TSF data confidentiality transfer protection (FPT_TCT)

This section describes the functional requirements for confidentiality protection of inter-TSF transfer of TSF data. The family is similar to the family Basic data exchange confidentiality (FDP_UCT) which defines functional requirements for confidentiality protection of exchanged user data.

5.4.1. Family Behaviour

This family requires confidentiality protection of exchanged TSF data.

5.4.2. Component levelling



FPT_TCT.1 requires the TOE to protect the confidentiality of information in exchanged the TSF data.

5.4.3. Management: FPT_TCT.1

There are no management activities foreseen.

5.4.4. Audit: FPT_TCT.1

There are no auditable events foreseen.

5.4.5. FPT_TCT.1 TSF data confidentiality transfer protection

Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]
FPT_TCT.1.1	The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] by providing the ability to [selection: *transmit, receive, transmit and receive*] TSF data in a manner protected from unauthorised disclosure.

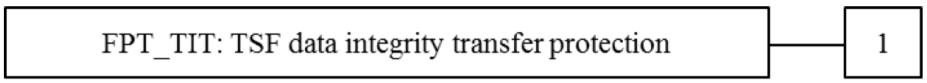
5.5. Inter-TSF TSF data integrity transfer protection (FPT_TIT)

This section describes the functional requirements for integrity protection of TSF data exchanged with another trusted IT product. The family is similar to the family Inter-TSF user data integrity transfer protection (FDP_UIT) which defines functional requirements for integrity protection of exchanged user data.

5.5.1. Family Behavior

This family requires integrity protection of exchanged TSF data.

5.5.2. Component levelling



FPT_TIT.1 requires the TOE to protect the integrity of information in exchanged the TSF data.

5.5.3. Management: FPT_TIT.1

There are no management activities foreseen.

5.5.4. Audit: FPT_TIT.1

There are no auditable events foreseen.

5.5.5. FPT_TIT.1 TSF data integrity transfer protection

Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]
FPT_TIT.1.1	The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] to [selection: *transmit, receive, transmit and receive*] TSF data in a manner protected from [selection: *modification, deletion, insertion, replay*] errors.
FPT_TIT.1.2	The TSF shall be able to determine on receipt of TSF data, whether [selection: *modification, deletion, insertion, replay*] has occurred.

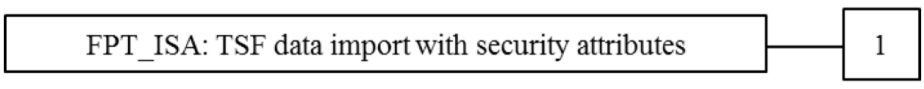
5.6. TSF data import with security attributes (FPT_ISA)

This section describes the functional requirements for TSF data import with security attributes from another trusted IT product. The family is similar to the family Import from outside of the TOE (FDP_ITC) which defines functional requirements for user data import with security attributes.

5.6.1. Family Behavior

This family requires TSF data import with security attributes.

5.6.2. Component levelling



FPT_ISA.1 requires the TOE to import TSF data with security attributes.

5.6.3. Management: FPT_ISA.1

There are no management activities foreseen.

5.6.4. Audit: FPT_ISA.1

There are no auditable events foreseen.

5.6.5. FPT_ISA.1 Import of TSF data with security attributes

Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency
FPT_ISA.1.1	The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] when importing TSF data, controlled under the SFP, from outside of the TOE.
FPT_ISA.1.2	The TSF shall use the security attributes associated with the imported TSF data.
FPT_ISA.1.3	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the TSF data received.
FPT_ISA.1.4	The TSF shall ensure that interpretation of the security attributes of the imported TSF data is as intended by the source of the TSF data.
FPT_ISA.1.5	The TSF shall enforce the following rules when importing TSF data controlled under the SFP from outside the TOE: [assignment: *additional importation control rules*]

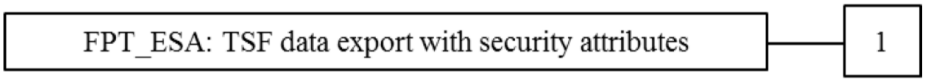
5.7. TSF data export with security attributes (FPT_ESA)

This section describes the functional requirements for TSF data export with security attributes to another trusted IT product. The family is similar to the family Export to outside of the TOE (FDP_ETC) which defines functional requirements for user data export with security attributes.

5.7.1. Family Behavior

This family requires TSF data export with security attributes.

5.7.2. Component levelling



FPT_ESA.1 requires the TOE to export TSF data with security attributes.

5.7.3. Management: FPT_ESA.1

There are no management activities foreseen.

5.7.4. Audit: FPT_ESA.1

There are no auditable events foreseen.

5.7.5. FPT_ESA.1 Export of TSF data with security attributes

Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency
FPT_ESA.1.1	The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] when exporting TSF data, controlled under the SFP(s), outside of the TOE.
FPT_ESA.1.2	The TSF shall export the TSF data with the TSF data's associated security attributes.
FPT_ESA.1.3	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported TSF data.
FPT_ESA.1.4	The TSF shall enforce the following rules when TSF data is exported from the TOE: [assignment: *additional exportation control rules*].

6. Security Requirements

The CC 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 PPs to which this ST claims conformance to.

All operations performed by the PP or ST author(s) are following the rule-set defined below. To differentiate between operations performed by the PP and ST author respectively, the latter ones are highlighted additionally in another font-colour.

The **refinement** operation is used to add detail to a requirement, and thus further restricts a requirement. Refinement of security requirements is (i) denoted by the word "refinement" in **bold** text and the added/changed words are in bold text, or (ii) directly included in the requirement text as **bold** text. In cases where words from a CC requirement component were deleted, these words are ~~crossed out~~.

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/ST authors are denoted as underlined text and the original text of the component is given by a footnote.

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as e.g. the length of a password. Assignments that have been made by the PP/*ST* authors are denoted by showing as text in italics and the original text of the component is given by a footnote.

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.

6.1. Security Functional Requirements based on PP-CSPL-V1.0

The TOE provides cryptographic security services for encryption and decryption of user data, entity authentication of external entities and to external entities, authentication prove and verification of user data, trusted channel establishment and random number generation.

The TOE enforces the Cryptographic Operation SFP for protection of theses cryptographic services. Corresponding Subjects, objects, and operations are defined in the SFRs **FDP_ACC.1/Oper** and **FDP_ACF.1/Oper**.

The TOE provides hybrid encryption and decryption combined with data integrity mechanisms for the cipher text as a cryptographic security service of the TOE. The encryption **FCS_COP.1/HEM** combines the generation of a data encryption key and message authentication code (MAC) key, the asymmetric encryption of the data encryption key with an asymmetric key encryption key, cf.

FCS_CKM.1/ECKA-EG, **FCS_CKM.1/RSA**, and the symmetric encryption of the data with the data encryption key and data integrity mechanism with MAC calculation for the cipher text. The receiver reconstructs the data encryption key and the MAC key, cf. **FCS_CKM.5/ECKA-EG**, calculates the MAC for the cipher text and compares it with the received MAC. If the integrity of the cipher text is determined, then the receiver decrypts the cipher text with the data decryption key, cf. **FCS_COP.1/HDM**.

In general, authentication is the provision of assurance of the claimed identity of an entity. The TOE authenticates human users by passwords, cf. **FIA_UAU.5.1** clause 1 (1-Factor Authentication). But a human user may also authenticate himself to a token and the token authenticates to the TOE (2-Factor Authentication). Cryptographic authentication mechanisms allow an entity to prove its identity or the origin of its data to a verifying entity by demonstrating its knowledge of a secret. The entity authentication is required by **FIA_UAU.5.1** clauses (2) to (6). Chapter **Authentication Proof of Identity (FIA_API)** describes SFRs for the authentication of the TOE to external entities required by the SFR FIA_API.1. This authentication may include attestation of the TOE as a genuine TOE sample, cf. 6.1.4. The authentication may be mutual as required for trusted channels in chapter 6.1.5.

Protocols may use symmetric cryptographic algorithms, where the proving and the verifying entity using the same secret key may demonstrate that the proving entity belongs to a group of entities sharing this key, e.g. the sender and receiver (cf. **FTP_ITC.1**, **FCS_COP.1/TCM**). In case of asymmetric entity authentication mechanisms, the proving entity uses a private key, and the verifying entity uses the corresponding public key, where the latter is usually closely linked to the claimed identity by means of a certificate. Depending on the security attributes of the cryptographic keys – e.g. encoded in the certificate (cf. **FPT_ISA.1/Cert**) –, the same cryptographic mechanisms for digital signature generation (FCS_COP.1/CDS-**) and signature verification (cf. FCS_COP.1/VDS-**) may be used for entity authentication, data authentication and non-repudiation as well.

A trusted channel requires mutual authentication of both endpoints with a key exchange of a key agreement, and the protection of confidentiality by encryption and cryptographic data integrity protection.

The TSF provide security management for user and TSF data, including cryptographic keys. Key management comprises administration and use of keying material in accordance with a security policy. This includes generation, derivation, registration, certification, deregistration, distribution, installation, storage, archival, revocation and destruction of keying material. The key management functionality of the TOE supports the generation, derivation, export, import, storage and destruction of cryptographic keys. The cryptographic keys are managed together with their security attributes.

The TOE enforces the Key Management SFP to protect all cryptographic keys (as data objects of TSF data) and key management services (as operation, cf. to SFR of the FMT class) provided for Administrators, Crypto-Officers, and Key Owners. Note that the cryptographic keys will be used for

cryptographic operations under the Cryptographic Operation SFP as well.

The subjects, objects and operations of the Update SFP are defined in the SFR **FDP_ACC.1/UCP** and **FDP_ACF.1/UCP**.

The SFRs for cryptographic mechanisms based on elliptic curves refer to the following table for selection of curves, key sizes and standards.

Table 4. Elliptic curves, key sizes and standards

Elliptic curve	Key size	Standard
brainpoolP256r1	256 bits	[RFC-5639] , [BSI-TR-03111], section 4.1.3
brainpoolP384r1	384 bits	[RFC-5639], [BSI-TR-03111], section 4.1.3
brainpoolP512r1	512 bits	[RFC-5639], [BSI-TR-03111], section 4.1.3
Curve P-256	256 bits	[FIPS_186-4] B.4 and D.1.2.3
Curve P-521	521 bits	[FIPS_186-4] B.4 and D.1.2.5

For Diffie-Hellman key exchange refer to the following groups

Table 5. Recommended groups for the Diffie-Hellman key exchange (cmp. [PP-CSPL-V1.0], Table 3)

Name	IANA no.	Specified in
256-bit random ECP group	19	[RFC-5903]
384-bit random ECP group	20	[RFC-5903]
521-bit random ECP group	21	[RFC-5903]
brainpoolP256r1	28	[RFC-6954]
brainpoolP384r1	29	[RFC-6954]
brainpoolP512r1	30	[RFC-6954]

6.1.1. Key management

Management of security attributes

FDP_ACC.1/KM Subset access control – Cryptographic operation

Hierarchical to	No other components.
Dependencies	FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/KM The TSF shall enforce the *Key Management SFP* [1] on

1. *subjects: Crypto-Officer* [2]*, KeyOwner;*
2. *objects: operational cryptographic keys;*
3. *operations: key generation, key derivation, key import, key export, key destruction* [3].

FMT_MSA.1/KM Management of security attributes – Key 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/KM The TSF shall enforce the Key Management SFP and Cryptographic Operation SFP [4] to restrict the ability to

1. set and change default values for [5] the security attributes Identity of the key, Key owner of the key, Key type, Key usage type, Key access control attributes, Key validity time period [6] to **no role** [7],
2. **modify or delete** [8] **the security attributes Identity of the key, Key owner, Key type, Key usage type, Key validity time period of an existing key** [9] **to none** [10]**,**
3. **modify independent on key usage** [11]**the security attributes Key usage counter of an existing key** [12] **to none** [13] **.**
4. **modify** [14] **the security attributes Key access control attribute of an existing key** [15] **to none** [16]**,**
5. **query** [17] **the security attributes Key type, Key usage type, Key access control attributes, Key validity time period and Key usage counter of an identified key** [18] **to Crypto-Officer, Key Owner** [19]**.**

Application Note 1 The refinements repeats parts of the SFR component in order to avoid iteration of the component.

Consideration of Application Note 1 The ST authors have nothing to consider here.

FMT_MSA.3/KM Static attribute initialisation – Key management

Hierarchical to No other components.

Dependencies	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.3.1/KM	The TSF shall enforce the *Key Management SFP, Cryptographic Operation SFP and Update SFP* [20] to provide *restrictive* [21] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/KM	The TSF shall allow ~~the~~ **no role** [22] to specify alternative initial values to override the default values when a **cryptographic key** ~~object or information~~ is created.

FMT_MTD.1/KM Management of TSF data – Key management

Hierarchical to	No other components.
Dependencies	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions

FMT_MTD.1.1/KM The TSF shall restrict the ability to

1. *create according to FCS_CKM.1* [23] *the cryptographic keys* [24] to Crypto-Officer [25],
2. **import according to *FPT_TCT.1/CK, FPT_TIT.1/CK and FPT_ISA.1/CK*** [26] **the *cryptographic keys*** [27] **to *Crypto-Officer*** [28],
3. **export according to *FPT_TCT.1/CK, FPT_TIT.1/CK and FPT_ESA.1/CK*** [29] **the *cryptographic keys*** [30] **to *Crypto-Officer*** [31] ***if security attribute of the key allows export (keys with security attribute Key Usage Counter must never be exported),***
4. **delete according to *FCS_CKM.4*** [32] **the *cryptographic keys*** [33] **to Crypto-Officer and Key Owner** [34].

Application Note 2	The bullets 2. to 4. are refinements to avoid an iteration of component and therefore printed in bold.
Consideration of Application Note 2	The ST authors have nothing to consider here.

Hash based functions

FCS_COP.1/Hash Cryptographic operation – Hash

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 *hash generation* [35] in accordance with a specified cryptographic algorithm *SHA-256, SHA-384, SHA-512* [36] and cryptographic key sizes *none* [37] that meet the following: **[FIPS_180-4]** [38] .
Application Note 3	The hash function is a cryptographic primitive used for HMAC, cf. **FCS_COP.1/HMAC**, digital signature creation, cf. FCS_COP.1/CDS-**, digital signature verification, cf. FCS_COP.1/VDS-**, and key derivation, cf. CS_CKM.5.
Consideration of Application Note 3	The TOE uses the hash function of this SFR in the implementation of the above mentioned functionality.

Management of Certificates

FMT_MTD.1/RK Management of TSF data – Root key

Hierarchical to	No other components.
Dependencies	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MTD.1.1/RK	The TSF shall restrict the ability to 1. *create* [39] *, modify, clear and delete* [40] the *root key pair* [41] to **none** [42]. 2. ***import and delete*** [43] ***a known as authentic public key of a certification authority in a PKI*** [44] **to Crypto-Officer** [45]**.**
Application Note 4	The root key is defined here with respect to the key hierarchy known to the TOE. In case of clause (1), i. e. may be a key pair of an TOE internal key hierarchy. In clause (2) it may be a root public key of a PKI or a public key of another certification authority in a PKI known as being an authentic certificate signing key. The PKI may be used for user authentication, key management and signature-verification. The second bullet is a refinement to avoid an iteration of component and therefore printed in bold.
Consideration of Application Note 4	The ST authors have nothing to consider here.

FPT_TIT.1/Cert TSF data integrity transfer protection – Certificates

Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]
FPT_TIT.1.1/Cert	The TSF shall enforce the *Key Management SFP* [46] to *receive* [47] a **certificate** ~~TSF data~~ in a manner protected from *modification and insertion* [48] errors.
FPT_TIT.1.2/Cert	The TSF shall be able to determine on receipt of a **certificate** ~~TSF data~~ , whether *modification and insertion* [48] has occurred.

FPT_ISA.1/Cert Import of TSF data with security attributes - Certificates

Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency
FPT_ISA.1.1/Cert	The TSF shall enforce the *Key management SFP* [46] when importing **certificates** ~~TSF data~~, controlled under the SFP, from outside of the TOE.
FPT_ISA.1.2/Cert	The TSF shall use the security attributes associated with the imported **certificate**~~TSF data~~.
FPT_ISA.1.3/Cert	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the **certificates** ~~TSF data~~ received.
FPT_ISA.1.4/Cert	The TSF shall ensure that **the** interpretation of the security attributes of the imported **certificate**~~TSF data~~ is as intended by the source of the **certificates**~~TSF data~~.
FPT_ISA.1.5/Cert	The TSF shall enforce the following rules when importing ~~TSF data~~ controlled under the SFP from outside the TOE:

1. *The TSF imports the TSF data in certificates only after successful verification of the validity of the certificate in the certificate chain until it is known as an authentic certificate according to FMT_MTD.1/RK.*
2. *The validity verification of the certificate shall include*
    a. *except for root certificates, the verification of the digital signature of the certificate issuer and*
    b. *a verification that the security attributes in the certificate pass the interpretation according to FPT_TDC.1* [49].

FPT_TDC.1/Cert Inter-TSF basic TSF data consistency - Certificate

Hierarchical to	No other components.
Dependencies	No dependencies.
FPT_TDC.1.1/Cert	The TSF shall provide the capability to consistently interpret *security attributes of cryptographic keys in the certificate and the identity of the certificate issuer* [50] when shared between the TSF and another trusted IT product.
FPT_TDC.1.2/Cert	The TSF shall use **the following rules:** 1. *the TOE reports about conflicts between the Key identities of stored cryptographic keys and cryptographic keys to be imported,* 2. *the TOE does not change the security attributes Key identity, Key owner, Key type, Key usage type and Key validity time period of a public key that is imported from the certificate,* 3. *the identity of the certificate issuer shall meet the identity of the signer of the certificate* [51] when interpreting **the certificate from a trust centre** ~~TSF data from another trusted IT product~~.
Application Note 5	The security attributes assigned to a certificate holder and the cryptographic key in the certificate are used as TSF data of the TOE. The certificate is imported from a trust centre directory service, but must be verified by the TSF (i.e. if it is verified successfully that the source is the trust centre's directory server of the trusted IT product).
Consideration of Application Note 5	The TOE always imports certificates with according security attributes and does not alter them. In addition, the certificates are accordingly checked as required here.

Key generation, agreement and destruction

Key generation (cf. FCS_CKM.1/ECC, FCS_CKM.1/RSA) is a randomized process which uses random secrets (cf. FCS_RNG.1), applies key generation algorithms and defines security attributes depending on the intended use of the keys. It has the property that it is computationally infeasible to deduce the output without prior knowledge of the secret input. *Key derivation (cf. FCS_CKM.5/ECC)* is a deterministic process by which one or more keys are calculated from a pre-shared key or shared secret or other information. It allows repeating the key generation if the same input is provided. *Key agreement* (cf. FCS_CKM.5/ECDHE) is a key-establishment procedure process for establishing a shared secret key between entities in such a way that neither of them can predetermine the value of that key independently of the other party's contribution. Key agreement allows each participant to enforce the cryptographic quality of the agreed key. The component FCS_CKM.1 was refined for key agreement because it normally uses random bits as input. Hybrid cryptosystems (FCS_CKM.1/ECKA-EG, FCS_CKM.1/AES_RSA) are a combination of a public key cryptosystem with an efficient symmetric key cryptosystem.

The user may need to specify the type of key, the cryptographic key generation algorithm, the security attributes and other necessary parameters.

FCS_RNG.1 Random number generation

Hierarchical to	No other components.
Dependencies	No dependencies.
FCS_RNG.1.1	The TSF shall provide a *deterministic* [52] random number generator that implements: *DRG.3.1 If initialized with a random seed provided by the hardware platform, the internal state of the RNG shall have at least 125 55 bits of entropy. DRG.3.2 The RNG provides forward secrecy. DRG.3.3 The RNG provides backward secrecy even if the current internal state is known* [53].
FCS_RNG.1.2	The TSF shall provide random numbers that meet *(DRG.3.4) The RNG gets initialized with a random seed during every start-up of the TOE and generates output for which 2^14 strings of bit length 128 are mutually different with probability 1 – 2^(-8) . (DRG.3.5) Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A* [54].
Application Note 6	The random bit generation shall be used for key generation and key agreement according to all instantiations of FCS_CKM.1, challenges in cryptographic protocols and cryptographic operations using random values according to FCS_COP.1/HEM and FCS_COP.1/TCE. The TOE also provides the random number generation as security service for the user.

Consideration of Application Note 6 The TOE provides random number generation as a service and uses the random number generator of **FCS_RNG.1** accordingly for the generation of cryptographic keys and key agreements. Note, that the TOE in addition provides retrieval of physical random numbers, being generated by the platform of the TOE as a service. Still, this physical random number generator is not part of TOE.

FCS_CKM.1/AES Cryptographic key generation – AES key

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 cryptographic AES keys in accordance with a specified cryptographic key generation algorithm AES [55] and specified cryptographic key sizes 128 bits, *256 bits* [56] that meet the following: **[ISO-18033-3]** [57].
Application Note 7	The cryptographic key(s) may be also used together with **FCS_COP.1/ED**, e. g. for internal purposes.
Consideration of Application Note 7	The TOE makes use of this possibility accordingly.

FCS_CKM.5/AES Cryptographic key derivation – AES key derivation

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.5.1/AES	The TSF shall derive cryptographic AES keys [58] from *input parameters of sufficient entropy* [59] in accordance with a specified cryptographic key derivation algorithms *AES key generation using a bit string derived from input parameters with a KDF* [60] and specified cryptographic key sizes *128 bits,256 bits* [61] that meet the following: *NIST SP800-56C* **[NIST-SP800-56C]** [57].

FCS_CKM.1/ECC Cryptographic key generation – Elliptic curve key pair ECC

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/ECC	The TSF shall generate cryptographic **elliptic curve** key **pairs** in accordance with a specified cryptographic key generation algorithm *ECC key pair generation with Curve P-256* [62] and specified cryptographic key sizes *256 bits* [63] that meet the following: ***[FIPS_186-4]*** *B.4 and D.1.2.3* [64].

Application Note 8 The elliptic key pair generation uses a random bit string as input for the ECC key generation algorithm. The keys generation according to **FCS_CKM.1/ECC** and key derivation according to **FCS_CKM.5/ECC** are intended for different key management use cases but the keys themselves may be used for same cryptographic operations.

Consideration of Application Note 8 The usage of the generated key is implemented accordingly.

ST Application Note 2 The selections in **FCS_CKM.1/ECC** refer to **Table 4** in this ST document.

FCS_CKM.5/ECC Cryptographic key derivation – ECC key pair derivation

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.5.1/ECC	The TSF shall derive cryptographic *elliptic curve keys pairs* [58] from *input parameters of sufficient entropy* [65] in accordance with a specified cryptographic key derivation algorithm *ECC key pair generation with Curve P-256* [66] *using bit string derived from input parameters with X9.63 Key Derivation Function* [67] *and specified cryptographic key sizes 256 bits* [68] *that meet the following:* ***[FIPS_186-4]*** *B.4 and D.1.2.3*, ***[BSI-TR-03111]*** [69].

Application Note 9 The elliptic key pair derivation applies a key derivation function (KDF), e.g. from **[BSI-TR-03111]** (Section 4.3.3.) to the input parameter. It uses the output string of a KDF instead of the random bit string as input for the ECC key generation algorithm (**[BSI-TR-03111]**, Section 4.1.1, Algorithms 1 or 2). The input parameters shall include a secret of the length of at least of the key size to ensure the confidentiality of the private key. The input parameters may include public known values or even values provided by external entities.

Consideration of Application Note 9 The TOE implements the generation of the keys accordingly and uses sufficient input as required.

ST Application Note 3 The selections in **FCS_CKM.5/ECC** refer to **Table 4** in this ST document.

FCS_CKM.1/RSA Cryptographic key generation – RSA key pair

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 cryptographic **RSA** key **pairs** in accordance with a specified cryptographic key generation algorithm RSA [55] and specified cryptographic key sizes *4096 bits* [70] that meet the following: *PKCS #1 v2.2* **[PKCS-1]** [57].
Application Note 10	The cryptographic key sizes assigned in **FCS_CKM.1/RSA** must be at least 2000 bits. Cryptographic key sizes of at least 3000 bits are recommended. The SFR **FCS_CKM.1/RSA** assigns given security attributes Key identity and Key owner.
Consideration of Application Note 10	For all generated keys the TOE immediateley assigns the required security attributes. In addition, the TOE implements checks to ensure, that RSA keys with insufficient key sizes can not be generated.

FCS_CKM.5/ECDHE Cryptographic key derivation – Elliptic Curve Diffie-Hellman ephemeral key agreement

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.5.1/ECDHE	The TSF shall derive cryptographic ephemeral keys [58] **for data encryption and MAC with AES-128,** AES-256 [71] from *an agreed shared secret* [72] in accordance with a specified cryptographic key derivation algorithm *Elliptic Curve Diffie Hellman ephemeral key agreement* Curve P-256 [73] and 256-bit random ECP group [74] with a key derivation from the shared secret *SHA-1 for AES-128, SHA256 for AES-256* [67] and specified cryptographic key sizes 128 bits or 256 bits [75] [70] that meet the following: *TR-03111* **[BSI-TR-03111]** [57].

Application Note 11 The input parameters for key derivation is an agreed shared secret established by means of Elliptic Curve Diffie-Hellman. Table 4 lists elliptic curves and Table 5 lists Diffie-Hellman Groups for the agreement of the shared secret. SHA-1 shall be supported for generation of 128 bits AES keys. SHA-256 shall be selected and used to generate 256 bits AES keys.

Consideration of Application Note 11 The TOE implements this function accordingly.

ST Application Note 4 The selections in **FCS_CKM.5/ECDHE** refer to **Table 4** and **Table 5** in this ST document.

FCS_CKM.1/ECKA-EG Cryptographic key generation – ECKA-EG key generation with ECC encryption

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/ECKA-EG The TSF shall generate **ephemeral** cryptographic **elliptic curve** key **pairs for ECKGA-EG [BSI-TR-03111]**, sender role in accordance with a specified cryptographic key generation algorithm ECC key pair generation with *Curve P-256* [76] and specified cryptographic key sizes *256 bits* [77] that meet the following: ***[FIPS_186-4] B.4 and D.1.2.3*** [78].

ST Application Note 5 The selections in **FCS_CKM.1/ECKA-EG** refer to **Table 4** in this ST document.

FCS_CKM.5/ECKA-EG Cryptographic key derivation – ECKA-EG key derivation

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.5.1/ECKA-EG The TSF shall derive cryptographic *data encryption and MAC* keys *for AES-128, AES-256* [79] from *a private and a public ECC key* [80] in accordance with a specified cryptographic key derivation algorithm *ECKGA-EG [TR-03111] Curve P-256* [81] *and X9.63 Key Derivation Function* [67] and specified cryptographic **symmetric** key sizes *128 bits 256 bits* [70] that meet the following: *TR03111* **[BSI-TR-03111]**, chapter 4.3.2.2 [57].

Application Note 12 FCS_CKM.5/ECKA-EG is used by both the sender (encryption) and the recipient (decryption) to compute a secret point S_{AB} on an elliptic curve and derived a shared secret Z_{AB}. The shared secret is then used as the input to the key derivation function to derive two symmetric keys: the encryption key and the MAC key. These are then used to encrypt or decrypt messages according to FCS_COP.1/HEM or FCS_COP.1/HDM, respectively. Sender and recipient use however different inputs to FCS_CKM.5/ECKA-EG. The sender first generates an ephemeral ECC key pair according to FCS_CKM.1/ECKA-EG and uses the generated ephemeral private key and the static public key of the recipient as input. The recipient first extracts the ephemeral public key from the message and uses the ephemeral public key and the static private key (cf. FCS_CKM.1/ECC for key generation) as the input to derive the symmetric keys. The selection of the elliptic curve, the ECC key size and length of the shared secret shall correspond to the selection of the AES key size, e. g. brainpoolP256r1 and 256 bits seed for ECC key and AES keys. FCS_CKM.1/ECKA-EG and FCS_CKM.5/ECKA-EG do not provide self-contained security services for the user but are necessary steps for FCS_COP.1/HEM and FCS_COP.1/HDM (refer to the next section **Hybrid encryption with MAC for user data**).

Consideration of Application Note 12 The explanation of the application note is considered and implemented in the TOE as required.

ST Application Note 6 The selections in FCS_CKM.5/ECKA-EG refer to Table 4 in this ST document.

FCS_CKM.1/AES_RSA Cryptographic key generation – Key generation and RSA encryption

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_RSA The TSF shall generate **and encrypt** a **seed, derive** cryptographic keys **from the seed for data encryption and MAC with AES-128,** AES-256 [82] in accordance with a specified cryptographic key generation algorithm *X9.63 Key Derivation Function [ANSI-X9.63] and RSA EME-OAEP [PKCS-1]* [55] and specified cryptographic **symmetric** key sizes *128 bits 256 bits* [83] that meet the following: *ISO/IEC18033-3 [ISO-18033-3], PKCS #1 v2.2 [PKCS-1]* [57].

Application Note 13 The asymmetric cryptographic key sizes used in **FCS_CKM.1/AES_RSA** must be at least 2000 bits. Cryptographic key sizes of at least 3000 bits are recommended. **FCS_CKM.1/AES_RSA** and **FCS_CKM.5/AES_RSA** do not provide self-contained security services for the user but they are only necessary steps for **FCS_COP.1/HEM** respective **FCS_COP.1/HDM** (refer to the next section **Hybrid encryption with MAC for user data**).

Consideration of Application Note 13 The TOE ensures, that RSA keys with insufficient key sizes are not be generated.

FCS_CKM.5/AES_RSA Cryptographic key derivation – RSA key derivation and decryption

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.5.1/AES_RSA	The TSF shall derive cryptographic *data encryption keys and MAC keys for AES-128, AES-256* [84] from **a decrypted** *RSA encrypted seed* [85] in accordance with a specified cryptographic key derivation algorithm *RSA EME-OAEP [PKCS-1] and X9.63 [ANSI-X9.63] Key Derivation Function* [67] and specified cryptographic **symmetric** key sizes *128 bits 256 bits* [86] that meet the following: *ISO/IEC 14888-2 [ISO-IEC_14888-2]* [57].

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 *deletion by overwriting with zeros* [87] that meets the following: *[FIPS140-2] zeroization standards, chapter 4.7.6* [57].

Refinement: The destruction of cryptographic keys shall ensure that any previous information content of the resource about the key is made unavailable upon the deallocation of the resource.

Key import and export

FCS_COP.1/KW Cryptographic operation – Key wrap

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, FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1/KW	The TSF shall perform *key wrap* [88] in accordance with a specified cryptographic algorithm *AES-Keywrap KWP* [89] and cryptographic key sizes **of the key encryption key** *128 bits 256 bits* [90] that meet the following: ***[NIST-SP800-38F]*** [57].
Application Note 14	The selection of the length of the key encryption key shall be equal or greater than the security bits of the wrapped key for its cryptographic algorithm.
Consideration of Application Note 14	The selection of the length of the key encryption key is equal or greater than the security bits of the wrapped key for its cryptographic algorithm.

FCS_COP.1/KU Cryptographic operation – Key unwrap

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, FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1/KU	The TSF shall perform *key unwrap* [88] in accordance with a specified cryptographic algorithm *AES-Keywrap KWP* [91] [92] and cryptographic key sizes **of the key encryption key** *128 bits 256 bits* [70] that meet the following: *NIST SP800-38F* ***[NIST-SP800-38F]*** [57].

FPT_TCT.1/CK TSF data confidentiality transfer protection – Cryptographic keys

Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]

FPT_TCT.1.1/CK	The TSF shall enforce the *Key Management SFP* [93] by providing the ability to *transmit and receive* [94] **a cryptographic key** ~~TSF data~~ in a manner protected from unauthorised disclosure **according to FCS_COP.1/KW and FCS_COP.1/KU**.

FPT_TIT.1/CK TSF data integrity transfer protection – Cryptographic keys

Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]
FPT_TIT.1.1/CK	The TSF shall enforce the *Key Management SFP* [93] to *transmit and receive* [94] **a cryptographic key** ~~TSF data~~ in a manner protected from *modification and insertion* [95] errors **according to FCS_COP.1/KW**.
FPT_TIT.1.2/CK	The TSF shall be able to determine on receipt of **cryptographic keys** ~~TSF data~~, whether *modification and insertion* [95] has occurred **according to FCS_COP.1/KU**.

FPT_ISA.1/CK Import of TSF data with security attributes – Cryptographic keys

Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency
FPT_ISA.1.1/CK	The TSF shall enforce the *Key Management SFP* [93] when importing **cryptographic keys** ~~TSF data~~, controlled under the SFP, from outside of the TOE.
FPT_ISA.1.2/CK	The TSF shall use the security attributes associated with the imported **cryptographic keys** ~~TSF data~~.
FPT_ISA.1.3/CK	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the **cryptographic keys** ~~TSF data~~ received.
FPT_ISA.1.4/CK	The TSF shall ensure that interpretation of the security attributes of the imported **cryptographic keys** ~~TSF data~~ is as intended by the source of the **cryptographic keys** ~~TSF data~~.

FPT_ISA.1.5/CK The TSF shall enforce the following rules when importing a **cryptographic keys** ~~TSF data~~ controlled under the SFP from outside the TOE:

1. *The TSF imports the TSF data in certificates only after successful verification of the validity of the certificate including the verification of the digital signature of the issuer and the validity time period.*
2. *no additional importation control rules* [96].

Application Note 15 The operational environment is obligated to use trust centre services for secure key management, cf. **OE.SecManag**.

Consideration of Application Note 15 The TOE guidance will enforce the operational environment to implement **OE.SecManag** as required.

FPT_TDC.1/CK Inter-TSF basic TSF data consistency – Key import

Hierarchical to No other components.

Dependencies No dependencies.

FPT_TDC.1.1/CK The TSF shall provide the capability to consistently interpret *security attributes of the imported cryptographic keys* [97] when shared between the TSF and another trusted IT product.

FPT_TDC.1.2/CK The TSF shall use **the following rules:**

1. *the TOE reports about conflicts between the Key identity of stored cryptographic keys and cryptographic keys to be imported,*
2. *the TOE does not change the security attributes Key identity, Key type, Key usage type and Key validity time period of the key being imported* [98]

when interpreting **the imported key data object** ~~TSF data from another trusted IT product~~.

FPT_ESA.1/CK Export of TSF data with security attributes – Cryptographic keys

Hierarchical to No other components.

Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency
FPT_ESA.1.1/CK	The TSF shall enforce the *Key Management SFP* [93] when exporting **cryptographic keys** ~~TSF data~~, controlled under the SFP, from outside of the TOE.
FPT_ESA.1.2/CK	The TSF shall export the **cryptographic keys** ~~TSF data~~ with the **cryptographic key's** ~~TSF data~~ associated security attributes.
FPT_ESA.1.3/CK	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported **cryptographic keys** ~~TSF data~~.
FPT_ESA.1.4/CK	The TSF shall enforce the following rules when a **cryptographic keys** ~~TSF data~~ is exported from the TOE: *For keys with the security attribute "Key Usage Counter", the TSF must ensure that decreasing the counter importing an older version of the key is impossible. Additionally there are no other exportation control rules* [99] .

Application Note 16 There are no fixed rules for presentation of security attributes defined. The element **FPT_ESA.1.4.CK** must define rules expected in FPT_TDC.1 Inter-TSF basic TSF data consistency if inter-TSF key exchange is intended.

W.r.t. to **FPT_ESA.1.4.CK** note the following naive attack: 1) A user exports a key having the attribute "Key Usage Counter". 2) The key is then re-imported and used several times. 3) The key is exported again and 4) the exported version of 1) instead of the one of 3.) is re-imported, thus effectively decreasing the attribute "Key Usage Counter". A straight-forward way to counter this is to prohibit keys with the attribute "Key Usage Counter" from being exported.

Consideration of Application Note 16 The SFR **FMT_MTD.1.1.KM** clause (3) forbids exports of keys with "Key Usage Counter" and refers explicitly to **FPT_ESA.1/CK**. If a user should try to export a key that has an associated signature counter, the export will be rejected. Please note that the TOE will also reject an import of a key with an associated signature counter.

6.1.2. Data encryption

FCS_COP.1/ED Cryptographic operation – Data encryption and decryption

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/ED	The TSF shall perform *data encryption and decryption* [88] in accordance with a specified cryptographic algorithm *symmetric data encryption according to AES-128 and AES-256* [100] *in CBC and no other* [101] *mode* and cryptographic key *size 128 bits, 256 bits* [102] that meet the following: *NIST-SP800-38A [NIST-SP800-38A], ISO 18033-3 [ISO-18033-3], ISO 10116 [ISO-IEC_10116]* [57].

Application Note 17 Data encryption and decryption should be combined with data integrity mechanisms in Encrypt-then-MAC order, i. e. the MAC is calculated over the ciphertext and verified before decryption. The modes of operation should combine encryption with data integrity mechanisms into authenticated encryption, e. g. Cipher Block Chaining Mode (CBC, cf. **[NIST-SP800-38A]**) should be combined with CMAC (cf. **FCS_COP.1/MAC**) or HMAC (cf. **FCS_COP.1/HMAC**). For combination of symmetric encryption, decryption and data integrity mechanisms by means of CCM or GCM refer to the next section **Hybrid encryption with MAC for user data**.

Consideration of Application Note 17 The TOE implements the data protection as detailed in Application Note 17.

6.1.3. Hybrid encryption with MAC for user data

FCS_COP.1/HEM Cryptographic operation – Hybrid data encryption and MAC calculation

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/HEM	The TSF shall perform *hybrid data encryption and MAC calculation* [88] in accordance with a specified cryptographic algorithm *asymmetric key encryption according to FCS_CKM.5/ECDHE* [103], *symmetric data encryption according to AES-128, AES-256* [104] *[FIPS_197] in CBC [NIST-SP800-38A]* [105] *mode with CMAC [NIST-SP800-38B] or HMAC [RFC-2104]* [106] *calculation* and cryptographic **symmetric** key sizes *128 bits, 256 bits* [107] that meet the following: *the referenced standards above according to the chosen selection* [57].

Application Note 18	Hybrid data encryption and MAC calculation is a self-contained security service of the TOE. The generation and encryption of the seed, derivation of encryption and MAC keys as well as AES encryption and MAC calculation are only steps of this service . Hybrid encryption is combined with MACs as data integrity mechanisms for the cipher text, i. e. encrypt-then-MAC creation for CMAC.
Consideration of Application Note 18	The ST authors have nothing to consider here.

FCS_COP.1/HDM Cryptographic operation – Hybrid data decryption and MAC verification

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/HDM	The TSF shall perform *hybrid MAC verification and data decryption* [88] in accordance with a specified cryptographic algorithm *asymmetric key decryption according to* ***FCS_CKM.5/ECDHE*** [108], *verification of CMAC* ***[NIST-SP800-38B]*** [109] *and symmetric data decryption according to AES with AES-128, AES-256* [110] ***[FIPS_197]*** *in mode CBC* ***[NIST-SP800-38A]*** [111] [92] and cryptographic **symmetric** key sizes *128 bits, 256 bits* [112] that meet the following: *the referenced standards above according to the chosen selection* [57].
Application Note 19	Hybrid data decryption and MAC verification is a self-contained security service of the TOE. The decryption of the seed and derivation of the encryption key and MAC key as well as the AES decryption and MAC verification are only steps of this service . The used symmetric key shall fit to the AES CMAC or GMAC and the AES algorithm for decryption of the cipher text for MAC, e. g. verification-then-decrypt for CMAC.
Consideration of Application Note 19	The TOE implements hybrid data decryption and MAC verification as described in Application Note 19.

6.1.4. Data integrity mechanisms

Cryptographic data integrity mechanisms comprise two types of mechanisms – symmetric message authentication code mechanisms and asymmetric digital signature mechanisms. A message authentication code mechanism comprises the generation of a MAC for the original message, the verification of a given pair of a message and MAC, and management of the underlying symmetric key(s).

The MAC may be applied to a plaintext without encryption, but when combined with encryption it should be applied to ciphertexts in Encrypt-then-MAC order.

FCS_COP.1/MAC Cryptographic operation – MAC using AES

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/MAC	The TSF shall perform *MAC generation and verification* [88] in accordance with a specified cryptographic algorithm *AES-128 and AES-256* [113] ***[FIPS_197]*** *CMAC* ***[NIST-SP800-38B]*** *and no other* [114] and cryptographic key sizes *128 bits, 256 bits* [115] that meet the following: *the referenced standards above according to the chosen selection* [57].
Application Note 20	The MAC may be applied to plaintexts and cipher texts. The algorithm AES-128 CMAC is mandatory.
Consideration of Application Note 20	The TOE does not apply a MAC to plaintexts. The only use case from the Protection Profile would be a PACE channel that is only protected against modification. However, the TOE always enforces encryption for the PACE channel.

FCS_COP.1/HMAC Cryptographic operation – HMAC

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/HMAC	The TSF shall perform *HMAC generation and verification* [88] in accordance with a specified cryptographic algorithm *HMAC-SHA256 and no other* [116] and cryptographic key sizes *128 bits and above* [70] that meet the following: *RFC2104* ***[RFC-2104]*** *, ISO 9797-2* ***[ISO-IEC_9797-2]*** [57].
Application Note 21	The cryptographic key is a random bit string generated by **FCS_RNG.1** or a referenced internal secret. The cryptographic key sizes assigned in **FCS_COP.1/HMAC** must be at least 128 bits.

Consideration of Application Note 21 The TOE chooses appropriate key lengths implementing the requirements of Application Note 21.

FCS_COP.1/CDS-ECDSA Cryptographic operation – Creation of digital signatures ECDSA

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/CDS-ECDSA	The TSF shall perform *signature-creation* [88] in accordance with a specified cryptographic algorithm *ECDSA with Curve P-256* [117] and cryptographic key sizes *256 bits* [118] that meet the following: ***[FIPS_186-4] B.4 and D.1.2.3*** [119].

ST Application Note 7 The selections in **FCS_COP.1/CDS-ECDSA** refer **Table 4** in this ST document.

FCS_COP.1/VDS-ECDSA Cryptographic operation – Verification of digital signatures ECDSA

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/VDS-ECDSA	The TSF shall perform *signature-verification* [88] in accordance with a specified cryptographic algorithm *ECDSA with Curve P-256* [120] and cryptographic key sizes *256 bits* [121] that meet the following: ***[FIPS_186-4] B.4 and D.1.2.3*** [122].

ST Application Note 8 The selections in **FCS_COP.1/VDS-ECDSA** refer **Table 4** in this ST document.

FCS_COP.1/CDS-RSA Cryptographic operation – Creation of digital signatures RSA

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/CDS-RSA	The TSF shall perform *signature-creation* [88] in accordance with a specified cryptographic algorithm *RSA and EMSA-PSS* [92] and cryptographic key sizes *4096 bits* [123] that meet the following: *[ISO-IEC_14888-2]*, *[PKCS-1]* [57].

FCS_COP.1/VDS-RSA Cryptographic operation – Verification of digital signatures RSA

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/VDS-RSA	The TSF shall perform *signature-verification* [88] in accordance with a specified cryptographic algorithm *RSA and EMSA-PSS* [92] and cryptographic key sizes *4096 bits* [124] that meet the following: *[ISO-IEC_14888-2]*, *[PKCS-1]* [57].

FDP_DAU.2/Sig Data Authentication with Identity of Guarantor - Signature

Hierarchical to	FDP_DAU.1 Basic Data Authentication
Dependencies	FIA_UID.1 Timing of identification
FDP_DAU.2.1/Sig	The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of user data [125] **imported according to FDP_ITC.2/UD by means of FCS_COP.1.CDS-ECDSA [126] and keys holding the security attribute Key identity assigned to the guarantor and Key usage type "digitalSignature".**
FDP_DAU.2.2/Sig	The TSF shall provide *external entities* [127] with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

Application Note 22 The TSF according to **FDP_DAU.2/Sig** is intended for a signature service for user data. The user data source shall select the security attributes Key owner of the guarantor and *Key usage type "digitalSignature"* of the cryptographic key for the signature service in the security attributes provided with the user data. The user data source subject shall meet the *Key access control attributes* for the signature-creation operation. The verification of the evidence requires a certificate showing the identity of the key owner.

Consideration of Application Note 22 The ST authors have nothing to consider here.

6.1.5. Authentication and attestation of the TOE, trusted channel

FIA_API.1/PACE Authentication Proof of Identity – PACE authentication to Application component

Hierarchical to	No other components.
Dependencies	No dependencies.
FIA_API.1.1/PACE	The TSF shall provide *PACE in ICC role* [128] to prove the identity of the *TOE* [129] to an external entity **and to establish a trusted channel according to FTP_ITC.1 case 1 or 2**.
ST Application Note 9	The TOE enforces the use of encryption within the PACE channel. As such, the TOE only implements case 2 according to FTP_ITC.1.

FIA_API.1/CA Authentication Proof of Identity – Chip authentication to user

Hierarchical to	No other components.
Dependencies	No dependencies.
FIA_API.1.1/CA	The TSF shall provide *Chip Authentication Version 2 according to [BSI-TR-03110] section 3.4* [128] to prove the identity of the *TOE* [129] to an external entity **and to establish a trusted channel according to FTP_ITC.1 case 3**.

FDP_DAU.2/Att Data Authentication with Identity of Guarantor - Attestation

Hierarchical to	FDP_DAU.1 Basic Data Authentication
Dependencies	FIA_UID.1 Timing of identification
FDP_DAU.2.1/Att	The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of *attestation data* [125] **by means of** FCS_COP.1.CDS-ECDSA [130], ***no further cryptographic authentication mechanisms*** [131] **and keys holding the security attributes Key identity assigned to the TOE sample, and Key usage type "contentCommitment".**
FDP_DAU.2.2/Att	The TSF shall provide *external entities* [127] with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

Application Note 23 The attestation data shall represent the TOE sample as a genuine sample of the certified product. The attestation data may include the identifier of the certified product, the serial number of the device or a group of product samples, the hash value of the TSF implementation and some TSF data as result of a self-test, or other data. It may be generated internally or may include internally generated and externally provided data. The assigned cryptographic mechanisms shall be appropriate for attestation meeting **OSP.SecCryM**, e. g. a digital signature, a group signature or a direct anonymous attestation mechanism as e.g. used for Trusted Platform Modules **[TPMLib-P1]** or FIDO U2F Authenticators **[FIDO-ECDAA]**.

Consideration of Application Note 23 The TOE creates a digital signature of the required data to enable the attestation.

FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to No other components.

Dependencies No dependencies.

FTP_ITC.1.1 The TSF shall provide a communication channel between TSF and another trusted IT product that is ~~logically distinct from other communication channels~~ *logically separated from other communication channels* [132] and provides assured identification of its end points *Authentication of the TOE and remote entity according to the case 1 and 2 in Table 6* [133] and protection of the channel data from modification or disclosure *according to the case 2 in Table 6* [134] **as required by** *cryptographic operation according to the case 2 in Table 6* [135].

FTP_ITC.1.2 The TSF shall permit *the remote trusted IT product* [136] **determined according to FMT_MOF.1.1 clause (3)** to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for *communication with entities defined according to FMT_MOF.1 clause (4)* [137].

Table 6. Operation in SFR for trusted channel

Case	Authentication of the TOE and remote entity	Key agreement	Protection of communication data	Cryptographic operation
1	FIA_API.1/PACE, FIA_UAU.5.1 (2)	FCS_CKM.1/PACE	modification	FCS_COP.1/TCM
2	FIA_API.1/PACE, FIA_UAU.5.1 (2)	FCS_CKM.1/PACE	modification	FCS_COP.1/TCM
			disclosure	FCS_COP.1/TCE

Case	Authentication of the TOE and remote entity	Key agreement	Protection of communication data	Cryptographic operation
3	FIA_API.1/CA, FIA_UAU.5.1 (4) or (5), and (6)	FCS_CKM.1/TCAP	modification	FCS_COP.1/TCM
			disclose	FCS_COP.1/TCE

FCS_CKM.1/PACE Cryptographic key generation – Key agreement for trusted channel PACE

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/PACE The TSF shall generate cryptographic keys ***for MAC with for FCS_COP.1/TCM and if selected encryption keys for FCS_COP.1/TCE*** in accordance with a specified cryptographic key ~~generation~~ **agreement** algorithm *PACE with Curve P-256* [138] *and Generic Mapping in ICC role* [92] and specified cryptographic key sizes *128 bits, 256 bits* [139] that meet the following: ***[ICAO-Doc9303]***, *Part 11, section 4.4* [57].

Application Note 24 PACE is used to authenticate the TOE and the application component, or TOE and human user using a terminal. It establishes a trusted channel with MAC integrity protection and – if selected – also encryption.

Consideration of Application Note 24 The TOE implements PACE with encryption.

FCS_CKM.1/TCAP Cryptographic key generation – Key agreement by Terminal and Chip authentication protocols

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/TCAP The TSF shall generate cryptographic keys ***for encryption according to FCS_COP.1/TCE and MAC according to FCS_COP.1/TCM*** in accordance with a specified cryptographic key ~~generation~~ **agreement** algorithm *Terminal Authentication version 2 and Chip Authentication Version 2* [92] and specified cryptographic key sizes *128 bits, 256 bits* [140] that meet the following: ***[BSI-TR-03110]***, *section 3.3 and 3.4* [57].

FCS_COP.1/TCE Cryptographic operation - Encryption for trusted channel

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/TCE	The TSF shall perform *encryption and decryption* [88] in accordance with a specified cryptographic algorithm *AES in CBC [NIST-SP800-38A]* [141] *mode* and cryptographic key sizes *128 bits, 256 bits* [142] that meet the following: ***[FIPS_197]*** [57].

FCS_COP.1/TCM Cryptographic operation - MAC for trusted channel

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/TCM	The TSF shall perform *MAC calculation and MAC verification* [88] in accordance with a specified cryptographic algorithm *AES CMAC [NIST-SP800-38B]* [143] and cryptographic key sizes *128 bits, 256 bits* [144] that meet the following: ***[FIPS_197]*** [57].

6.1.6. User identification and authentication

FIA_ATD.1 User attribute definition – Identity based authentication

Hierarchical to	No other components.
Dependencies	No dependencies.
FIA_ATD.1.1	The TSF shall maintain the following list of security attributes belonging to individual users: 1. *Identity*, 2. *Authentication reference data*, 3. *Role*.

FMT_MTD.1/RAD Management of TSF data – Authentication reference data and Authentication Data Records

Hierarchical to No other components.

Dependencies FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MTD.1.1/RAD The TSF shall restrict the ability to

1. *create* [145] *the initial Authentication reference data of all authorized users* [146] *to User Administrator* [147] [148],
2. ***delete*** [145] ***the Authentication reference data of an authorized user*** [146] ***to User Administrator*** [149] [148],
3. ***modify*** [145] ***the Authentication reference data*** [146] ***to the corresponding authorized user*** [148].
4. ***create*** [145] ***the permanently stored session key of a trusted channel as Authentication reference data*** [146] ***to User Administrator*** [150] [148],
5. ***define*** [145] ***the time in range from 24 hours to 24 hours*** [151] ***after which the user security attribute Role of the authentication data record is reset according to FMT_SAE.1*** [146] ***to User Administrator*** [152] [148],
6. ***define*** [145] ***the value Unidentified user*** [153] ***to which the security attribute Role of the authentication data record shall be reset according to FMT_SAE.1*** [146] ***to none*** [154] [148] .

Application Note 25 The Administrator is responsible for user management. The Administrator creates and revokes a user as a known authorized user of the TSF by creating resp. deleting authentication data records and additionally authentication reference data for the user identities in these records, as defined in clause (1). The Administrator may define additional authentication reference data as described in clause (3), i. e. the trusted channel combines initial authentication of communication endpoints (cf. FIA_UAU.5.1 clause (3) and (4)) with an agreement of session keys used for authentication of exchanged messages (cf. FIA_UAU.5.1 clause (5)). The session keys may be permanently stored for trusted communication with the known authorized entity. The user manages its own authentication reference data to prevent impersonation based of known authentication data (e.g. as addressed by FMT_MTD.3). The bullets (2) to (6) are refinements in order to avoid an iteration of component and therefore printed in bold.

Consideration of Application Note 25 The ST authors have nothing to consider here.

FMT_MTD.3 Secure TSF data

Hierarchical to	No other components.
Dependencies	FMT_MTD.1 Management of TSF data
FMT_MTD.3.1	The TSF shall ensure that only secure values are accepted for *passwords* [146] **by enforcing a change of initial passwords to a different operational password on the first successful authentication of the user**.

FIA_AFL.1 Authentication failure handling

Hierarchical to	No other components.
Dependencies	FIA_UAU.1 Timing of authentication
FIA_AFL.1.1	The TSF shall detect when *5* [155] unsuccessful authentication attempts occur related to *user authentication* [156].
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been met [157], the TSF shall *block the corresponding user authentication for 2^(number failed authentication tries) seconds* [158].

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. *Identity*, 2. *Role* [160].
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: *the initial role of the user is unidentified user* [159] .
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. *after successful identification of the user, the attribute Role of the subject shall be changed from Unidentified user to Unauthenticated user*;
2. *after successful authentication of the user for a selected role, the attribute Role of the subject shall be changed from Unauthenticated User to that role*;[161]
3. *after successful re-authentication of the user for a selected role, the attribute Role of the subject shall be changed to that role* .

FMT_SAE.1 Time-limited authorisation

Hierarchical to	No other components.
Dependencies	FMT_SMR.1 Security roles FPT_STM.1 Reliable time stamps
FMT_SAE.1.1	The TSF shall restrict the capability to specify an expiration time for *a Role* [162] to *none* [163] [148] .
FMT_SAE.1.2	For each of these security attributes, the TSF shall be able to *reset the Role to the value assigned according to FMT_MTD.1/RAD, clause (6)* [164] , after the expiration time for the indicated security attribute has passed.
Application Note 26	The TSF shall implement means to handle an expiration time for the roles within a session (i.e. between power-up and power-down of the TOE) which may not necessarily meet the requirements for a reliable time stamp as required by FPT_STM.1. If the security target requires FPT_STM.1 (e.g. if the PP-module "Time Stamp and Audit" claimed), this time stamp shall be used to meet FMT_SAE.1.
Consideration of Application Note 26	This ST claims the PP-module "Time Stamp and Audit", so the time stamps resulting from **FPT_STM.1** are used to implement **FMT_SAE.1**.

FIA_UID.1 Timing of identification

Hierarchical to	No other components.
Dependencies	No dependencies.
FIA_UID.1.1	The TSF shall allow

1. *self test according to FPT_TST.1*,
2. *identification of the TOE to the user*,
3. *no further actions* [165]

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

FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of ~~that user~~ **the Unauthenticated User**.

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. *self test according to FPT_TST.1*,
2. *authentication of the TOE to the user after authentication of the user to the TOE*,
3. *identification of the user to the TOE and selection of a role* [166] *for authentication*,
4. *none* [167]

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

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Application Note 27 Clause (2) and (3) in FIA_UAU.1.1 allows mutual identification for mutual authentication, e. g. by exchange of certificates.

Consideration of Application Note 27 The ST authors have nothing to consider here.

FIA_UAU.5 Multiple authentication mechanisms

Hierarchical to No other components.

Dependencies No dependencies.

FIA_UAU.5.1 The TSF shall provide

1. *password authentication*,
2. *PACE with Generic Mapping with the TOE in ICC and the user in PCD context with the establishment of trusted channel according to FTP_ITC.1*,
3. *certificate based Terminal Authentication Version 2 according to section 3.3 in [BSI-TR-03110] with the TOE in ICC and the user in PCD context*,
4. *Terminal Authentication Version 2 with the TOE in ICC context and user in PCD context modified by omitting the verification of the certificate chain (simplified TA2)*,
5. *Chip Authentication Version 2 with establishment of a trusted channel according to FTP_ITC.1*,
6. *message authentication by MAC verification of received messages* [168]

to support user authentication.

FIA_UAU.5.2 The TSF shall authenticate any user's claimed identity according to the **rules**

1. *password authentication shall be used for authentication of human users if enabled according to FMT_MOF.1.1, clause (1)*,
2. *PACE shall be used for authentication of human users using terminals with the establishment of a trusted channel according to FTP_ITC.1*,
3. *PACE may be used for authentication of IT entities with the establishment of a trusted channel according to FTP_ITC.1*,
4. *certificate based Terminal Authentication Version 2 may be used for authentication of users whose certificate is imported as TSF data*,
5. *the simplified version of Terminal Authentication Version 2 may be used for authentication of identified users associated with a known user's public key*,
6. *message authentication by MAC verification of received messages shall be used after initial authentication of a remote entity according to clauses (2) or (3) for a trusted channel according to FTP_ITC.1*,
7. [st-assignment]#message authentication by MAC verification of received messages from a Cluster-CSPL #[169].

FIA_UAU.6 Re-authenticating

Hierarchical to No other components.

Dependencies No dependencies.

FIA_UAU.6.1 The TSF shall re-authenticate the user under the conditions

1. *changing to a role not selected for the current valid authentication session*,
2. *power on or reset*,
3. *every message received from entities after establishing trusted channel according to FIA_UAU.5.1, clause (2), (3) or (6)*,
4. *no other conditions under which re-authentication is required* [170].

6.1.7. Access control

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

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/UD	The TSF shall enforce the *Cryptographic Operation SFP* [93] when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.2.2/UD	The TSF shall use the security attributes associated with the imported user data.
FDP_ITC.2.3/UD	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/UD	The TSF shall ensure that the interpretation of the security attributes of the imported user data is as intended by the source of the user data.
FDP_ITC.2.5/UD	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE:

1. *user data imported for encryption according to FCS_COP.1/ED shall be imported with the attribute Key identity of the key and the identification of the requested cryptographic operation*,
2. *user data imported for encryption according to FCS_COP.1/HEM shall be imported with the attribute Key identity of the public key encryption key or key agreement method*,
3. *user data imported for decryption according to FCS_COP.1/HDM shall be imported with the attribute Key identity of the asymmetric decryption key, encrypted seed and data integrity check sum*,
4. *user data imported for digital signature creation shall be imported with the attribute Key identity of the private signature key*,
5. *user data imported for digital signature verification shall be imported with digital signature and Key identity of the public signature key* [171].

Application Note 28 Keys to be used for the cryptographic operation of the imported user data are identified by security attribute Key identity.

Consideration of Application Note 28 The ST authors have nothing to consider here.

FDP_ETC.2 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	The TSF shall enforce the *Cryptographic Operation SFP* [93] when exporting user data, controlled under the SFP(s), outside of the TOE.
FDP_ETC.2.2	The TSF shall export the user data with the user data's associated security attributes.
FDP_ETC.2.3	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.
FDP_ETC.2.4	The TSF shall enforce the following rules when user data is exported from the TOE:

1. *user data exported as ciphertext according to FCS_COP.1/HEM shall be exported with reference to the key decryption key, encrypted data encryption key and data integrity check sum*,
2. *user data exported as plaintext according to FCS_COP.1/HDM shall be exported only if the MAC verification confirmed the integrity of the ciphertext*,
3. *user data exported as signed data according to FCS_COP.1/CDS-ECDSA or FCS_COP.1/CDS-RSA shall be exported with a digital signature and Key identity of the used signature-creation key* [172] .

Application Note 29 In case of internally generated data exported as signed data, the Key identity of the used key should be exported as well in order to identify the corresponding signature-verification key. Note that the TOE may implement more than one signature-creation key for signing internally generated data.

Consideration of Application Note 29 The possibility to identify the key, being used to sign, is part of the exported and signed data to enable verification.

FDP_ETC.1 Export of user data without security attributes

Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_ETC.1.1	The TSF shall enforce the *Cryptographic Operation SFP* [173] when exporting user data **as plaintext according to FCS_COP.1/HDM,** controlled under the SFP(s), outside of the TOE.
FDP_ETC.1.2	The TSF shall export the ~~user data~~ **successfully MAC verified and decrypted ciphertext as plaintext according to FCS_COP.1/HDM** without the user data's associated security attributes.

FDP_ACC.1/Oper Subset access control – Cryptographic operation

Hierarchical to	No other components.
Dependencies	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1/Oper	The TSF shall enforce the *Cryptographic Operation SFP* [93] on

1. *subjects: Crypto-Officer [174], Key Owner, none [175],* ;
2. *objects: operational cryptographic keys , user data*;
3. *operations: cryptographic operation* [176].

FDP_ACF.1/Oper Security attribute based access control – Cryptographic operations

Hierarchical to	No other components.
Dependencies	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1/Oper The TSF shall enforce the *Cryptographic Operation SFP* [93] to objects based on the following:

1. *subjects: subjects with security attribute Role Crypto-Officer [177], Key Owner, no other role [178]*;
2. *objects:*
   a. *cryptographic keys with security attributes: Identity of the key, Key owner, Key type, Key usage type, Key access control attributes, Key validity time period*;
   b. *user data* [179].

FDP_ACF.1.2/Oper The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

1. *A Subject in Crypto-Officer [180] role is allowed to perform cryptographic operations on cryptographic keys in accordance with their security attributes.*
2. *The Subject Key Owner is allowed to perform cryptographic operations on user data with cryptographic keys in accordance with the security attribute Key owner, Key type, Key usage type, Key access control attributes and Key validity time period*;
3. *no other rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects* [181].

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

1. *subjects with the security attribute Role are allowed to perform cryptographic operations on user data and cryptographic keys with security attributes as shown in the rows of **Table 7**.*
2. *the TOE itself is allowed to use a private signature key of a user when signing audit or system log messages for this user.* [182].

FDP_ACF.1.4/Oper The TSF shall explicitly deny access of subjects to objects based on the following additional rules:

1. *No subject is allowed to use cryptographic keys by cryptographic operation other than those identified in the security attributes Key usage type and the Key access control attributes*;
2. *No subject is allowed to decrypt ciphertext according to FCS_COP.1/HDM if MAC verification fails.*
3. *no additional rules, based on security attributes, that explicitly deny access of subjects to objects* [183].

ST Application Note 10 FDP_ACF.1.3.Oper (1) refers to **Table 7** in this ST document.

Access control rules for cryptographic operation:

Table 7. Security attributes and access control

Security attribute Role of the Subject	**Security attribute of the cryptographic key**	**Cryptographic operation referenced by SFR allowed for the subject on user data with the cryptographic key**
Crypto-Officer [184]	*Key type: symmetric* *Key usage type: Key wrap* *Key validity time period:-*	*FCS_COP.1/KW*
Crypto-Officer [185]	*Key type: symmetric* *Key usage type: Key unwrap* *Key validity time period:-*	*FCS_COP.1/KU*
(any authenticated user)	*Key type: public* *Key usage type: ECKA-EG* *Key validity time period:as in certificate*	*FCS_COP.1/HEM,* *FCS_CKM.1/ECKA-EG*
Key Owner	*Key type: private* *Key usage type: ECKA-EG* *Key validity time period:-*	*FCS_COP.1/HDM* *FCS_CKM.5/ECKA-EG*
(any authenticated user)	*Key type: public* *Key usage type: RSA_ENC* *Key validity time period:as in certificate*	*FCS_COP.1/HEM* *FCS_CKM.1/AES_RSA*
Key Owner	*Key type: private* *Key usage type: RSA_ENC* *Key validity time period:as in certificate*	*FCS_COP.1/HDM* *FCS_CKM.5/AES_RSA*

Security attribute Role of the Subject	Security attribute of the cryptographic key	Cryptographic operation referenced by SFR allowed for the subject on user data with the cryptographic key
Key Owner	*Key type: private* *Key usage type: DS-ECDSA* *Key validity time period:-*	*FCS_COP.1/CDS-ECDSA*
(any authenticated user)	*Key type: public* *Key usage type: DS-ECDSA* *Key validity time period:-*	*FCS_COP.1/VDS-ECDSA*
Key Owner	*Key type: private* *Key usage type: DS-RSA* *Key validity time period:-*	*FCS_COP.1/CDS-RSA*
(any authenticated user)	*Key type: public* *Key usage type: DS-RSA* *Key validity time period:-*	*FCS_COP.1/VDS-RSA*
Key Owner	*Key type: secret* *Key usage type: AES encryption and decryption* *Key validity time period:-*	*FCS_COP.1/ED*

ST Application Note 11 It should be noted that the last line of the previous table has been added by the ST author.

6.1.8. Security Management

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. *management of security functions behaviour (FMT_MOF.1)*,
2. *management of Authentication reference data (FMT_MTD.1/RAD)*,
3. *management of security attributes of cryptographic keys (FMT_MSA.1/KM, FMT_MSA.2, FMT_MSA.3/KM)*,
4. *no further functions* [186].

FMT_SMR.1 Security roles

Hierarchical to	No other components.
Dependencies	FIA_UID.1 Timing of identification
FMT_SMR.1.1	The TSF shall maintain the roles: *Unidentified User, Unauthenticated User, Key Owner, Application component* *Crypto-Officer, User Administrator, Update Agent* [187] *Cluster-CSPL* [188].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.
ST Application Note 12	The additional role of *Cluster-CSPL* is used by a Master CSPL when sending information to a Slave CSPL in the context of clustering.
Application Note 30	The ST may select the general role Administrator or more detailed administrator roles as supported by the TOE. It should be noted that the role *application component* may refer to a SMAER unit using the TOE.
Consideration of Application Note 30	The ST chose more detailed administrator roles.

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 *security attributes* 1. *Key identity*, 2. *Key type*, 3. *Key usage type*, 4. *no additional security attribute* [189]. **The cryptographic keys shall have** 1. **a Key identity uniquely identifying the key among all keys implemented in the TOE,** 2. **the Key type defined as exactly one of secret key, private key, or public key,** 3. **a Key usage type identifying at least one cryptographic mechanism the key can be used for.**

FMT_MOF.1 Management of security functions behaviour

Hierarchical to	No other components.
Dependencies	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions

FMT_MOF.1.1 The TSF shall restrict the ability to

1. *enable* [190] *the functions password authentication according to FIA_UAU.5.1, clause (1)* [191] *to User Administrator* [192],
2. ***disable*** [190] ***the functions password authentication according to FIA_UAU.5.1, clause (1)*** [191] ***to User Administrator*** [193],
3. ***determine the behavior of*** [190] ***the functions trusted channel according to FTP_ITC.1.2*** [191] ***by defining the remote trusted IT products permitted to initiate communication via the trusted channel to User Administrator*** [194],
4. ***determine the behavior of*** [190] ***the functions trusted channel according to FTP_ITC.1.3*** [191] ***by defining the entities for which the TSF shall enforce communication via the trusted channel to User Administrator*** [195].

Application Note 31 The refinements of **FMT_MOF.1.1** in bullets (2) to (4) are made in order to avoid iteration of the component. In case of the client-server architecture, the applications using the TOE and supporting the cryptographically protected trusted channel belong to the entities for which the TSF shall enforce a trusted channel according to FDP_ITC.1, cf. **FMT_MOF.1.1** in bullet (4).

Consideration of Application Note 31 The TOE allows to enforces a trusted channel for applications using the TOE for signature creation.

6.1.9. Protection of the TSF

FPT_FLS.1 Failure with preservation of secure state

Hierarchical to	No other components.
Dependencies	No dependencies.

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur:

1. *self test fails* ,
2. *no additional action* [196].

Refinement: When the TOE is in a secure error mode the TSF shall not perform any cryptographic operations and all data output interfaces shall be inhibited by the TSF.

FPT_TST.1 TSF testing

Hierarchical to	No other components.
Dependencies	No dependencies.
FPT_TST.1.1	The TSF shall run a suite of self tests during initial start-up, periodically during normal operation, at the request of the authorised user [197] to demonstrate the correct operation of *the random number generator, the cryptographic functionality and the access control functionality* [198].
FPT_TST.1.2	The TSF shall provide authorised users with the capability to verify the integrity of *TSF data* [199] .
FPT_TST.1.3	The TSF shall provide authorised users with the capability to verify the integrity of *TSF **implementation*** [200].

6.1.10. Import and verification of Update Package

The TOE imports Update Code Package as user data objects with security attributes according to FDP_ITC.2/ UCP, verifies the authenticity of the received Update Code Package according to FCS_COP.1/VDSUCP, and decrypts authentic Update Code Package according to FCS_COP.1/DecUCP.

FDP_ITC.2/UCP Import of user data with security attributes – Update Code Package

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/UCP	The TSF shall enforce the *Update SFP* [201] when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.2.2/UCP	The TSF shall use the security attributes associated with the imported user data.
FDP_ITC.2.3/UCP	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/UCP 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/UCP The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE:

1. *encrypted Update Code Package are stored only after successful verification of authenticity according to FCS_COP.1/VDSUCP*,
2. *authentic Update Code Package are decrypted according to FCS_COP.1/DecUCP* [171].

FPT_TDC.1/UCP Inter-TSF basic TSF data consistency

Hierarchical to No other components.

Dependencies No dependencies.

FPT_TDC.1.1/UCP The TSF shall provide the capability to consistently interpret *security attributes Issuer and Version Number* [202] when shared between the TSF and another trusted IT product.

FPT_TDC.1.2/UCP The TSF shall use **the following rules**:

1. *the Issuer must be identified and known*,
2. *the Version Number must be identified*

when interpreting the TSF data from another trusted IT product.

FCS_COP.1/VDSUCP Cryptographic operation – Verification of digital signature of the Issuer

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/VDSUCP The TSF shall perform *verification of the digital signature of the authorized Issuer* [88] in accordance with a specified cryptographic algorithm *Curve P-256* [92] and cryptographic key sizes *256 bits* [70] that meet the following: *BSI-TR-03111* ***[BSI-TR-03111]*** [57].

Application Note 32 The authorized *Issuer* is identified in the security attribute of the received Update Code Package and the public key of the authorized *Issuer* shall be known as TSF data before receiving the Update Code Package. Only the public key of the authorized Issuer shall be used for the verification of the digital signature of the Update Code Package.

Consideration of Application Note 32 The TOE verifies UCPs accordingly. To do so it stores the corresponding public key internally.

FCS_COP.1/DecUCP Cryptographic operation – Decryption of authentic Update Code Package

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/DecUCP	The TSF shall perform *decryption of authentic encrypted Update Code Package* [88] in accordance with a specified cryptographic algorithm *AEC decryption in CBC mode* [92] and cryptographic key sizes *256 bit* [70] that meet the following: ***[FIPS_197]*** *and* ***[NIST-SP800-38A]*** [57].

FDP_ACC.1/UCP Subset access control – Update code Package

Hierarchical to	No other components.
Dependencies	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1/UCP	The TSF shall enforce the *Update SFP* [93] on 1. *subjects: Update Agent* [203]; 2. *objects: Update Code Package*; 3. *operations: import, store* .[204]

FDP_ACF.1/UCP Security attribute based access control – Import Update Code Package

Hierarchical to	No other components.
Dependencies	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1/UCP	The TSF shall enforce the *Update SFP* [93] to objects based on the following: 1. *subjects: Update Agent* [203]; 2. *objects: Update Code Package with security attributes Issuer and Version Number* [207];

FDP_ACF.1.2/UCP The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

1. *Update Agent* [203] *is allowed to import Update Code Package according to FDP_ITC.2/UCP.*;
2. *Update Agent* [203] *is allowed to store a Update Code Package if*
   a. *authenticity is successfully verified according to FCS_COP.1/VDSUCP and the Update Code Package is decrypted according to FCS_COP.1/DecUCP*
   b. *the Version Number of the Update Code Package is equal or higher than the Version Number of the TSF.* [208]

FDP_ACF.1.3/UCP The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *no further rules* [205].

FDP_ACF.1.4/UCP The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *no further rules* [206].

FDP_RIP.1/UCP Subset residual information protection

Hierarchical to No other components.

Dependencies No dependencies.

FDP_RIP.1.1/UCP The TSF shall ensure that any previous information content of a resource is made unavailable upon the *deallocation of the resource* ***after unsuccessful verification of the digital signature of the Issuer according to FCS_COP.1/VDSUCP*** [209] the following objects: *received Update Code Package* [210].

6.2. Security Functional Requirements based on PP-CSPL-TS-AU-V1.0

6.2.1. Time Stamp

FDP_DAU.2/TS Data Authentication with Identity of Guarantor - Signature with time stamp and optional key usage counter

Hierarchical to FDP_DAU.1 Basic Data Authentication

Dependencies FIA_UID.1 Timing of identification

FDP_DAU.2.1/TS The TSF shall provide a capability to generate evidence that can be used as a guarantee of the existence at certain point in time, sequence and validity of

1. *user data imported according to* ***FDP_ITC.2/UD***
2. *exported audit records according to FMT_MTD.1/Audit clause (1) and FAU_STG.3 clause (1)* [212]

with

1. **time stamp of the evidence generation according to FPT_STM.1,**
2. **and optionally the key usage counter of the signature key by means of digital signature generated according to FCS_COP.1.CDS-ECDSA [213] and keys holding the dedicated values of the security attributes Key identity that indicate key ownership of the TOE sample and Key usage type "Time stamp service".**

FDP_DAU.2.2/TS The TSF shall provide *external entities* [211] with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

Application Note 1 [PP-CSPL-TS-AU-V1.0] The TSF according to **FDP_DAU.2/TS** is intended for time stamp service of the TOE for any provided user data and exported audit records. The user data source shall select the security attribute Key usage type "TimeStamp" of the signature key of the time stamp service. The signature key of exported audit records shall be defined according to **FMT_MOF.1.1.TSA** clause (5). The Key usage counter allows to verify the sequence of signed data e. g. in an audit trail. The verification of the evidence requires a certificate showing the identity of the TOE sample and the key usage type of time stamp service. The format of input data and output data shall meet the **[BSI-TR-03151]**.

Consideration of Application Note 1 [PP-CSPL-TS-AU-V1.0] The ST authors have nothing to consider here.

6.2.2. Access Control on time stamp service

FDP_ITC.2/TS Import of user data with security attributes - User data for time stamping

Hierarchical to No 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/TS	The TSF shall enforce the Cryptographic Operation SFP [93] when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.2.2/TS	The TSF shall use the security attributes associated with the imported user data.
FDP_ITC.2.3/TS	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/TS	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/TS	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: 1. *user data imported for time stamp generation to FDP_DAU.2/TS shall be imported with security attributes Key identity of the signature key and Key usage type TimeStamp, and the identification of the requested cryptographic operation* [171] .

Application Note 2 [PP-CSPL-TS-AU-V1.0]	Keys to be used for the cryptographic operation of the imported user data are identified by security attribute Key identity.
Consideration of Application Note 2 [PP-CSPL-TS-AU-V1.0]	The ST authors have nothing to consider here.

FDP_ETC.2/TS Export of user data with security attributes - User data with time stamp

Hierarchical to	No components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_ETC.2.1/TS	The TSF shall enforce the *Cryptographic Operation SFP* [214] when exporting user data, controlled under the SFP(s), outside of the TOE.
FDP_ETC.2.2/TS	The TSF shall export the user data with the user data's associated security attributes.
FDP_ETC.2.3/TS	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.
FDP_ETC.2.4/TS	The TSF shall enforce the following rules when user data is exported from the TOE: *(1) user data exported as time stamped data according to **FDP_DAU.2/TS** shall be exported with digital signature and Key identity of the used signature-creation key* [215]

Application Note 3 [PP-CSPL-TS-AU-V1.0]	In case of internally generated data (e.g. audit records) the exported signed data shall be attributed with the Key identity of the used signature-creation key. Note that the TOE may implement more than one signature-creation key for signing internally generated data.
Consideration of Application Note 3 [PP-CSPL-TS-AU-V1.0]	The TOE attributes the exported signed data with the Key identity.

FDP_ACF.1/TS Security attribute based access control - Cryptographic operations

Hierarchical to	No components.
Dependencies	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1/TS	The TSF shall enforce the Cryptographic Operation SFP [204] to objects based on the following: 1. *subjects: subjects with security attribute Role Application Component, no other role* [217]; 2. *objects: user data* [218].
FDP_ACF.1.2/TS	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: 1. *Application Component, Auditor* [219] *is allowed to perform cryptographic operation according to **FDP_DAU.2/TS** on user data with cryptographic keys with Key usage type TimeStamp.* 2. *no further rules*. [220]
FDP_ACF.1.3/TS	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *no additional rules, based on security attributes, that explicitly authorise access of subjects to objects*. [216]
FDP_ACF.1.4/TS	The TSF shall enforce the *Cryptographic Operation SFP* to objects based on the following: 1. *No subject is allowed to use cryptographic keys by cryptographic operation other than those identified in the security attributes Key usage type and the Key access control attributes;* 2. *no further rules* [221].

6.2.3. Security Managament

FMT_SMF.1/TSA Specification of Management Functions

Hierarchical to	No components.
Dependencies	FIA_UID.1 Timing of identification

FMT_SMF.1.1/TSA The TSF shall be capable of performing the following management functions:

1. *management of security functions behaviour* ***FMT_MOF.1/TSA*** [222]

FMT_SMR.1/TSA Security roles

Hierarchical to	No components.
Dependencies	FIA_UID.1 Timing of identification

FMT_SMR.1.1/TSA The TSF shall maintain the roles **additional to those required by FMT_SMR.1 in [PP-CSPL-V1.0]**: *Auditor, Timekeeper* [223]

FMT_SMR.1.2/TSA The TSF shall be able to associate users with roles.

Application Note 4 [PP-CSPL-TS-AU-V1.0] The ST may select the general role Administrator or more detailed Administrator roles as supported by the TOE. The ST may select

- *Auditor* in **FMT_SMR.1/TSA** separated from Administrator roles selected in the SFR FMT_SMR.1 according to **[PP-CSPL-V1.0]** and, or
- *Timekeeper* role in **FMT_SMR.1/TSA** separated from Administrator roles selected in the SFR FMT_SMR.1 according to **[PP-CSPL-V1.0]** and, or
- *no other roles* in **FMT_SMR.1/TSA** and assign the management of audit TSF in FMT_MTD.1/Audit to a selected Administrator role in the SFR FMT_SMR.1 according to **[PP-CSPL-V1.0]**.

The assignment of security management of audit and other functions must not result in a conflict of duties

Consideration of Application Note 4 [PP-CSPL-TS-AU-V1.0] This ST selects more detailed Administrator roles.

FMT_MOF.1/TSA Management of security functions behaviour

Hierarchical to	No components.
Dependencies	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions

FMT_MOF.1.1/TSA The TSF shall restrict the ability to

1. *modify the behaviour of*[190] *the functions adjustment of the internal clock according to* ***FPT_STM.1*** *clause (1)*[191] *to* ***Timekeeper, Cluster-CSPL***[224],
2. ***modify the behaviour of***[190] ***the functions adjustment of the internal clock according to FPT_STM.1 clause (2)***[191] ***to Timekeeper, Cluster-CSPL***[225] ,
3. ***determine the behaviour of and modify the behaviour of***[190] ***the functions select the auditable events according to FAU_GEN.1***[191] ***to Auditor***[226],
4. ***determine the behaviour of and modify the behaviour of***[190] ***the functions automatic export of audit trails according to FAU_STG.3.1 clause (1)***[191] ***to Auditor***[226],
5. ***determine the behaviour of and modify the behaviour of***[190] ***the functions FDP_DAU.2/TS by selection of signature key used to sign exported audit trails to User Administrator***[226].

Application Note 5 [PP-CSPL-TS-AU-V1.0] The SFR defines additional management of security functions behaviour for new SFR with respect to **[PP-CSPL-V1.0]**. The refinements of **FMT_MOF.1.1.TSA** in bullets (2) to (5) are made in order to avoid further iterations of the component.

Consideration of Application Note 5 [PP-CSPL-TS-AU-V1.0] The ST authors have nothing to consider here.

ST Application Note 13 The refinements in the operations of bullet points 1 and 2 have been performed to a allow a master CSPL to trigger the time update on its slave.

6.2.4. Security Audit

FAU_GEN.1 Audit data generation

Hierarchical to	No 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:

1. Start-up and shutdown of the audit functions;
2. All auditable events for the not specified [227] level of audit; and
3. *Discrete adjustment of the real time clock*
    a. *by automatic adjustment of the clock according to **FPT_STM.1.1** clause (2) if selected as auditable event,*
    b. *by **Timekeeper as a refinement of** Administrator according to **FPT_STM.1.1** clause (1) or (2),*
    c. *failure of adjustment according to **FPT_STM.1.1***
4. *other auditable events*
    a. *Start-up after power-up*
    b. *Import of UCP (**FDP_ITC.2/UCP**),*
    c. *Authentication failure handling (**FIA_AFL.1**): the reaching of the threshold for the unsuccessful authentication attempts with claimed Identity of the user,*
    d. *(4) Generation of (selected types of) signature key pairs (all FCS_CKM.1 instantiations for generation of permanent stored keys), (6) Cryptographic key destruction (FCS_CKM.4) of permanent stored keys* [228]

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

1. Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
2. For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, *further information regarding the event if applicable* [229].

Application Note 6 [PP-CSPL-TS-AU-V1.0] The SFR **FDP_ITC.2/UCP**, **FIA_AFL.1**, FCS_CKM.1, FCS_COP.1, **FCS_CKM.4**, **FPT_FLS.1** and **FMT_MOF.1** are defined in **[PP-CSPL-V1.0]**. The SFR **FPT_STM.1**, **FMT_MOF.1/TSA** and **FMT_MTD.1/Audit** are defined in this PP-Module.

Consideration of Application Note 6 [PP-CSPL-TS-AU-V1.0] All these SFRs are part of this ST, as it claims both PPs.

FMT_MTD.1/Audit Management of TSF data

Hierarchical to No components.

Dependencies FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MTD.1.1/Audit The TSF shall restrict the ability to

1. *manual export*,
2. *clear after manual export,*
3. *select audited events in **FAU_GEN.1** and **FAU_GEN.1/CL***,
4. *define the number of audit records causing automatic export and clearing of exported audit records according to FAU_STG.3.1 clause (1),*
5. *define the percentage of storage capacity of audit records if actions are assigned in **FAU_STG.3.1** clause (2).* [145]

the audit records [146] to Auditor, Crypto-Officer [226].

Application Note 7 [PP-CSPL-TS-AU-V1.0] The selection of auditable events according to **FMT_MTD.1.1.Audit**, clause (3) enables or disables or specifies the generation of audit records as defined in **FAU_GEN.1**. The role Administrator may be selected only if it is selected in **FMT_SMR.1** in the **[PP-CSPL-V1.0]** and any conflict of duties is prevented (cf. application note to **FMT_SMR.1/TSA**).

Consideration of Application Note 7 [PP-CSPL-TS-AU-V1.0] This ST selects Auditor and not Administrator and prevents conflicts of duties this way.

ST Application Note 14 It should be noted that the owner of each key (in the role Application Component) inherently also has the role *auditor* in the context of this SFR.

FAU_STG.1 Protected audit trail storage

Hierarchical to No components.

Dependencies FAU_GEN.1 Audit data generation

FAU_STG.1.1 The TSF shall protect the stored audit records in the audit trail from unauthorised deletion.

FAU_STG.1.2 The TSF shall be able to *prevent* [230] unauthorised modifications to the stored audit records in the audit trail.

FAU_STG.3 Action in Case of Possible Audit Data Loss

Hierarchical to	No components.
Dependencies	FAU_STG.1 Protected audit trail storage

FAU_STG.3.1 The TSF shall

1. *automatically export audit trails and clear automatically exported audit records* [231] *if the audit trail exceeds an Auditor* [226] *defined number of audit records within 1* [232]
2. ***no actions*** [233] ***if the audit trail exceeds an none*** [226] ***settable percentage of storage capacity*** [234].

Application Note 8 [PP-CSPL-TS-AU-V1.0] The ST writer shall perform the open operations in **FAU_STG.3.1** element. If the number of audit records in clause (1) is set to 1 then the TSF export each audit record automatically. If the number of number of audit records in clause (1) is set higher than maximum number of audit records in the audit trail then the TSF does not export audit records automatically. The assignment of clause (2) may be "no actions" if an appropriate number of audit records is assigned in clause (1).

Consideration of Application Note 8 [PP-CSPL-TS-AU-V1.0] The number of audit records in clause (1) was set to 1 and clause (2) was resolved according to Application Note 8 **[PP-CSPL-TS-AU-V1.0]**.

FPT_STM.1 Reliable time stamps

Hierarchical to	No components.
Dependencies	No dependencies.

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps **by means of**

.

(2) internal clock with accuracy 5 seconds per day [235] **with automatic adjustment of the clock by an externally trustable source in a cryptographically verifiable manner (e.g. by signed Network Time Protocol) and the ability of adjustment of the clock by the Timekeeper** [236] [237]

Application Note 9 [PP-CSPL-TS-AU-V1.0]

The external trustable source (e.g. signed Network Time Protocol) provides a reliable time source for adjustment of the internal clock. The time intervals of adjustments in clause (2) may be configured by the Administrator. Any adjustment or failure of adjustment of the internal clock is an auditable event according to **FAU_GEN.1.1**. The refinement with selection defines different cases for internal clocks and are therefore printed in bold.

Note that it is not expected that the internal clock continues to operate when the TOE is switched off. An implementation that e.g. counts CPU ticks with sufficient accuracy while switched on would suffice to fulfil the requirements, provided that all auditable events are logged properly.

Consideration of Application Note 9 [PP-CSPL-TS-AU-V1.0]

The TOE uses signed Network Time Protocol as an external trustable source and enables an administrator to trigger the synchronisation of the internal clock with the clock of the server of the signed Network Time Protocol. This synchronisation also happens automatically at periodic intervals and on startup as part of the TOE's self test.

FPT_TIT.1/Audit TSF data integrity transfer protection - Audit functionality

Hierarchical to	No components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]
FPT_TIT.1.1/Audit	The TSF shall enforce the *Update SFP*, *Key Management SFP* [238] to *transmit* [239] TSF data **audit records** in a manner protected from *modification, deletion, insertion and replay* [240] errors.
FPT_TIT.1.2/Audit	The TSF shall be able to determine on receipt of TSF data **time**, whether *modification* [240] has occurred.

Application Note 10 [PP-CSPL-TS-AU-V1.0]

The Update SFP is enforced by the export of audit records about import of UCP, cf. **FAU_GEN.1.1** clause 4) (2). The selection of the Key Management SFP or Cryptographic Operation SFP depends on the selection of auditable events of key management, cryptographic operations and adjustment of the internal clock (e. g. used for verification of validity time period) in **FAU_GEN.1.1** clause 3). The TSF transmits audit records and receives time as TSF data for security audit. The TSF protects the audit records by means of digital signature against modification and by means of time stamps and key usage counter of the signature key as part of the signature against deletion, insertion and replay as required in **FPT_TIT.1.1**.

Consideration of Application Note 10 [PP-CSPL-TS-AU-V1.0] The selection of Key Management SFP was made, due to the selected auditable events.

6.3. Security Functional Requirements based on PP-CSPL-Cluster-V1.0

6.3.1. Security Audit

FAU_GEN.1/CL Audit data generation

Hierarchical to No components.

Dependencies FPT_STM.1 Reliable time stamps

FAU_GEN.1.1/CL The TSF shall be able to generate an audit record of the following auditable events:

1. Start-up and shutdown of the audit functions;
2. All auditable events for the not specified [227] level of audit; and
3. *other auditable events:*
    a. *Generation of cluster keys for the secure channel according to **FMT_MTD.1/CL** and **FCS_CKM.5/CLDH**,*
    b. *Export of Authentication Data Records and cryptographic keys from the MasterCSPLight according to **FPT_ESA.1.3.CL**,Management of Authentication Data Records (**FMT_MTD.1/RAD**): creation and deletion of Authentication Data Record*
    c. *Import according to **FPT_ISA.1/CL** of Authentication Data Records and cryptographic keys into Slave-CSPLights.* [241].

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

1. Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
2. For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, *further information regarding the event if applicable* [242].

Application Note 5 [PP-CSPL-Cluster-V1.0] The SFR **FAU_GEN.1/CL** adds auditable events to **FAU_GEN.1** required by **[PP-CSPL-TS-AU-V1.0]**. The SFR **FPT_STM.1** is required by **[PP-CSPL-Cluster-V1.0]**.

Consideration of Application Note 5 [PP-CSPL-Cluster-V1.0]	By claiming **[PP-CSPL-Cluster-V1.0]** the SFR **FPT_STM.1** became part of this ST.
Application Note 6 [PP-CSPL-Cluster-V1.0]	**FMT_MTD.1/RAD** is defined in the **[PP-CSPL-V1.0]**.
Consideration of Application Note 6 [PP-CSPL-Cluster-V1.0]	By claiming **[PP-CSPL-V1.0]**, **FMT_MTD.1/RAD** became part of this ST.

6.3.2. Clustering

The cluster of TOE samples is set up by the Administrator as Cluster-CSPLight(s) by

- selecting one TOE sample of the cluster as Master-CSPLight, all other TOE samples of the cluster are Slave-CSPLight(s),
- initialization of secure channels between the Master-CSPLight and the Slave-CSPLight(s),
- transfer of TSF data as security attributes of known users and cryptographic keys with security attributes between Master-CSPLight and Slave-CSPLight(s) using the application.

FDP_ACC.1/CL Subset access control - Clustering

Hierarchical to	No components.
Dependencies	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1/CL	The TSF shall enforce the Clustering SFP [93] on 1. *subjects: Crypto-Officer, Cluster-CSPL ~~Administrator~~*, 2. *objects: objects: cluster keys, Authentication Data Records, cryptographic keys, audit logs, system logs;* 3. *operations: generation, export, import* [204].
ST Application Note 15	The scope of FCP_ACC.1/CL has been extended by the ST author (as has the scope of O.Cluster). This allows to ensure that a slave TOE will be able to replace a master TOE without needing further information.

FMT_MTD.1/CL Management of TSF data - Authentication Data Records and cryptographic keys

Hierarchical to	No components.
Dependencies	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management

FMT_MTD.1.1/CL The TSF shall restrict the ability to

1. *generate according to FCS_CKM.5/CLDH [145] the cluster keys to Crypto-Officer [148].*
2. ***export from the Master-CSPLight according to FPT_ESA.1/CL, FPT_TCT.1/CL and FPT_TIT.1/CL [145] the Authentication Data Records [146] to _Cluster-CSP_L,***
3. ***import into Slave-CSPLights according to FPT_ISA.1/CL, FPT_TCT.1/CL and FPT_TIT.1/CL [145] the Authentication Data Records [146] to Cluster-CSPL***
4. ***export from the Master-CSPLight according to FPT_ESA.1/CL, FPT_TCT.1/CL and FPT_TIT.1/CL [145] the cryptographic keys [146] to Cluster-CSPL [243],***
5. ***import into Slave-CSPLights according to FPT_ISA.1/CL, FPT_TCT.1/CL and FPT_TIT.1/CL [145] the cryptographic keys [146] to Cluster-CSPL [243].***
6. **export from the Master-CSPLight audit and system log messages to Cluster-CSPL**
7. **export from the Master-CSPLight audit and system log messages to Cluster-CSPL**

ST Application Note 16 The scope of FMT_MTD.1.1.CL has been extended by the ST author (as has the scope of O.Cluster). This allows to ensure that a slave TOE will be able to replace a master TOE without needing further information.

Application Note 1 [PP-CSPL-Cluster-V1.0] Authentication Data Records and cryptographic keys are TSF data. The selection in **FMT_MTD.1/CL** allows for a more detailed separation of duties between the roles if supported by the TOE. The bullets (2) to (5) are refinements to avoid further iterations of the component **FMT_MTD.1.1.CL** and therefore printed in bold.

Consideration of Application Note 1 [PP-CSPL-Cluster-V1.0] The ST authors have nothing to consider here.

FCS_CKM.5/CLDH Cryptographic key derivation - Cluster keys

Hierarchical to	No components.
Dependencies	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction

FCS_CKM.5.1/CLDH The TSF shall derive cryptographic *cluster keys* [58] from *an agreed shared secret* [244] in accordance with a specified cryptographic key derivation algorithm *anonymous Diffie-Hellman Key Agreement for ECC key pair generation with* Curve P-256 [245] and specified cryptographic key sizes 256 bits [246] that meet the following: **[FIPS_186-4]** B.4 and D.1.2.3 [247].

Application Note 2 [PP-CSPL-Cluster-V1.0] The cryptographic cluster keys shall be used for encryption according to **FCS_COP.1/ED** and **FPT_TCT.1/CL** and MAC protection according to **FCS_COP.1/MAC** and **FPT_TIT.1/CL** during transfer of Authentication Data Records and the cryptographic keys between MasterCSPLight and Slave-CSPLight.

Consideration of Application Note 2 [PP-CSPL-Cluster-V1.0] The TOE uses the keys accordingly.

FPT_TCT.1/CL TSF data confidentiality transfer protection - Cluster

Hierarchical to	No components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset informationflow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]

FPT_TCT.1.1/CL The TSF shall enforce the *Clustering SFP* [248] by providing the ability to *transmit and receive* [249] **Authentication Data Records and cryptographic keys** ~~TSF data~~ in a manner protected from unauthorised disclosure **according to FCS_COP.1/ED**.

FPT_TIT.1/CL TSF data integrity transfer protection - Cluster

Hierarchical to	No components.

Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset informationflow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]
FPT_TIT.1.1/CL	The TSF shall enforce the *Clustering SFP* [248] to *transmit and receive* [249] **Authentication Data Records and cryptographic keys** ~~TSF data~~ in a manner protected from *modification* [240] errors **according to** **FCS_COP.1/MAC**.
FPT_TIT.1.2/CL	The TSF **in role Slave-CSPLight** shall be able to determine on receipt of **Authentication Data Records and cryptographic keys** ~~TSF data~~, whether modification [240] has occurred according to **FCS_COP.1/MAC**.

FPT_ISA.1/CL Import of TSF data with security attributes - Cluster

Hierarchical to	No components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset informationflow control] [FMT_MTD.1 Management of TSF data, or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency
FPT_ISA.1.1/CL	The TSF **in role Slave-CSPLight** shall enforce the *Clustering SFP* [248] when importing **Authentication Data Records and cryptographic keys** ~~TSF data~~, controlled under the SFP, from ~~outside of the TOE~~ **Master-CSPLight**.
FPT_ISA.1.2/CL	The TSF **in role Slave-CSPLight** shall use the security attributes associated with the imported **Authentication Data Records and cryptographic keys** ~~TSF data~~.
FPT_ISA.1.3/CL	The TSF **in role Slave-CSPLight** shall ensure that the protocol used provides for the unambiguous association between the security attributes and the **Authentication Data Records and cryptographic keys** ~~TSF data~~ received.
FPT_ISA.1.4/CL	The TSF in **role Slave-CSPLight** shall ensure that interpretation of the security attributes of the imported **Authentication Data Records and cryptographic keys** ~~TSF data~~ is as intended by the source of the **Authentication Data Records and cryptographic keys** ~~TSF data~~.
FPT_ISA.1.5/CL	The TSF in **role Slave-CSPLight** shall enforce the following rules when importing **Authentication Data Records and cryptographic keys** ~~TSF data~~ controlled under the SFP from ~~outside of the TOE~~ **MasterCSPLight**:

1. *TSF in role Slave-CSPLight always imports Authentication Data Records with security attributes from Master-CSPLight*
2. *TSF in role Slave-CSPLight imports cryptographic keys with security attributes from Master-CSPLight only if the security attribute Clustering of the key allows transfer* [250] .

FPT_ESA.1/CL Export of TSF data with security attributes - Cluster

Hierarchical to	No components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset informationflow control] [FMT_MTD.1 Management of TSF data, or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency
FPT_ESA.1.1/CL	The TSF **in role Master-CSPLight** shall enforce the *Clustering SFP* [248] when importing **Authentication Data Records and cryptographic keys** ~~TSF data~~, controlled under the SFP, from ~~outside of the TOE~~ **Slave-CSPLight**.
FPT_ESA.1.2/CL	The TSF **in role Master-CSPLight** shall export the **Authentication Data Records and cryptographic keys** ~~TSF data~~ with the TSF data's associated security attributes.
FPT_ESA.1.3/CL	The TSF **in role Master-CSPLight** shall ensure that the security attributes, when exported ~~outside the TOE~~ **to Slave-CSPLight**, are unambiguously associated with the exported **Authentication Data Records and cryptographic keys** ~~TSF data~~.
FPT_ESA.1.4/CL	The TSF in **role Master-CSPLight** shall enforce the following rules when **Authentication Data Records and cryptographic keys** ~~TSF data~~ is exported ~~ofrom the TOE~~ **to the Slave-CSPLight**: 1. *TSF in role Master-CSPLight exports Authentication Data Records with security attributes to any Slave-CSPLight* 2. *TSF in role Master-CSPLight exports cryptographic keys with security attributes to Slave-CSPLight only if the security attribute Clustering of the key allows transfer* [251] .

FPT_TDC.1/CL Inter-TSF basic TSF data consistency - Clustering

Hierarchical to	No components.
Dependencies	[No dependencies.

FPT_TDC.1.1/CL	The TSF shall provide the capability to consistently interpret *Authentication Data Records and cryptographic keys with their security attributes* [202] when shared between the TSF and **TOE sample in the cluster** ~~another trusted IT product~~.
FPT_TDC.1.2/CL	The TSF shall use the following rules: 1. *the TSF in Slave-CSPLight role shall interpret the imported Authentication Data Records with their security attributes in the same way as it interprets the Authentication Data Records when it exports them in Master-CSPLight role,* 2. *the TSF in Slave-CSPLight role shall interpret the imported cryptographic keys with their security attributes in the same way as it interprets the Authentication Data Records when it exports them in Master-CSPLight role,* [252] when interpreting the **Authentication Data Records and cryptographic keys** ~~TSF data~~ from **Master-CSPLight** ~~another trusted IT product~~.

6.4. Security assurance requirements

The PP requires the TOE to be evaluated according to EAL2 augmented with ALC_CMS3 (Implementation representation CM coverage) and ALC_LCD.1 (Developer-Defined Lifecycle Model), and with specific refinements on ALC_CMS.3, ADV_ARC.1 and ATE_IND.2.

6.4.1. Assurance refinements

Refinement on ALC_CMS.3.1C:

The implementation representation listed shall comprise the implementation representation of the TOE defining the TSF to a level of detail such that the compliance of the TOE and TSF to the requirements imposed by the platform guidances on which the TOE is designed to run on, can be verified by that evidence.

Refinement on ADV_ARC.1.3D:

The security guidance documentation of each platform (hardware platform and operating system) on which the TOE is designed to run shall be provided in addition.

Refinement on ADV_ARC.1.1C to 1.5C:

The security architecture description shall include an assessment how each single security requirement imposed by the platform documentation (guidance documentation and if available evaluation or certification results) has been followed in the TOE design and implementation concept.

Examples for such security requirements could include but are not limited to:

- **Dedicated library calls: Dedicated calls protecting against attacks may be provided by the platform for cryptographic operation. For example, dedicated calls implement operations that are hardened against timing side channel attacks, while others execute faster, but are not hardened. The platform guidance may require such library calls to be used.**
- **Key usage limitations: Key usage above a certain limit may reveal side channel information which can then be exploited. The implementation must ensure that the key usage limit is adhered to.**
- **Dedicated calls to ensure a correct program flow are provided (i.e. for boolean verification calls) to ensure protection against attacks that disturb the execution flow. Such library calls must be made use of in critical operations.**
- **Dedicated library calls are provided for the secure generation of cryptographic random numbers. Other random number generation functionality is present, but is not suitable to generate cryptographic random numbers. It must be ensured that correct random number generation library calls are used.**

Refinement on ADV_ARC.1.1E:

The evaluators task includes to check consistency of the requirements considered in the architectural description against those outlined in the platform documentation.

Refinement on ATE_IND.2.1D:

Providing the TOE for testing shall include in addition the implementation representation of the TOE as defined by ALC_CMS.3.

Refinement of ATE_IND.2.2C:

The resources provided shall include additionally appropriate tools or access to the TOE development environment in order to enable the evaluator to perform source code review most efficiently.

Refinement of ATE_IND.2.3E:

The evaluators test activities shall include a verification of the TOE implementation representation provided in order to confirm code compliance of the TOE implementation representation to the security guidance of the hardware platform and operating system and libraries which the TOE/TSF is intended to be run on. Therefore, the evaluator shall assess and verify that all platform guidance requirements are met and indicate possible vulnerabilities to the AVA evaluation activity for the TOE for further consideration.

6.5. Security requirements rational

6.5.1. Dependency rational

This chapter demonstrates that each dependency of the security requirements is either satisfied, or justifies the dependency not being satisfied.

Note, the column SFR components showing the concrete SFR satisfying the dependencies are typical use cases. It does not exclude that the SFR in the first column may solve dependencies of other SFR as well. E. g. the SFR FCS_CKM.1 defines requirements for ECC key generation, and a generated ECC key pair may not only be directly used for ECDSA digital signatures according to FCS_COP.1/CDS-RSA and FCS_COP.1/VDSRSA, but also for encryption and decryption of the AES key in FCS_COP.1/HEM and FCS_COP.1/HDM.

Table 8. Dependency rational

SFR	Dependencies of the SFR	SFR components
FCS_CKM.1/AES	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/ED FCS_CKM.4
FCS_CKM.1/AES_RSA	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/HEM with FCS_CKM.1/AES_RSA, FCS_CKM.4
FCS_CKM.1/ECC	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/CDS-ECDSA, FCS_COP.1/VDS-ECDSA, FCS_CKM.4
FCS_CKM.1/ECKA-EG	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/HEM with FCS_CKM.1/ECKA-EG, FCS_CKM.4
FCS_CKM.1/PACE	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/TCE, FCS_COP.1/TCM, FCS_CKM.4
FCS_CKM.1/RSA	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/CDS-RSA, FCS_COP.1/VDS-RSA FCS_CKM.4

SFR	Dependencies of the SFR	SFR components
FCS_CKM.1/TCAP	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/TCE, FCS_COP.1/TCM FCS_CKM.4
FCS_CKM.4	[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.1/ECC, FCS_CKM.1/RSA, FCS_CKM.1/ECKA-EG, FCS_CKM.1/AES_RSA, FCS_CKM.1/TCAP, FCS_CKM.1/PACE
FCS_CKM.5/AES	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/ED FCS_CKM.4
FCS_CKM.5/AES_RSA	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/HDM with FCS_CKM.5/AES_RSA, FCS_CKM.4
FCS_CKM.5/ECC	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/CDS-ECDSA, FCS_COP.1/VDS-ECDSA FCS_CKM.4
FCS_CKM.5/ECDHE	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/HEM with FCS_CKM.5/ECDHE, FCS_CKM.4
FCS_CKM.5/ECKA-EG	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/HDM with FCS_CKM.5/ECKA-EG, FCS_CKM.4
FCS_COP.1/CDS-ECDSA	[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_CKM.1/ECC, FCS_CKM.4
FCS_COP.1/CDS-RSA	[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_CKM.1/RSA, FCS_CKM.4

SFR	Dependencies of the SFR	SFR components
FCS_COP.1/DecUCP	[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	Import of UCP decryption key as TSF data with confidentiality protection FPT_TCT.1/CK and FCS_COP.1/KU, FCS_CKM.4
FCS_COP.1/ED	[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_CKM.1/AES, FCS_CKM.4
FCS_COP.1/Hash	[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	Hash functions do not use keys
FCS_COP.1/HDM	[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_CKM.5/ECKA-EG, FCS_CKM.5/AES_RSA, FCS_CKM.5/ECDHE (note deterministic FCS_CKM.5 play the role of randomized FCS_CKM.1), FCS_CKM.4
FCS_COP.1/HEM	[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_CKM.1/ECKA-EG, FCS_CKM.1/AES_RSA, FCS_CKM.5/ECDHE, FCS_CKM.4
FCS_COP.1/HMAC	[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_RNG.1 generates random strings as HMAC keys, FCS_CKM.4

SFR	Dependencies of the SFR	SFR components
FCS_COP.1/KU	[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_CKM.1/AES, FCS_CKM.4
FCS_COP.1/KW	[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_CKM.1/AES, FCS_CKM.4
FCS_COP.1/MAC	[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_CKM.1/AES, FCS_CKM.4
FCS_COP.1/TCE	[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_CKM.1/TCAP, FCS_CKM.1/PACE, FCS_CKM.4
FCS_COP.1/TCM	[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_CKM.1/TCAP, FCS_CKM.1/PACE, FCS_CKM.4
FCS_COP.1/VDS-ECDSA	[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	FPT_ISA.1/Cert (note keys are TSF data), FCS_CKM.4

SFR	Dependencies of the SFR	SFR components
FCS_COP.1/VDS-RSA	[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	FPT_ISA.1/Cert (note keys are TSF data), FCS_CKM.4
FCS_COP.1/VDSUCP	[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	Import of signature verification key of UCP Issuer as TSF data FPT_ISA.1/Cert, FPT_TIT.1.Cert, FCS_CKM.4
FCS_RNG.1	no dependencies	-
FDP_ACC.1/KM	FDP_ACF.1 Security attribute based access control	Dependency on FDP_ACF.1 is not fulfilled. Access control to key management functions are specified by FMT_MTD.1/KM because cryptographic keys are TSF data.
FDP_ACC.1/Oper	FDP_ACF.1 Security attribute based access control	FDP_ACF.1/Oper
FDP_ACC.1/UCP	FDP_ACF.1 Security attribute based access control	FDP_ACF.1/UCP
FDP_ACF.1/Oper	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.1/Oper, FMT_MSA.3/KM
FDP_ACF.1/UCP	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.1/UCP, FMT_MSA.3 is not included, because the security attributes of UCP are imported according to FDP_ITC.2/UCP without default values.
FDP_DAU.2/Att	FIA_UID.1 Timing of identification	FIA_UID.1
FDP_DAU.2/Sig	FIA_UID.1 Timing of identification	FIA_UID.1
FDP_ETC.1	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	FDP_ACC.1/Oper

SFR	Dependencies of the SFR	SFR components
FDP_ETC.2	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	FDP_ACC.1/Oper
FDP_ITC.2/UCP	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_ACC.1/UCP trusted communication is provided by FCS_COP.1/VDSUCP and FCS_COP.1/DecUCP, FPT_TDC.1/UCP
FDP_ITC.2/UD	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_ACC.1/Oper trusted communication is provided by FCS_COP.1/HDM and FCS_COP.1/VDS-*, FPT_TDC.1/CK because import of user data is intended for cryptographic operation with key.
FDP_RIP.1/UCP	No dependencies	-
FIA_AFL.1	FIA_UAU.1 Timing of authentication	FIA_UAU.1
FIA_API.1/CA	No dependencies	-
FIA_API.1/PACE	No dependencies	-
FIA_ATD.1	No dependencies	-
FIA_UAU.1	FIA_UID.1 Timing of identification	FIA_UID.1
FIA_UAU.5	No dependencies	-
FIA_UAU.6	No dependencies	-
FIA_UID.1	No dependencies	-
FIA_USB.1	FIA_ATD.1 User attribute definition	FIA_ATD.1
FMT_MOF.1	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMF.1, FMT_SMR.1
FMT_MSA.1/KM	[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	FDP_ACC.1/KM, FDP_ACC.1/Oper, FMT_SMF.1, FMT_SMR.1

SFR	Dependencies of the SFR	SFR components
FMT_MSA.2	[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	FDP_ACC.1/KM, FDP_ACC.1/Oper, FMT_MSA.1/KM, FMT_SMR.1
FMT_MSA.3/KM	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1/KM, FMT_SMR.1
FMT_MTD.1/KM	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMF.1, FMT_SMR.1
FMT_MTD.1/RAD	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMF.1, FMT_SMR.1
FMT_MTD.1/RK	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMF.1, FMT_SMR.1
FMT_MTD.3	FMT_MTD.1 Management of TSF data	FMT_MTD.1/RAD
FMT_SAE.1	FMT_SMR.1 Security roles, FPT_STM.1 Reliable time stamps	FMT_SMR.1, FPT_STM.1 (due this ST to claiming [PP-CSPL-Cluster-V1.0]).
FMT_SMF.1	No dependencies.	-
FMT_SMR.1	FIA_UID.1 Timing of identification	FIA_UID.1
FPT_ESA.1/CK	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1/KM, FMT_MTD.1/KM, FMT_MSA.1/KM, FPT_TDC.1/CK
FPT_FLS.1	No dependencies.	-
FPT_ISA.1/Cert	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1/KM, FMT_MTD.1/RK, FMT_MSA.1/KM, FPT_TDC.1/Cert

SFR	Dependencies of the SFR	SFR components
FPT_ISA.1/CK	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1/KM, FMT_MTD.1/RK, FMT_MTD.1/KM,FMT_MSA.1/KM, FPT_TDC.1/Cert
FPT_TCT.1/CK	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]	FDP_ACC.1/KM, FMT_MTD.1/RK, FMT_MTD.1/KM
FPT_TDC.1/Cert	No dependencies	-
FPT_TDC.1/CK	No dependencies	-
FPT_TDC.1/UCP	No dependencies	-
FPT_TIT.1.Cert	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]	FDP_ACC.1/KM, FMT_MTD.1/RK
FPT_TIT.1/CK	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]	FDP_ACC.1/KM, FMT_MTD.1/KM
FPT_TST.1	No dependencies	-
FTP_ITC.1	No dependencies	-
FAU_GEN.1	FPT_STM.1 Reliable time stamps	FPT_STM.1
FAU_STG.1	FAU_GEN.1 Audit data generation	FAU_GEN.1
FAU_STG.3	FAU_STG.1 Protected audit trail storage	FAU_STG.1
FDP_ACF.1/TS	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.1/Oper, FMT_MSA.3/KM
FDP_DAU.2/TS	FIA_UID.1 Timing of identification	FIA_UID.1
FDP_ETC.2/TS	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	FDP_ACC.1/Oper

SFR	Dependencies of the SFR	SFR components
FDP_ITC.2/TS	[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_ACC.1/Oper, trusted communication is provided by FCS_COP.1/HDM and FCS_COP.1/VDS-*, FTP_ITC.1, FPT_TDC.1/CK because import of user data is intended for cryptographic operation with key with appropriate security attribute "TimeStamp".
FMT_MOF.1/TSA	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMR.1/TSA and FMT_SMR.1 FMT_SMF.1/TSA
FMT_MTD.1/Audit	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMR.1/TSA and FMT_SMR.1 FMT_SMF.1/TSA
FMT_SMF.1/TSA	No dependencies	-
FMT_SMR.1/TSA	FIA_UID.1 Timing of identification	FIA_UID.1
FPT_STM.1	No dependencies	-
FPT_TIT.1/Audit	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]	FDP_ACC.1/UCP and FDP_ACC.1/KM and FDP_ACC.1/Oper if selected, FMT_MTD.1/Audit
FAU_GEN.1/CL	FPT_STM.1 Reliable time stamps	FPT_STM.1
FCS_CKM.5/CLDH	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/ED, FCS_COP.1/MAC and FCS_CKM.4
FDP_ACC.1/CL	FDP_ACF.1 Security attribute based access control	Dependency on FDP_ACF.1 is not fulfilled. Access control to key management functions are specified by FMT_MTD.1/CL because cryptographic keys are TSF data.

SFR	Dependencies of the SFR	SFR components
FMT_MTD.1/CL	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMF.1, FMT_SMR.1
FPT_ESA.1/CL	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1/CL FMT_MTD.1/CL, FMT_MTD.1/RAD and FMT_MTD.1/KM, FMT_MSA.1/KM applies for exported and imported keys and required in [PP-CSPL-V1.0] ,FPT_TDC.1/CL
FPT_ISA.1/CL	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1/CL FMT_MTD.1/CL, FMT_MTD.1/RAD and FMT_MTD.1/KM, FMT_MSA.1/KM applies for exported and imported keys and required in [PP-CSPL-V1.0], FPT_TDC.1/CL
FPT_TCT.1/CL	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_ACC.1/CL FMT_MTD.1/CL
FPT_TDC.1/CL	No dependencies	-
FPT_TIT.1/CL	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]	FDP_ACC.1/CL FMT_MTD.1/CL

6.5.2. Security functional requirements rationale

Table 9 traces each SFR back to the security objectives for the TOE.

Table 9. Security functional requirements rationale

	O.I&A	O.AuthentTOE	O.Enc	O.DataAuth	O.RBGS	O.TChann	O.AccCtrl	O.SecMan	O.TST	O.SecUpCP	O.Audit	O.TimeService	O.Cluster
FCS_CKM.1/AES			X	X				X					
FCS_CKM.1/AES_RSA			X	X				X					
FCS_CKM.1/ECC		X	X	X				X					
FCS_CKM.1/ECKA-EG			X	X				X					
FCS_CKM.1/PACE		X				X		X					
FCS_CKM.1/RSA		X	X	X				X					
FCS_CKM.1/TCAP		X				X		X					
FCS_CKM.4			X	X				X					
FCS_CKM.5/AES			X	X				X					
FCS_CKM.5/AES_RSA			X	X				X					
FCS_CKM.5/ECC			X	X				X					
FCS_CKM.5/ECDHE			X	X				X					
FCS_CKM.5/ECKA-EG			X	X				X					
FCS_COP.1/CDS-ECDSA		X		X									
FCS_COP.1/CDS-RSA		X		X									
FCS_COP.1/DecUCP										X			

	O.I&A	O.AuthentTOE	O.Enc	O.DataAuth	O.RBGS	O.TChann	O.AccCtrl	O.SecMan	O.TST	O.SecUpCP	O.Audit	O.TimeService	O.Cluster
FCS_COP.1/ED			X					X					
FCS_COP.1/Hash			X					X					
FCS_COP.1/HDM			X	X									
FCS_COP.1/HEM			X	X									
FCS_COP.1/HMAC		X		X									
FCS_COP.1/KU								X					
FCS_COP.1/KW								X					
FCS_COP.1/MAC				X									
FCS_COP.1/TCE						X							
FCS_COP.1/TCM						X							
FCS_COP.1/VDS-ECDSA				X									
FCS_COP.1/VDS-RSA				X									
FCS_COP.1/VDSUCP										X			
FCS_RNG.1					X			X					
FDP_ACC.1/KM							X	X					
FDP_ACC.1/Oper							X						

	O.I&A	O.AuthentTOE	O.Enc	O.DataAuth	O.RBGS	O.TChann	O.AccCtrl	O.SecMan	O.TST	O.SecUpCP	O.Audit	O.TimeService	O.Cluster
FDP_ACC.1/UCP										X			
FDP_ACF.1/Oper							X						
FDP_ACF.1/UCP										X			
FDP_DAU.2/Att		X											
FDP_DAU.2/Sig				X									
FDP_ETC.1				X									
FDP_ETC.2			X	X									
FDP_ITC.2/UCP										X			
FDP_ITC.2/UD			X	X									
FDP_RIP.1/UCP										X			
FIA_AFL.1	X												
FIA_API.1/CA	X	X				X							
FIA_API.1/PACE	X	X				X							
FIA_ATD.1	X							X					
FIA_UAU.1	X					X							
FIA_UAU.5	X												
FIA_UAU.6	X												
FIA_UID.1	X												
FIA_USB.1	X												
FMT_MOF.1	X					X							

	O.I&A	O.AuthentTOE	O.Enc	O.DataAuth	O.RBGS	O.TChann	O.AccCtrl	O.SecMan	O.TST	O.SecUpCP	O.Audit	O.TimeService	O.Cluster
FMT_MSA.1/KM			X	X		X	X	X					
FMT_MSA.2							X	X					
FMT_MSA.3/KM							X	X		X			
FMT_MTD.1/KM								X					
FMT_MTD.1/RAD	X												
FMT_MTD.1/RK	X		X	X				X					
FMT_MTD.3	X												
FMT_SAE.1	X												
FMT_SMF.1								X					
FMT_SMR.1	X							X					
FPT_ESA.1/CK								X					
FPT_FLS.1									X				
FPT_ISA.1/Cert	X			X				X		X			
FPT_ISA.1/CK								X					
FPT_TCT.1/CK								X		X			
FPT_TDC.1/CK			X	X				X					
FPT_TDC.1/Cert	X		X	X				X					

	O.I&A	O.AuthentTOE	O.Enc	O.DataAuth	O.RBGS	O.TChann	O.AccCtrl	O.SecMan	O.TST	O.SecUpCP	O.Audit	O.TimeService	O.Cluster
FPT_TDC.1/UCP										X			
FPT_TIT.1.Cert	X			X				X		X			
FPT_TIT.1/CK								X					
FPT_TST.1									X				
FTP_ITC.1						X							
FAU_GEN.1											.	X	
FAU_STG.1											.	X	
FAU_STG.3											.	X	
FDP_ACF.1/TS													X
FDP_DAU.2/TS											.	X	X
FDP_ETC.2/TS													X
FDP_ITC.2/TS													X
FMT_MOF.1/TSA													X
FMT_MTD.1/Audit											.	X	
FMT_SMF.1/TSA											.	X	X
FMT_SMR.1/TSA											.	X	X
FPT_STM.1											.	X	X
FPT_TIT.1/Audit											.	X	
FAU_GEN.1/CL											.	X	

	O.I&A	O.AuthentTOE	O.Enc	O.DataAuth	O.RBGS	O.TChann	O.AccCtrl	O.SecMan	O.TST	O.SecUpCP	O.Audit	O.TimeService	O.Cluster
FCS_CKM.5/CLDH												.	X
FDP_ACC.1/CL												.	X
FMT_MTD.1/CL												.	X
FPT_ESA.1/CL												.	X
FPT_ISA.1/CL												.	X
FPT_TCT.1/CL												.	X
FPT_TDC.1/CL												.	X
FPT_TIT.1/CL												.	X

The following part of the chapter demonstrates that the SFRs meet all security objectives for the TOE.

The security objective for the TOE **O.I&A** "Identification and authentication of users" is met by the following SFR:

- The SFR **FIA_ATD.1** lists the security attributes Identity, Authentication reference data and Role belonging to individual users and the SFR FMT_SMR.1 defines the security roles maintained by TSF.
- The SFR **FIA_USB.1** requires the TSF to associate the user security attributes Identity and Role with subjects acting on the behalf of that user.
- The SFR **FIA_UID.1** defines the TSF-mediated actions allowed on behalf of Unidentified User.
- The SFR **FIA_UAU.1** defines the TSF-mediated actions allowed on behalf of Unauthenticated User.
- The SFR **FIA_UAU.5** requires the TSF lists the authentication mechanisms and the rules for their application.
- The SFR **FIA_API.1/CA** and **FIA_API.1/PACE** require the TSF to authenticate external entities using Chip Authentication and PACE to communication endpoints of trusted channels.

- The SFR **FIA_UAU.6** requires the TSF to request re-authentication of users under the listed conditions.
- The SFR **FMT_MOF.1** requires the TSF to enable and disable of human user authentication.
- The SFR **FMT_MTD.1/RAD** and The SFR **FMT_MTD.1/RK** defines the management function of and the access limitation to authentication mechanisms and their TSF data including the root public keys.
- The SFR **FMT_MTD.3** enforce secure values for password mechanisms.
- The SFR **FMT_SAE.1** requires the TSF to limit the validity of user authentication and reset the security attribute Role to a values defined by an administrator according to **FMT_MTD.1/RAD**.
- The SFR **FIA_AFL.1** requires the TSF to detect and react on failed authentication attempts.
- The SFR **FPT_ISA.1/Cert** and **FPT_TIT.1.Cert** require the TSF to import certificates integrity protected and with their security attributes including those for entity authentication.
- The SFR **FPT_TDC.1/Cert** requires the TSF to interpret the certificates correctly.

The security objective for the TOE **O.AuthentTOE** "Authentication of the TOE to external entities" is met by the following SFR:

- The SFR **FCS_CKM.1/ECC**, **FCS_CKM.1/RSA** require the TSF to generate TOE authentication keys and SFR **FCS_CKM.1/PACE** and **FCS_CKM.1/TCAP** require the TSF to agree keys for authentication of the TOE to external entities.
- The SFR **FCS_COP.1/CDS-ECDSA** and **FCS_COP.1/CDS-RSA** require the TSF to generate digital signatures for authentication of the TOE to external entities.
- SFR **FCS_COP.1/HMAC** requires the TSF to generate HMAC for authentication of the TOE to external entities.
- The SFR **FIA_API.1/CA**, and **FIA_API.1/PACE** require the TSF to authenticate themselves using Chip Authentication, and PACE to communication endpoints of trusted channels.
- The SFR **FDP_DAU.2/Att** requires the TSF to generate evidence that can be used as a guarantee of the validity of attestation data to external entities.

The security objective for the TOE **O.Enc** "Confidentiality of user data by means of encryption and decryption" is met by the following SFR:

- The SFR **FCS_CKM.1/ECC** and **FCS_CKM.1/RSA** require (long term) key generation for the encryption and decryption security service of the TSF.
- The SFR **FCS_CKM.1/AES**, **FCS_CKM.1/AES_RSA**, **FCS_CKM.5/ECDHE**, and **FCS_CKM.1/ECKA-EG**, require key generation and **FCS_CKM.5/AES**,

FCS_CKM.5/AES_RSA, FCS_CKM.5/ECKA-EG and FCS_CKM.5/ECC require key derivation for encryption and decryption security service of the TSF. Note the keys must be generated or agreed with the appropriate key type for encryption respectively for decryption or in case of symmetric cryptographic mechanisms for both according to FMT_MSA.1/KM.

- The FCS_COP.1/ED requires encryption and decryption as cryptographic operations for the encryption and decryption security service of the TSF.
- The FCS_COP.1/HDM requires hybrid decryption and the SFR FCS_COP.1/HEM requires hybrid encryption and decryption as cryptographic operations for the encryption and decryption security service of the TSF.
- The SFR FDP_ETC.2 require the TSF to export encrypted user data with reference to the key and data integrity checksums for decryption and FDP_ITC.2/UD require import of encrypted user data with reference to decryption key and data integrity checksums for decryption.
- The SFR FCS_CKM.4 requires the TSF to implement secure key destruction.
- The SFR FMT_MTD.1/RK requires the TSF management of root keys for key hierarchy known to the TSF if used for encryption.
- The SFR FPT_TDC.1/Cert requires the TSF to interpret consistently the security attributes of certificates (including those used for encryption and decryption).
- The SFR FPT_TDC.1/CK requires the TSF to interpret consistently the security attributes of keys (including those used for encryption and decryption).

The security objective for the TOE O.DataAuth "Data authentication by cryptographic mechanisms" is met by the following SFR:

- The SFR FCS_CKM.1/ECC and FCS_CKM.1/RSA require (long term) key generation for the signature security service of the TSF. The SFR FCS_CKM.1/AES, FCS_CKM.1/ECKA-EG, FCS_CKM.1/AES_RSA require key generation and FCS_CKM.5/AES_RSA, FCS_CKM.5/ECDHE, FCS_CKM.5/ECC, FCS_CKM.5/ECKA-EG key derivation for MAC generation and verification. Note the keys must be generated or agreed with the appropriate key type for signature-creation, signature-verification or, in case of symmetric cryptographic mechanisms for data authentication according to FMT_MSA.1.KM.
- The SFR FDP_ETC.2 require the TSF to export signed data with and signature and public key reference for signature verification and FDP_ITC.2/UD import of signed data with signature and public key reference for signature verification. The SFR FDP_ETC.1 require the TSF to export successfully MAC verified and decrypted ciphertext as plaintext according to FCS_COP.1/HDM without the user data's associated security attributes:
- The SFR FCS_COP.1/Hash requires the TSF to implement cryptographic primitive hash function used for HMAC, cf. FCS_COP.1/HMAC, digital signature creation, cf.

FCS_COP.1/CDS-* and digital signature verification, cf. FCS_COP.1/VDS-*.

- The **FCS_COP.1/CDS-ECDSA** and **FCS_COP.1/CDS-RSA** require asymmetric cryptographic mechanisms for signature-creation.
- The SFR **FCS_COP.1/VDS-ECDSA** and **FCS_COP.1/VDS-RSA** require asymmetric cryptographic mechanisms for signature-verification.
- The SFR for keyed hash **FCS_COP.1/HMAC** and block cipher based MAC **FCS_COP.1/MAC** require the TSF to provide symmetric data integrity mechanisms.
- The SFR **FCS_COP.1/HEM** requires hybrid MAC calculation and **FCS_COP.1/HDM** requires hybrid MAC verification for the ciphertext as security service of the TSF.
- The SFR **FPT_ISA.1/Cert** requires import of certificates with security attributes and integrity protection according to **FPT_TIT.1.Cert**.
- The SFR **FCS_CKM.4** requires the TSF to implement secure key destruction.
- The SFR **FPT_TDC.1/Cert** requires the TSF to interpret consistently the security attributes in certificates (including those used for data authentication).
- The SFR **FPT_TDC.1/CK** requires the TSF to interpret consistently the security attributes keys (including those used for data authentication).

The security objective for the TOE **O.RBGS** "Random bit generation service" is met directly by the SFR FCS_RNG.1 as providing random bits for the service to the user.

The security objective for the TOE **O.TChann** "Trusted channel" is met by the following SFR:

- The SFR **FTP_ITC.1** requires different types of trusted channel depending on the capability of the other endpoint. The cases are defined in table 4. The remote entity and the TOE may use mutual authentication and key agreement by means of PACE according to **FCS_CKM.1/PACE**, shall provide integrity protection according to **FCS_COP.1/TCM** and may support confidentiality of the communication data according to **FCS_COP.1/TCE**. The cases 3 requires support of trusted channel with mutual authentication by **FIA_API.1/CA**, **FIA_UAU.5**, key agreement TCAP according to **FCS_CKM.1/TCAP**, encryption and MAC data authentication.
- The TOE authenticate themselves according to **FIA_API.1/PACE** in case of PACE. It authenticates themselves according to **FIA_API.1/CA** in case of TCAP as Proximity Integrated Circuit Card (PICC).
- The SFR **FMT_MOF.1** limits the configuration of the trusted channel according to **FTP_ITC.1.3** to an administrator.
- The SFR **FMT_MSA.1/KM** describe the requirements for management of key security attributes for these mechanisms.

The security objective for the TOE **O.AccCtrl** "Access control" is met by the following SFR:

- The SFR **FIA_ATD.1** defines the security attributes of individual users including Role which is used for access control according to **FDP_ACF.1/Oper**.
- The SFR **FDP_ACC.1/Oper** describes the subset access control for the Cryptographic Operation SFP.
- The SFR **FDP_ACF.1/Oper** defines the access control rules of the Cryptographic Operation SFP.
- The Cryptographic Operation SFP is defined by means of security attributes managed according to the SFR **FMT_MSA.1/KM**, **FMT_MSA.2** and **FMT_MSA.3/KM**.

The security objective for the TOE **O.SecMan** "Security management" is met by the following SFR:

- The SFR **FIA_ATD.1** defines the security attributes of individual users including Role which is used to enforce the Key Management SFP.
- The SFR **FDP_ACC.1/KM** defines subjects, objects and operations of the Key Management SFP.
- The SFR **FMT_SMF.1** lists the security management functions provided by the TSF.
- The SFR **FMT_SMR.1** lists the security role supported by the TOE especially the administrator and – if supported - Crypto-Officer responsible for key management.
- The SFR **FCS_CKM.1/AES**, **FCS_CKM.1/ECC**, **FCS_CKM.1/ECKA-EG**. **FCS_CKM.1/PACE**, **FCS_CKM.1/RSA**, **FCS_CKM.1/AES_RSA**, **FCS_CKM.1/TCAP** require the TSF to implement key generation function according to the assigned standards.
- The SFR **FCS_CKM.5/ECDHE** require the TSF to implement key agreement function according to the assigned standards.
- The SFR **FCS_CKM.5/AES** and **FCS_CKM.5/ECKA-EG** require the TSF to implement key derivation function according to the assigned standards.
- The SFR **FCS_CKM.1/AES_RSA** and **FCS_CKM.5/AES_RSA** require the TSF to implement AES session key generation function with RSA key encryption respective RSA key decryption and AES key derivation according to the assigned standards.
- The SFR **FCS_RNG.1** requires the TSF to implement a random number generator for key generation, key agreement functions and cryptographic operations.
- The SFR **FCS_COP.1/ED** requires the TSF to provide encryption and decryption according to AES which may be used for key management.
- The SFR **FCS_COP.1/Hash** requires the TSF to implement cryptographic primitive hash function for key derivation, cf. FCS_CKM.5.

- The SFR **FPT_ISA.1/CK** requires import and **FPT_ESA.1/CK** the export of cryptographic keys with security attributes and protection of confidentiality according to SFR **FPT_TCT.1/CK** and integrity protection according to **FPT_TIT.1/CK**.
- The SFR FPT_ISA.1.Cert requires import of certificates with security attributes and integrity protection according to **FPT_TIT.1.Cert**.
- The SFR **FPT_TDC.1/Cert** requires consistent interpretation of certificate's content. The SFR FPT_TDC.1/ CK requires consistent interpretation of security attributes imported with the key.
- The SFR **FCS_COP.1/KW** and **FCS_COP.1/KU** require the TSF key wrapping and unwrapping for key management.
- The SFR **FCS_CKM.4** requires the TSF to implement secure key destruction.
- The SFR **FMT_MSA.1/KM** and **FMT_MSA.3/KM** limit the setting of default values and specification of alternative initial values for security attributes of cryptographic keys to administrators. The SFR **FMT_MSA.1/KM** prevents modification or deletion of security attributes of keys.
- **FMT_MSA.2** enforce secure values for security attributes.
- The SFR **FMT_MTD.1/KM** and **FMT_MTD.1/RK** restricts the management of cryptographic keys espacially the import of root public keys to specifically authorized users.

The security objective for the TOE **O.TST** "Self-test" is directly met by the SFR FPT_TST.1 and FPT_FLS.1. The TSF shall preserve a secure state if self test fails.

The security objective for the TOE **O.SecUpCP** "Secure import of Update Code Package" is met by the following SFR:

- The SFR **FDP_ACC.1/UCP** and **FDP_ACF.1/UCP** requires the TSF to provide access control to enforce SFP Update. Note the verification of the authenticity of UCP and decryption of authentic UCP are performed under control of the TSF.
- The SFR **FCS_COP.1/VDSUCP** requires the verification of digital signature of the Issuer and **FCS_COP.1/DecUCP** requires decryption of authentic of UCP.
- The SFR **FDP_ITC.2/UCP** requires the TSF to import UCP as user data with security attributes if the authenticity of UCP is successful verified.
- The SFR **FPT_TDC.1/UCP** requires the TSF to import consistently the security attributes of the UCP.
- The SFR **FMT_MSA.3/KM** requires to provide restrictive initial security attributes to enforce the SFP Update.
- The SFR **FDP_RIP.1/UCP** requires the TSF to remove the received UCP after unsuccessful

verification of its authenticity.

- The UCP signature verification key may be updated according to **FPT_ISA.1/Cert** with integrity protection according to **FPT_TIT.1.Cert**.
- The UCP decryption key may be updated with confidentiality protection according to **FPT_TCT.1/CK** with **FCS_COP.1/KU**.

The security objective for the TOE **O.TimeService** "Time services" is met by the following SFR:

- The SFR **FPT_STM.1** requires the TSF to provide time stamps for the real time service.
- The SFR **FDP_DAU.2/TS** requires the TSF to provide cryptographic protected time stamps for time stamp service supported by **FCS_COP.1/CDS-ECDSA** resp. **FCS_COP.1/CDS-RSA** for signature creation defined in the Base-PP.
- The SFR **FDP_ACF.1/TS** defines access control on time stamp service to enforce the Cryptographic Operation SFP defined in **[PP-CSPL-V1.0]**.
- The SFR **FDP_ITC.2/TS** for user data import with security attributes indicating the signature key for time stamps.
- The SFR **FDP_ETC.2/TS** requires the TSF to export user data with time stamps.
- The SFR **FMT_SMF.1/TSA** defines the management functions and **FMT_SMR.1/TSA** the roles for the time service and the time stamp service additional to those defined in **[PP-CSPL-V1.0]**.
- The SFR **FMT_MOF.1/TSA** defines the management of the time service and the time service TSF.

The security objective for the TOE **O.Audit** "Audit" is met by the following SFR:

- The SFR **FAU_GEN.1** requires the TSF to generate the audit records of auditable events.
- The SFR **FAU_STG.1** and **FAU_STG.3** requires the TSF to protect and to prevent loss of audit records.
- The SFR **FMT_MTD.1/Audit** restricts the ability to export and to delete exported audit records to an Administrator. It prevents undetected deletion of audit records by generation of an audit record about deletion. The export, clear and selection of events causing audit data as management TSF data is an auditable event, cf. **FAU_GEN.1**, clause (11).
- The SFR **FPT_TIT.1/Audit** requires the TSF to protect audit records when transmitted and time when imported.
- The SFR **FMT_SMF.1/TSA** defines the management functions and FMT_SMR.1/TSA the roles for the audit TSF additional to those defined in **[PP-CSPL-V1.0]**.

- The SFR **FMT_MOF.1/TSA** requires the TSF to provide the capability to define the auditable events in clause (3) and the behaviour of automatic export of audit records in clause (4).
- The SFR **FPT_TIT.1/Audit** defines the TSF data integrity transfer protection for the audit functionality
- The SFR **FPT_STM.1** requires the TSF to provide time stamps being part of the audit records

The security objective for the TOE **O.Audit** "Audit" is met by the SFR **FAU_GEN.1** in **[PP-CSPL-TS-AU-V1.0]** and additionally by SFR **FAU_GEN.1/CL** to generate the audit records of auditable events for clustering.

The security objective for the TOE O.Cluster "Cluster" is met by the following SFR:

- The SFR **FDP_ACC.1/CL** defines subjects, objects and operations of the Clustering SFP.
- The SFR **FMT_MTD.1/CL** restricts the management of TSF data Authentication Data Records and cryptographic key by initiating the cluster to an administrator, and export and import of TSF data to an authorised identified role.
- The SFRs **FPT_ESA.1/CL** and **FPT_ISA.1/CL** require that export and import of TSF data is performed with security attributes.
- The SFR **FPT_TCT.1/CL** requires protection of confidentiality and the SFR **FPT_TIT.1/CL** the protection of integrity of the TSF data when transferred between Master-CSPLight and Slave-CSPLight.
- The SFR **FCS_CKM.5/CLDH** requires the TSF to agree on cryptographic keys. Note, **[PP-CSPL-V1.0]** defines the SFRs **FCS_COP.1/ED** and **FCS_COP.1/MAC** for encryption and MAC of the transferred TSF data.
- The SFR **FPT_TDC.1/CL** requires the TSF interpret consistently the TSF exchanged between TOE samples of the cluster.

6.5.3. Security assurance requirements rationale

Developers and users require for the TOE a low to moderate level of independently assured security in the absence of ready availability of the complete development record.

The EAL2 was chosen because it provides assurance by a full security target and an analysis of the SFRs in that ST, using a functional and interface specification, guidance documentation and a basic description of the architecture of the TOE, to understand the security behaviour. The analysis is supported by independent testing of the TSF, evidence of developer testing based on the functional specification, selective independent confirmation of the developer test results, and a vulnerability analysis (based upon the functional specification, TOE design, security architecture description and

guidance evidence provided) demonstrating resistance to penetration attackers with a basic attack potential. EAL2 also provides assurance through use of a configuration management system and evidence of secure delivery procedures.

ALC_CMS.3 has been augmented to include the implementation representation as needed for ADV_ARC and ATE_IND refinements and to get evidence that the implementation representation provided is the one of the TOE. This means that the implementation representation is part of the configuration list.

CSPLight usually requires an initial configuration and/or the installation of key material and trust certificates. Hence, ALC_LCD.1 has been augmented such that the lifecycle of the TOE is defined by the developer and thus made explicit.

For getting confidence that the platform of the TOE (operational environment) is used by the TOE in a way that the requirements on security as outlined in the platform documentation (guidance documentation and if available evaluation or certification results) have been followed in the TOE design and implementation, refinements of ADV_ARC, ATE_IND and ALC_CMS have been defined.

The goal is to ensure that the TOE implementation does not include obvious vulnerabilities caused by incorrect use of the platform and that all relevant platform guidance requirements are adhered to. Therefore, only those requirements have to be considered that are related to the TOE functionality and security claims of the Security Target of the TOE.

The refinement of ADV_ARC ensures that the developer outlines how he has considered the requirements from the platform within his TOE security architecture and design concept. The evaluators task is to check consistency of the requirements considered against those outlined in the platform documentation.

As a second step of verification that the relevant platform requirements have been considered correctly, the independent evaluator activity at ATE_IND has been refined. The evaluator has to perform a specific „source code review“, by means of cross check of the requirements from the platform to the implementation representation of the TOE by examine the implementation representation of the TOE using appropriate tools and the evidence from ADV_ARC.

[1] [assignment: access control SFP]

[2] [selection: Administrator, Crypto-Officer]

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

[4] [assignment: access control SFP, information flow control SFP]

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

[6] [assignment: list of security attributes]

[7] [selection: Administrator, Crypto-Officer]

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

[9] [assignment: list of security attributes]
[10] [assignment: the authorised identified roles]
[11] [selection: change_default, query, modify, delete, [assignment: other operations]]
[12] [assignment: list of security attributes]
[13] [assignment: the authorised identified roles]
[14] [selection: change_default, query, modify, delete, [assignment: other operations]]
[15] [assignment: list of security attributes]
[16] [selection: Administrator, Crypto-Officer, Key Owner]
[17] [selection: change_default, query, modify, delete, [assignment: other operations]]
[18] [assignment: list of security attributes]
[19] [selection: Administrator, Crypto-Officer, Key Owner]
[20] [assignment: access control SFP, information flow control SFP]
[21] [selection, choose one of: restrictive, permissive, [assignment: other property]]
[22] [selection: Administrator, Crypto-Officer]
[23] [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
[24] [assignment: list of TSF data]
[25] [selection: Administrator, Crypto-Officer, Key Owner]
[26] [selection: change_default, query, modify, delete, clear,[assignment: other operations]]
[27] [assignment: list of TSF data]
[28] [selection: Administrator, Crypto-Officer, Key Owner]
[29] [selection: change_default, query, modify, delete, clear,[assignment: other operations]]
[30] [assignment: list of TSF data]
[31] [selection: Administrator, Crypto-Officer, Key Owner]
[32] [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
[33] [assignment: list of TSF data]
[34] [selection: Administrator, Crypto-Officer, Key Owner]
[35] [assignment: list of cryptographic operations]
[36] [assignment: cryptographic algorithm]
[37] [assignment: cryptographic key sizes]
[38] [assignment: list of standards]
[39] “create” denotes initial setting a root key
[40] [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
[41] [assignment: list of TSF data]
[42] [selection: Administrator, Crypto-Officer]
[43] [selection: change_default, query, modify, delete, clear,[assignment: other operations]]
[44] [assignment: list of TSF data]
[45] [selection: Administrator, Crypto-Officer]
[46] [assignment: access control SFP, information flow control SFP]
[47] [selection: transmit, receive, transmit and receive]
[48] [selection: modification, deletion, insertion, replay]
[49] [assignment: additional importation control rules]
[50] [assignment: list of TSF data types]
[51] [assignment: list of interpretation rules to be applied by the TSF]

[52] [selection: _physical, non-physical true, deterministic, hybrid physical, hybrid deterministic_]
[53] [assignment: _list of security capabilities_]
[54] [assignment: _a defined quality metric_]
[55] [assignment: cryptographic key generation algorithm]
[56] [selection: *256 bits, [assignment: additional cryptographic key sizes > 128 bits]*]
[57] [assignment: list of standards]
[58] [assignment: key type]
[59] [assignment: _input parameters_]
[60] [assignment: cryptographic key derivation algorithm]
[61] [selection: *256 bits, [assignment: additional cryptographic key sizes > 128 bits]*]
[62] [selection: elliptic curves in <<Table.CurvesKeySizesStandards>>]
[63] [selection: key size in <<Table.CurvesKeySizesStandards>>]
[64] [selection: standards in <<Table.CurvesKeySizesStandards>>]
[65] [assignment: input parameters]
[66] [selection: elliptic curves in <<Table.CurvesKeySizesStandards>>]
[67] [assignment: KDF]
[68] [selection: key size in <<Table.CurvesKeySizesStandards>>]
[69] [selection: standards in <<Table.CurvesKeySizesStandards>>]
[70] [cryptographic key sizes]
[71] [selection: AES-256, none other]
[72] [assignment: input parameters]
[73] [selection: elliptic curves in <<Table.CurvesKeySizesStandards>>]
[74] [selection: DH group in <<Table.DiffieRecomms>>]
[75] [selection:256 bits, none other]
[76] [elliptic curves in <<Table.CurvesKeySizesStandards>>]
[77] [key sizes in <<Table.CurvesKeySizesStandards>>]
[78] [standards in <<Table.CurvesKeySizesStandards>>]
[79] [selection: AES-256, none other]
[80] [assignment: input parameters]
[81] [selection: elliptic curves in <<Table.CurvesKeySizesStandards>>]
[82] [selection: AES-256, none other]
[83] [_256 bits, none other_]
[84] [selection: _AES-256, none other]
[85] [assignment: _input parameters_]
[86] [_256 bits, none other_]
[87] [assignment: cryptographic key destruction method]
[88] [assignment: list of cryptographic operations]
[89] [selection: KW, KWP]
[90] [selection: 256 bits, none other]
[91] [selection: KW, KWP]
[92] [assignment: cryptographic algorithms]
[93] [assignment: access control SFP, information flow control SFP]
[94] [selection: transmit, receive, transmit and receive]

[95] [selection: modification, deletion, insertion, replay]

[96] [assignment: additional importation control rules]

[97] [assignment: list of TSF data types]

[98] [assignment: list of interpretation rules to be applied by the TSF]

[99] [assignment: additional exportation control rules]

[100] [selection: AES-256, no other algorithm]

[101] [selection: CTR, OFB, CFB, no other]

[102] [_selection: 256 bits, no other key size_]

[103] [selection: FCS_CKM.1/ECKA-EG, FCS_CKM.1/AES_RSA, FCS_CKM.5/ECDHE]

[104] [selection: AES-256, none other]

[105] [selection: CBC <<NIST-SP800-38A>>, CCM <<NIST-SP800-38C>>, GCM <<NIST-SP800-38D>>]

[106] [selection: CMAC <<NIST-SP800-38B>>, GMAC <<NIST-SP800-38D>>, HMAC <<RFC-2104>>]

[107] [selection: 256 bits, no other key size]

[108] [selection: FCS_CKM.5/ECDHE, FCS_CKM.5/ECKA-EG, FCS_CKM.5/AES_RSA]

[109] [selection: CMAC <<NIST-SP800-38B>>, GCM <<NIST-SP800-38D>>, HMAC <<RFC-2104>>]

[110] [selection: AES-128, AES-256]

[111] [selection: CBC <<NIST-SP800-38A>>, CCM <<NIST-SP800-38C>>, GMAC <<NIST-SP800-38D>>]

[112] [selection: 256 bits, no other key size]

[113] [selection: AES-256, none other]

[114] [selection: GMAC <<NIST-SP800-38D>>, no other]

[115] [selection: 256 bits, no other key size]

[116] [selection: HMAC-SHA-1, HMAC-SHA384, no other]

[117] [selection: elliptic curves in <<Table.CurvesKeySizesStandards>>]

[118] [selection: key size in <<Table.CurvesKeySizesStandards>>]

[119] [selection: standards in <<Table.CurvesKeySizesStandards>>]

[120] [selection: elliptic curves in <<Table.CurvesKeySizesStandards>>]

[121] [selection: key size in <<Table.CurvesKeySizesStandards>>]

[122] [selection: standards in <<Table.CurvesKeySizesStandards>>]

[123] [assignment: cryptographic key sizes]

[124] [assignment: cryptographic key sizes]

[125] [assignment: list of objects or information types]

[126] [selection: FCS_COP.1/CDS-RSA, FCS_COP.1/CDS-ECDSA]

[127] [assignment: list of subjects]

[128] [assignment: authentication mechanism]

[129] [assignment: object, authorized user or role]

[130] [selection: _FCS_COP.1/CDS-RSA, FCS_COP.1/CDS-ECDSA, ECDAA according to +[selection: <<TPMLib-P1>><<FIDO-ECDAA>>]

[131] [assignment: other cryptographic authentication mechanisms]

[132] [selection: *_logically separated from other communication channels, using physical separated ports_*]

[133] [selection: *_Authentication of the TOE and remote entity according to the case in <<Table.OperationInSRF>>_*]

[134] [assignment: *_according to the case in <<Table.OperationInSRF>>_*]

[135] [selection: *_cryptographic operation according to the case in <<Table.OperationInSRF>>*_]

[136] [selection: the TSF, the remote trusted IT product]

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

[138] [selection: elliptic curves in <<Table.CurvesKeySizesStandards>>]

[139] [selection: _128 bits, 192 bits, 256 bits_]

[140] [selection: _128 bits, 192 bits, 256 bits_]

[141] [selection: CBC <<NIST-SP800-38A>>, CCM<<NIST-SP800-38C>>, GCM <<NIST-SP800-38D>>]

[142] [selection: _128 bits, 192 bits, 256 bits_]

[143] [selection: CMAC <<NIST-SP800-38B>>, GMAC <<NIST-SP800-38D>>]

[144] [selection: _128 bits, 192 bits, 256 bits_]

[145] [selection: change_default, query, modify, delete, clear, [assignment: other operations]]

[146] [assignment: list of TSF data]

[147] [selection: Administrator, User Administrator]

[148] [assignment: the authorised identified roles]

[149] [selection:Administrator, User Administrator]

[150] [selection: Administrator, User Administrator]

[151] [assignment: time frame]

[152] [selection: Administrator, User Administrator]

[153] selection: Unidentified user, Unauthenticated user]

[154] [selection: Administrator, User Administrator]

[155] [selection: [assignment: *positive integer number*], an ~~administrator~~ ***[selection: Administrator, User Administrator]***] *configurable positive integer within [assignment: range of acceptable values]*

[156] [assignment: _list of authentication events_]

[157] [selection: _met, surpassed_]

[158] [assignment: _list of actions_]

[159] [assignment: rules for the initial association of attributes]

[160] [assignment: list of user security attributes]

[161] [assignment: rules for the changing of attributes]

[162] [assignment: list of security attributes for which expiration is to be supported]

[163] [selection: Administrator, User Administrator]

[164] [assignment: list of actions to be taken for each security attribute]

[165] [assignment: list of other TSF-mediated actions]

[166] [selection: a role, a set of role]

[167] [assignment: list of other TSF mediated actions]

[168] [assignment: list of multiple authentication mechanisms]

[169] [assignment: additional rules]

[170] [assignment: list of other conditions under which re-authentication is required]

[171] [assignment: additional importation control rules]

[172] [assignment: additional exportation control rules]

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

[174] [selection: Administrator, Crypto-Officer]

[175] [assignment: other roles]

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

[177] [selection: Administrator, Crypto-Officer]

[178] [assignment: other roles]

[179] [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]

[180] [selection: Administrator, Crypto-Officer]

[181] [assignment: other rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]

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

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

[184] [selection: Administrator, Crypto-Officer, Key Owner]

[185] [selection: Administrator, Crypto-Officer, Key Owner]

[186] [assignment: additional list of security management functions to be provided by the TSF]

[187] [selection: _Administrator, Crypto-Officer, User Administrator, Update Agent, Timekeeper_]

[188] [selection: [*assignment: other roles]*, *no other roles*]

[189] [assignment: additional security attributes]

[190] [selection: determine the behaviour of, disable, enable, modify the behaviour of]

[191] [assignment: list of functions]

[192] [selection: Administrator, User Administrator]

[193] [selection: Administrator, User Administrator]

[194] [selection: Administrator, User Administrator]

[195] [selection: Administrator, User Administrator]

[196] assignment: list of types of additional failures

[197] [selection: during initial start-up, periodically during normal operation, at the request of the authorised user, at the conditions[assignment: conditions under which self test should occur]]

[198] [assignment: _parts of TSF_]

[199] [selection: [assignment: parts of TSF data], TSF data]

[200] [assignment: parts of TSF], TSF]

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

[202] [assignment: list of TSF data types]

[203] [selection: Administrator, Update Agent]

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

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

[206] [assignment: _rules, based on security attributes, that explicitly deny access of subjects to objects_]

[207] [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]

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

[209] [selection: allocation of the resource to, deallocation of the resource from]

[210] [assignment: list of objects]

[211] [assignment: list of subjects]

[212] [assignment: list of objects or information types]

[213] [selection: FCS_COP.1/CDS-ECDSA, FCS_COP.1/CDS-RSA]

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

[215] [assignment: additional exportation control rules]

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

[217] [assignment: other roles]

[218] [assignment: access control SFP]

[219] [assignment: other roles]

[220] [assignment: other rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]

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

[222] [assignment: list of management functions to be provided by the TSF]

[223] [selection: Auditor, Timekeeper, no other roles]

[224] [selection: Administrator, Timekeeper]

[225] [selection: Administrator, Timekeeper]

[226] [selection: Administrator, Auditor]

[227] [selection: choose one of: minimum, basic, detailed, not specified]

[228] [selection: (4) Generation of (selected types of) signature key pairs (all FCS_CKM.1 instantiations for generation of permanent stored keys) (5) Execution of (selected types of) cryptographic operation (all FCS_COP.1 instantiations), (6) Cryptographic key destruction (FCS_CKM.4) of permanent stored keys, (7) Failure with preservation of secure state (FPT_FLS.1): entering and exiting secure state, (8) Management of security functions (FMT_MOF.1, FMT_MOF.1/TSA) , (9) Management of TSF data (FMT_MTD.1/AUDIT): Export, clear and selection of events causing audit data, (10) No other event], (11) [assignment: additional specifically defined auditable events]]

[229] [assignment: other audit relevant information]

[230] [selection: choose one of: prevent, detect]

[231] [assignment: actions to be taken in case of possible audit storage failure]

[232] [assignment: pre-defined range]

[233] [assignment: actions to be taken in case of possible audit storage failure]

[234] [settable percentage of storage capacity]

[235] [assignment: approximate deviation]

[236] [selection: Administrator, Timekeeper]

[237] **[selection: (1)internal clock with accuracy [assignment: approximate deviation] with the ability of adjustment of the clock by the [selection: Administrator,Timekeeper], (2)internal clock with accuracy [assignment: approximate deviation] with automatic adjustment of the clock by an externally trustable source in a cryptographically verifiable manner (e.g. by signed Network Time Protocol) and the ability of adjustment of the clock by the [selection: Administrator, Timekeeper].**

[238] [selection: Key Management SFP, Cryptographic Operation SFP]

[239] [selection: transmit, receive, transmit and receive]

[240] [selection: modification, deletion, insertion, replay]

[241] [assignment: other specifically defined auditable events]

[242] [assignment: other audit relevant information]

[243] [selection: Application Component, Administrator, Crypto-Officer]

[244] [assignment: input parameters]

[245] [selection: elliptic curves in the <<Table.CurvesKeySizesStandards>>]

[246] [selection: key size in the <<Table.CurvesKeySizesStandards>>]

[247] [selection: standards in the <<Table.CurvesKeySizesStandards>>]

[248] [assignment: access control SFP, information flow control SFP]

[249] [selection: transmit, receive, transmit and receive]

[250] [assignment: additional importation control rules]

[251] [assignment: additional importation control rules]

[252] [assignment: list of interpretation rules to be applied by the TSF]

7. TOE Summary Specification

The following sections describe, how the TOE meets each SFR. Therefore, the SFRs are assigned to Security Features (SF) as listed in **Logical Scope of the TOE**.

7.1. SF 1: Key Management

The TOE provides several key management functionalities, including:

- Management of key security attributes,
- Hash Generation,
- Certificate Management
- Random Number Generation, and
- Key Operations (Generation/ Derivation/ Agreement/ Destruction/ & Wrapping).

Those will be discussed on their own in the following sections. Due to the complexity of this SF, the corresponding SFR mapping will be done for each category listed before.

Management of key security attributes

The TOE have to implement several management functionalities with regards to cryptographic keys, including an access control security functional policy on the corresponding cryptographic keys. This policy governs all functions of the TOE that use (or reslt in) a key, including but not limited to key creation, key derivation, key deletion, key property modification, key import and key export. Additonally, this SF supports further SFs listed below.

In summary, this SF category is related to **FDP_ACC.1/KM**, **FMT_MSA.1/KM**, **FMT_MSA.3/KM** and **FMT_MTD.1/KM**.

Hash Generation

Hashing is used by the TOE internally for HMAC, Key Derivation, timestamping, digital signature creation and verification. Additionally there exists an external interface allowing the hashing of arbitrary data by the TOE. Overall, the TOE supports hashing of data using

- SHA-256,
- SHA-384, and
- SHA-512.

In summary, this SF category is related to **FCS_COP.1/Hash**

Certificate Management

The TOE supports the folloing certificate management capabilities:

- Management of root public key of a PKI
- Verification of the integrity of certificates
- Import of TSF data from valid certificates

The TOE imports and verifies certificates to extract the contained information about cryptographic public keys. Whenever a certificate is imported into the TOE, its validity is verified. This check includes (except for root certificates) a check of the validity of the complete certificate chain which must end in a previously imported root certificate. The extracted keys are then stored in the TOE with security attributes based on the imported certificate. (The TOE uses the validity of the certificate, its public key and the usage type of the key. All other information of the certificate will be ignored)

In summary, this SF category is related to **FMT_MTD.1/RK**, **FPT_TIT.1.Cert**, **FPT_ISA.1/Cert**, and **FPT_TDC.1/Cert**.

Random Number Generation

The TOE implements a deterministic random number generator that is compliant with the requirements of class DRG.3 of **[RNG_classes]**. The implementation follows the example of the iterated hash RNG in Example 39 of **[SP800]** which is further defined in [21]. The RNG is seeded by obtaining random data from its hardware platform from a True RNG.

In summary, this SF category is related to **FCS_RNG.1**.

Key Operations

The service *Key generation* is available to users in the role *Crypto-Officer* only.

The TOE supports key generation from entropy for:

- AES (128, 256 bit),
- Elliptic Curve Cryptography (SECP256r1), and
- RSA (4096 bit).

Additionally the TOE supports key derivation for:

- AES (128, 256 bit) - from a passphrase,
- AES (128, 256 bit) - from a rsa encrypted seed,
- AES (128, 256 bit) - from a shared secret optained by key agreement (ECKGA),

- Elliptic Curve Cryptography (SECP256r1) - from a passphrase, and
- Elliptic Curve Cryptography (SECP256r1) - by key agreement (ECDHE).

The *key generation* uses the RNG functionality provided by the TOE (see above).

The TOE also supports *key destruction*, which allows the zeroization of a key. While all secret and privateKey information is subject to this once the key is no longer needed in memory, users in the role *Crypto-Officer* are permitted to use this functionality on permanently stored key. (Note Nils: PACE-Pinned Keys sind keys die der USER_ADMIN permanent zerstört).

Furthermore, the TOE supports *key wrapping* for secure key export or key import respectively. Exporting a key without encrypting it firstly with a generated/derived wrapping key is not supported by the TOE. Additionally the TOE only allows the use of wrapping keys that are at least as strong as the keys they are meant to protect. Both of these steps necessary for exporting a key are only executable by an authenticated user in the role od a *Crypto-Officer*.

As described above, the TOE supports the follwing key agreements:

- Elliptic Curve Diffie-Hellman, and
- ECKA-EG.

In summary, this SF category is related to **FCS_CKM.1/AES**, **FCS_CKM.5/AES**, **FCS_CKM.1/ECC**, **FCS_CKM.5/ECC**, **FCS_CKM.1/RSA**, **FCS_CKM.5/ECDHE**, **FCS_CKM.1/ECKA-EG**, **FCS_CKM.5/ECKA-EG**, **FCS_CKM.1/AES_RSA**, **FCS_CKM.5/AES_RSA**, **FCS_CKM.4**, **FCS_COP.1/KW**, **FCS_COP.1/KU**, **FPT_TCT.1/CK**, **FPT_TIT.1/CK**, **FPT_ISA.1/CK**, **FPT_TDC.1/CK**, and **FPT_ESA.1/CK**

7.2. SF2: Cryptographic operations for encryption & decryption

The TOE supports symmetric data encryption and decryption using AES in CBC mode with cryptographic key lengths of 128 and 256 bits in accordance with **FCS_COP.1/ED**. This functionality is primally used to be provided by the TOE to external entities for encryption and decryption. Additionally, this functionality is used by the TOE internally for data transfer protection in a CSPLight cluster setup (cf. **FPT_TCT.1/CL**). Each encryption uses an initialization vector which is generated using a new random data, or in the case of the secure channel using an internally maintained counter.

This SF is dependent on the prior generation of a key as provided by **SF1: Key Management**. As mentioned, the *key generation* requires a user in the role *Crypto-Officer*. Furthermore, the access to the secret key required for encryption and decryption requires either the *Crypto-Officer* or the *Key Owner* role.

In summary, this SF is related to the following SFRs: **FCS_COP.1/ED**.

7.3. SF3: Hybrid encryption for user data

To combine the advantages of asymmetric and symmetric encryption, the TOE provides hybrid data encryption. The symmetric encryption part always performed by using an Authenticated Encryption scheme using the *Encrypt-then-MAC* semantic as required by the corresponding SFRs. The TOE uses *AES-CBC encryption* with either *HMAC* or *CMAC* authentication (see below).

As defined in **SF2: Cryptographic operations for encryption & decryption** the functions for decrypting data that are provided via external interfacses and requires either a user in the role *Key Owner*, and depend on the prior generation of a key as provided by **SF1: Key Management**.

In summary, this SF is related to the following SFRs: **FCS_COP.1/HEM** and **FCS_COP.1/HDM**.

7.4. SF4: Data integrity mechanisms

As part of *SF3: Hybrid encryption for user data*, the TOE supports data integrity protection via symmetric cryptography, defined below. Additionally the TOE also supports data integrity protection via asymmetric cryptography (digital signature creation/verification) The signature creation and verification with timestamping keys may only be performed by entities in the *Application Component* or *Auditor* role.

In detail, the TOE supports the folling functionalities:

Message Authentication Codes

The symmetric Message Authentication Code (MAC) algorithms supported are

- HMAC, and
- CMAC.

Both are requiring a key of at least 128 bits in size. Hereby, HMAC is used exclusively in combination with the Hash Function SHA-256 yielding a tag length of 32 bytes. The output length of CMAC, which uses the block cipher AES, is always 16 bytes, except for PACE where a tag length of 8 byte is required.

Digital Signatures

As mentioned above, the TOE supports the creation and verification of Digital Signatures via asymmetric cryptographic keys, whereby ECDSA and RSA are supported. The usable key sizes are fixed for both algorithms: A 256 bit elliptic curve key (SECP256r1) for ECDSA and a 4096 bit modulus for RSA, so following the recommendastions in **[BSI-TR-02102-2]**.

RSA signatures always use PSS Padding (PKCS#1 v2.1). The nonce required to create the ECDSA signature is created using the internal random number generator (**FCS_RNG.1**).

In summary, this SF is related to the following SFRs: **FCS_COP.1/MAC**, **FCS_COP.1/HMAC**, **FCS_COP.1/CDS-ECDSA**, **FCS_COP.1/VDS-ECDSA**, **FCS_COP.1/CDS-RSA**, **FCS_COP.1/VDS-RSA**, and **FDP_DAU.2/Sig**.

7.5. SF5: Authentication & attestation of the TOE, trusted channel

The TOE supports a trusted channel establishment. Hereby, a trusted channel includes authenticity and integrity of the data transmitted or received at all times. The establishment follows the following approaches:

- PACE,
- Terminal Authentication 2, and
- Chip Authentication 2.

Hereby, PACE is established with Curve SECP256r1 with cryptographic key sizes of 128 bits or 256 bits. Trusted channel establishment is supported by the random number generator and key agreement provided by **SF1: Key Management**.

Requests to an external interface of the TOE that require integrity and confidentiality protection undergo a MAC verification first. Only if this verification is successful, the data is decrypted and the requested operation is carried out. The corresponding responses of the TOE are encrypted and integrity protected, respectively.

Additionally, the TOE supports attestation via an external interface using ECDSA to find out whether the TOE sample is a genuine sample of the certified product. This functionality is based on the creation of digital signatures using ECDSA in accordance with **FCS_COP.1/CDS-ECDSA** as provided by **SF4: Data integrity mechanisms**.

In summary, this SF is related to the following SFRs: **FIA_API.1/PACE**, **FIA_API.1/CA**, **FDP_DAU.2/Att**, **FTP_ITC.1**, **FCS_CKM.1/PACE**, **FCS_CKM.1/TCAP**, **FCS_COP.1/TCE**, and **FCS_COP.1/TCM**.

7.6. SF6: User identification & authentication

The *User Administrator* may create, delete, modify authentication reference data and pin/unpin permanent session key for users and entities. These are stored in a persistent storage provided by the hardware platform that the TOE is operated on.

The initial password must be changed to a different secure operational password after the first successful password authentication. The corresponding policy enforces a password length of 32 byte, incentivising the user to transmit SHA-256 hashes of his pw.

In addition, the TOE supports blocking of users authentication tries for 2^(number of failed tries) seconds starting with 5 unsuccessful tries (i.e. the user has to wait for 32=2^5 seconds after the fifth consecutive failed attempt to log in). This applies to all logins. The counter of unsuccesfull logins of a user is reset to 0 after each successfull authentication of that user.

Furthermore. the TOE manages the binding of roles to a user/entity in an active session context. Initially the role is set to *Unidentified User*. This user an only execute the self test or start an authentication to become another role. Once the identity is provided to the TOE and the corresponding ID in the set of persistently stored *entity IDs*, the user/entity is associated with the role *Unauthenticated User*. This user an only execute the self test or start an authentication to become another role. Once the correct credentials are provided to the TOE and the role in the claimed role is also associated with the user/entity in the persistent storage of the TOE, the role of the user are changed to to the claimed role.

The TOE supports time-limited authorization by terminating active sessions after the 24 hours are exceeded.

The TOE enforces that only the TSF self test, the identification of the TOE to the user may happen before user identification.

The TOE supports multiple authentication mechanisms as follows: * The TOE supports password authentication for users via a local endpoint and changing the initial password to a secure operation password after the first successful authentication. * The TOE supports establishment of PACE as an authentication mechanism for both users and entities. * The TOE supports pinned PACE-Keys as an authentication mechanism. * The TOE supports Terminal Authentication 2, and Chip Authentication 2 as an authentication mechanism.

The TOE holds active sessions in volatile memory. Hence, any power on or reset terminates active sessions. The TOE does not support changing the role of an active session. Selecting the roles of a session is only possible during session establishment. All messages received via a trusted channel are authenticated by successful MAC verification.

In summary, this SF is related to the following SFRs: **FIA_ATD.1**, **FMT_MTD.1/RAD**, **FMT_MTD.3**, **FIA_AFL.1**, **FIA_USB.1** , **FMT_SAE.1**, **FIA_UID.1**, **FIA_UAU.1**, **FIA_UAU.5**, and **FIA_UAU.6**.

7.7. SF7: Access control

The TOE validates external user data inputs associated with cryptographic operations. In case of failed validation, the TOE will notify the requesting entity of the existing issue(s).

The TOE checks whether the requesting entity is authorized to perform any kind of data exports. For example, only an *Application Component* is allowed to timestamp data. The TOE enforces this by checking whether the required role is active in the current active session. Access is denied if the user or entity is unauthorized. Unauthenticated users are only allowed to start the self test of the TOE and to use the authentication function.

After successful authentication, a user assumes one of the following roles:

- Application Component,
- Crypto-Officer,
- User Administrator,
- Update Agent,
- Auditor,
- Timekeeper.

Additionally a user is also the Key Owner of all keys assigned to it.

If a user or entity wants to claim a role that is not in the set of assigned roles the authentication fails.

The TOE follows several rules concerning the format of data export which it follows, e.g. data exported accoring to **FCS_COP.1/HEM** always contains the id of the decryption key.

In summary, this SF is related to the following SFRs: **FDP_ITC.2/UD**, **FDP_ITC.2/TS**, **FDP_ETC.2**, **FDP_ETC.2/TS**, **FDP_ETC.1**, **FDP_ACC.1/Oper**, **FDP_ACF.1/Oper**, and **FDP_ACF.1/TS**.

7.8. SF8: Security Management

The TOE supports secure management of security functions, roles, attributes, and functions behavior as follows: The TOE maintains the roles as defined in **SF7: Access control**. The roles *Unidentified User*, *Unauthenticated User* are used during the authentication process only and kept in volatile memory. The role *Key Owner* is implicitly given to a user in respects to all keys he owns.

Additionally, the TOE ensures that no unsecure value is accepted for any TSF-related security attribute by enforcing mandatory inputs and performing input validation. The TOE implements this restrictive

policy be enforcing that the values for attributes are provided with each request. With other words: the TOE does never assume a default value for any attribute.

As mentioned in **SF6: User identification & authentication**, the TOE ensures that only authenticated users with the necessary role are allowed to execute corresponding functions and prevents unauthorized access by keeping a context for active sessions and comparing the required role of a function against the actual role of the authenticated user.

In summary, this SF is related to the following SFRs: **FMT_SMF.1**, **FMT_SMR.1**, **FMT_MSA.2**, **FMT_MOF.1**, **FMT_MTD.1/RAD**, **FMT_MSA.1/KM**, and **FMT_MSA.3/KM**.

7.9. SF9: Protection of the TSF

As required, the TOE supports self-testing of the TSF (cf. **FPT_TST.1**) to demonstrate their correct operation and preserves its secure state in case of failure. The relevant functions of the secure hardware platform are part of the scope of the self test.

A successful pass of the included selt-test routine is essential, before entering the operational mode of the TOE. Hence, it is executed as first step of the TOE before activating critical functions and/or interfaces. This includes at least:

- Check the availability of the timesource
- Check the configuration of the TOE
- Check the functionality of cryptographic subroutines
- Check the functionality of key access controll subroutines
- Check the availability of entropy

For all cases of failed self-test routines, the TOE switches to (stays in respectively) its secure state in order to protect the user data in case. The TOE can only leave this secure state after the self test has been succesfully executed again. In summary, this SF is related to the following SFRs: **FPT_FLS.1**, and **FPT_TST.1**.

7.10. SF10: Secure Update

The TOE provides a functionality for a secure update of the TOE. Therefore, the UpdateCodePackage (UCP) functionality is included in the TOE and is provided to the Administrator only. The Administrator imports a signed binary package consisting of the following elements:#package name, version number, signature, signature key reference, encryption key reference, encryption iv, encrypted software binary.

Before applying the UCP, the TOE verifies the signature value as signature over the UCP. The signature key used for signature verification is stated by the signature key reference, the key used for decryption by the TOE is stated by the encryption key reference. If the verification of the UCP fails, the import is denied.

If successful, the TOE verifies the security attribute issuer and version. The TOE will only proceed, if the provided update has a higher or equal version number compared to the currently installed TOE version. If this is not the case, the installation is denied.

By means of the decryption key the TOE decrypts the encrypted UCP content and change the environment to start the new binary once the current has terminated. After this, a restart of the TOE application software is performed automatically.

If at any point the import is denied, the import procedure is aborted and all imported data and data derived from that are destroyed by means of zeroization before the related resource is released.

In summary, this SF is related to the following SFRs: **FDP_ACC.1/UCP**, **FDP_ACF.1/UCP**, **FDP_ITC.2/UCP**, **FCS_COP.1/VDSUCP**, **FDP_RIP.1/UCP**, **FCS_CKM.4**, **FPT_TDC.1/UCP**, **FCS_COP.1/DecUCP**.

7.11. SF11: Time Stamp

The TOE synchronizes the used internal local time periodically with a trusted time service via an authenticated Network Time Protocol (NTP) service provider. Hereby, only those servers are used, which support sufficient cryptographic authentication. The TOE manages a trust anchor to identify and authenticate trusted NTP servers. This synchronization is performed via clock skewing, as long as the difference between the local and remote time is no larger than 128ms.

In addition, users authenticated in the role *Timekeeper* may manually force a time synchronization of the TOE by calling the corresponding function. Afterwards the automatic time synchronization process which relies on a trusted time source will continue to operate as defined above.

The TOE suppports reliable timestamping of audit data or any other data exported from the TOE, which makes it possible to cryptographically guarantee, that the piece of data was created before the signature time of the timestamp and prevents the backdating. This service is also used for the generation of signed log messages in the context of **[BSI-TR-03151]**.

In summary, this SF is related to the following SFRs: **FDP_DAU.2/TS**, **FDP_ITC.2/TS**, **FDP_ETC.2/TS**, **FDP_ACF.1/TS**, **FMT_SMF.1/TSA**, **FMT_SMR.1/TSA**, and **FMT_MOF.1/TSA**.

7.12. SF12: Security Audit

The TOE generates audit records of auditable events as follows:

- Start-up of the TOE
- Start-up and shutdown of the audit functions
- Discrete adjustment of the used RTC
- Import of update packages
- Authentication failure handling once a user reached the limit of failed attempts
- Generation of signature key pairs
- Cryptographic key destruction of permanently stored keys
- Generation of cluster keys for the secure channel
- Export of Authentication Data Records and cryptographic keys from the master node,
- Management of Authentication Data Records, creation and deletion of Authentication Data Record
- Import of Authentication Data Records and cryptographic keys into slave nodes

The TOE records the date and time, type, identity of the affected user or entity (if applicable), the result of the event (success or failure), and more details regarding the event (if applicable).

The audit trail can be exported and are always digitally signed and accompanied by a timestamp. Audit log messages are encoded using DER (Distinguished encoding rules) according to the ASN.1 specification in **[BSI-TR-03151]**.

Audit logs always are exported and persisted immediately to the local storage of the environment of the TOE.

In summary, this SF is related to the following SFRs: **FAU_GEN.1**, **FAU_GEN.1/CL**, **FMT_MTD.1/Audit**, **FAU_STG.1**, **FAU_STG.3**, **FPT_STM.1**, and **FPT_TIT.1/Audit**.

7.13. SF13: Clustering

The TOE supports clustering of TOE instances with a master node and 1 slave node. The master node (configured by the *Crypto Officer* only) communicates updates concerning its cryptographic keys and authentication data records to the slave of the same cluster configured via a secure channel. Hereby, the TOE supports the generation of audit logs for data imports and exports, the generation of a log entry once a TOE instance becomes master of the cluster and authentication data records. As mandatory

security feature, only the master node has write privileges, all other ones are in read-only mode. Changes to the stored information (which is relevant for clustering) of the master node will be propagated to the other node in the cluster periodically via a secured connection. This is specifically the case for keys (which have the attribute *clusterable* amongst their key access control attributes), user information including authenticiation records, audit logs and system logs.

All keys shared within a cluster by the master node will be transferred with all related attributes.

In summary, this SF is related to the following SFRs: **FDP_ACC.1/CL**, **FMT_MTD.1/CL**, **FCS_CKM.5/CLDH**, **FPT_TCT.1/CL**, **FPT_TIT.1/CL**, **FPT_ISA.1/CL**, **FPT_ESA.1/CL**, and **FPT_TDC.1/CL**.

Overview of all keys in the Swissbit Cloud CSP-L

The following table provides a generic overview over all keys of the Swissbit Cloud CSP-L sorted after the functions of the Swissbit Cloud CSP-L.

Table 10. Overview of keys

Function	**keys**	**clusterable ?**
ChangeAuditTrailSignatureKey	Ec_priv_digSigTime_perm	yes
DecryptData (primary key)	Rsa_priv_enc_perm	yes
DecryptData (primary key)	Ec_priv_enc_perm	yes
EncryptData (primary key)	Ec_pub_enc_eph	no
DecryptData (primary key)	Aes_sec_enc_perm	yes
DecryptData (derived key)	Aes_sec_enc_eph	no
DecryptData (derived key)	Aes_sec_mac_eph	no
EncryptData (primary key)	Rsa_pub_enc_perm	yes
EncryptData (primary key)	Ec_pub_enc_perm	yes
EncryptData (primary key)	Ec_priv_enc_eph	no
EncryptData (primary key)	Aes_sec_enc_perm	yes
EncryptData (derived key)	Aes_sec_enc_eph	no
EncryptData (derived key)	Aes_sec_mac_eph	no
DeriveKeyFromKeyAgreement	Ec_pub_agree_perm	yes
DeriveKeyFromKeyAgreement	Ec_priv_agree_perm	yes
ExportKey (primary key)	Aes_sec_wrap_perm	yes
ExportKey (derived key)	Aes_sec_enc_eph	no
ExportKey (derived key)	Aes_sec_mac_eph	no
ImportKey (primary key)	Aes_sec_wrap_perm	yes
ImportKey (derived key)	Aes_sec_enc_eph	no
ImportKey (derived key)	Aes_sec_mac_eph	no
SignData	Ec_priv_digSig_perm	yes
SignData	Ec_priv_digSigTime_perm	yes

SignData	Rsa_priv_digSig_perm	yes
TimestampData	Ec_priv_digSigTime_perm	yes
VerifyData	Ec_pub_digSig_perm	yes
VerifyData	Ec_pub_digSigTime_perm	yes
VerifyData	RSA_pub_digSig_perm	yes
SetMaster	Aes_sec_cluster_perm	yes (but unsuse)
SetSlave	Aes_sec_cluster_perm	yes (but unused)
+Clustering (derived key)	Aes_sec_enc_eph	no
+Clustering (derived key)	Aes_sec_mac_eph	no
+SignAuditLogs CSPL	Ec_priv_digSigTime_perm	yes
+SignAuditLogs User	Ec_priv_digSigTime_perm	yes
+SignUpdateTimeLogs User	Ec_priv_digSigTime_perm	yes
ImportUCP (signature verification)	Ec_pub_digSig_perm	yes
ImportUCP (signature verification)	Rsa_pub_digSig_perm	yes
ImportUCP (decryption)	Aes_sec_enc_perm	yes
+RootKey	Ec_pub_root_perm	yes
+RootKey	Rsa_pub_root_perm	yes
SignData (ContentCommitment)	Ec_priv_commit_perm	yes
TAStep1 (simplified Identification)	Ec_pub_digSig_perm	yes
TAStep1 (simplified Identification)	Rsa_pub_digSig_perm	yes
Pinned Pace key	Aes_sec_agree_perm	yes

Related Documents

Note that version numbers of the following documents are partially missing. A separate, central list of references including document versions is created and provided to prevent version mismatches in different documents.

- **[ANSI-X9.63]** ANSI-X9.63, Key Agreement and Key Transport Using Elliptic Curve Cryptography, 2011
- **[PP-CSPL-V1.0]** Protection Profile Cryptographic Service Provider Light (CSPL), BSI-CC-PP-0111-2019, BSI, version 1.0
- **[PP-CSPL-TS-AU-V1.0]** Protection Profile Cryptographic Service Provider Light - Time Stamp Service and Audit (PPC-CSPLight-TS-Au), BSI-CC-PP-0112-2020, version 1.0
- **[PP-CSPL-Cluster-V1.0]** Protection Profile Cryptographic Service Provider Light Time Stamp Service and Audit - Clustering (PPC-CSPLight-TS-Au-Cl), BSI-CC-PP-0113-2020, version 1.0
- **[BSI-TR-02102-2]** Technische Richtlinie TR-02102-2 Kryptographische Verfahren: Empfehlungen und Schlüssellängen, Teil 2 – Verwendung von Transport Layer Security (TLS), TR-02102-2, Version 2019-01
- **[BSI-TR-03110]** BSI, Technical Guideline TR-03110 Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token – Part 2 - Protocols for electronic IDentification, Authentication and trust Services (eIDAS), Version 2.21, 2016
- **[BSI-TR-03111]** Technische Richtlinie BSI TR-03111 Elliptische-Kurven-Kryptographie (ECC), TR-03111, Version 2.10
- **[BSI-TR-03151]** Technical Guideline BSI TR-03151 Secure Element API (SE API), TR-03151, Version 1.1.1
- **[CC2]** Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, CCMB-2017-04-002, Version 3.1 Revision 5, April 2017.
- **[CC3]** Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, CCMB-2017-04-003, Version 3.1 Revision 5, April 2017.
- **[FIDO-ECDAA]** FIDO Alliance, Alliance Proposed Standard FIDO ECDAA Algorithm, **https://fidoalliance.org/specs/fido-u2f-v1.2-ps-20170411/fido-ecdaa-algorithm-v1.2-ps20170411.html**, 11 April 2017
- **[FIPS140-2]** Security Requirements for Cryptographic Modules, FIPS 140-2, May 2001
- **[FIPS_180-4]** Secure Hash Standard (SHS), FIPS 180-4, October 2015
- **[FIPS_186-4]** Digital Signature Standard (DSS), FIPS 186-4, July 2013

- **[FIPS_197]** ADVANCED ENCRYPTION STANDARD (AES), FIPS 197, November 2001
- **[ICAO-Doc9303]** Machine Readable Travel Documents , ICAO, Doc 9303,Part 11: Security Mechanisms for MRTDSs, Seventh Edition, 2015
- **[ISO-18033-3]** ISO/IEC 18033-3 Information technology - Security techniques, Encryption algorithms - Part 3: Block ciphers, 2010
- **[ISO-IEC_14888-2]** ISO/IEC 14888-2 Information technology – Security techniques, Digital signatures with appendix – Part 2: Integer factorization based mechanisms, , 2008
- **[ISO-IEC_10116]** ISO/IEC 10116 Information Technology - Security techniques, Modes of operation for an n-bit block cipher, 2017
- **[ISO-IEC_9797-2]** SO/IEC 9797-2 Information Technology - Security techniques, Message Authentication Codes (MACs), Part 2: Mechanisms using a dedicated hash-function, 2011
- [NIST-SP800-38C] NIST, Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality, May 2004
- **[NIST-SP800-38F]** NIST , SP800-38F Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping, 2012
- **[NIST-SP800-38A]** NIST, SP800-38A Recommendation for Block Cipher Modes of Operation: Methods and Techniques
- **[NIST-SP800-38B]** NIST, SP800-38B Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, May 2005
- [NIST-SP800-38D] NIST, SP800-38D Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007
- **[NIST-SP800-56C]** NIST, Recommendation for Key Derivation through Extraction-then-Expansion, Special Publication SP800-56C, November 2011
- **[SP800]** NIST SP 800-90A Rev. 1 Recommendation for Random Number Generation Using Deterministic Random Bit Generators
- **[PKCS-1]** PKCS #1 v2.2: RSA Cryptographic Standard, **https://www.emc.com/emc-plus/rsa-labs/pkcs/files/h11300-wp-pkcs-1v2-2-rsacryptography-standard.pdf**, 27.10.2012
- **[PP-SMAERS]** Common Criteria Protection, Profile Security Module Application for Electronic Record-keeping Systems (SMAERS), BSI-CC-PP-0105-V2-2020, version 1.0
- **[FCG]** Fiscal Code of Germany in the version promulgated on 1 October 2002 (Federal Law Gazette [Bundesgesetzblatt] I p. 3866; 2003 I p. 61), last amended by Article 6 of the Law of 18. July 2017 (Federal Law Gazette I p. 2745)
- **[RFC-2104]** RFC2104, HMAC: Keyed-Hashing for Message Authentication

- **[RFC-5639]** Elliptic Curve Cryptography (ECC) Brainpool Standard, Curves and Curve Generation, March 2010
- **[RFC-5903]** RFC5903, Elliptic Curve Groups modulo a Prime (ECP Groups) for IKE and IKEv2
- **[RFC-6954]** RFC6954, Using the Elliptic Curve Cryptography (ECC) Brainpool Curves for the Internet Key Exchange Protocol Version 2 (IKEv2)
- **[RNG_classes]** A proposal for: Functionality classes for random number generators1 Version 2.0 18 September 2011
- **[TPMLib-P1]** Trusted Platform Module Library, Part 1: Architecture, Family "2.0", Level 00, Revision 01.38, September 29, 2016
- **[CSPL-AGD]** Swissbit Cloud CSP-L - Guidance Documentation, Version 1.1.5