Security Target

fiskaly Security Module Application for Electronic Record-keeping Systems

TOE Version 1.0.6

Document Version 1.2.0

2021-08-31

fiskaly GmbH

Contents

Document and TOE Reference

Document Type:	Security Target
Document Version:	1.2.0
Document built from commit:	59c73729c3e8f14438556b8e687395965f5f1764
Date:	2021-08-31
Author:	fiskaly GmbH
Certification-ID:	BSI-DSZ-CC-1130-V2
TOE Identification:	fiskaly Security Module Application for Electronic Record-keeping Systems
TOE Version:	1.0.6
CC Version:	3.1 Revision 5
Assurance Level:	EAL2 augmented with ALC_LCD.1 and ALC_CMS.3
PP Conformance:	BSI-CC-PP-0105-V2-2020 (version 1.0)

Chapter 1

Introduction

This document is the Security Target for the Common Criteria evaluation of the fiskaly Security Module Application for Electronic Record-keeping Systems.

In order to combat tax fraud, electronic record-keeping systems in Germany must be equipped with a 'Certified Technical Security System' (CTSS; 'Zertifizierte Technische Sicherheitseinrichtung') that consists of a storage medium, a security module, and a unified digital interface. The security module is subject to common criteria security certifications. W.r.t. to security requirements for the security module – defined by Bundesamt für Sicherheit in der Informationstechnik – the module consists of two components:

(1) an application component that handles the business logic and functionality required to serve an electronic record-keeping system. This component is dubbed the *Security Module Application for Electronic Record-keeping Systems* (SMAERS).

(2) a generic and reusable cryptographic component that implements the core cryptographic functionality required. This component is dubbed *Cryptographic Service Provider* (CSP).

The PP SMAERS [15] defines the security requirements of the SMAERS component. Depending on the overall architecture, different security requirements exist for a CSP. These are defined in two protection profiles (CSP [9] and CSPLight [13]) and additional protection profile configurations regarding auditing/timestamping and clustering. The TOE uses a CSPLight compliant with [14] (Protection Profile Cryptographic Service Provider Light Time Stamp Service and Audit - Clustering). In the following, the abbreviation CSP is redundantly used for all allowed configurations mentioned.

1.1 TOE Overview

The *fiskaly Security Module Application for Electronic Record-keeping Systems* is a software that enables compliance with the German KassenSichV [2] as part of a *Certified Technical Security System (CTSS)* [8]. Its main purpose is the transformation of incoming *transaction data* to signed *log messages*. For that reason, the *fiskaly Security Module Application for Electronic*

Record-keeping Systems makes use of cryptographic operations and security services provided by a *Cryptographic Service Provider Light (CSPLight)* [13].

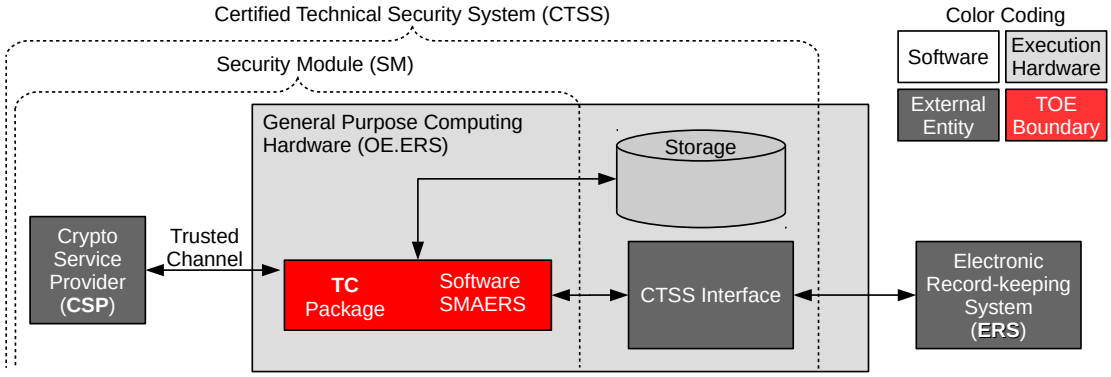


Figure 1.1: Overview of the CTSS components

Figure 1.1 shows an overview of the *Certified Technical Security System (CTSS)*. The *Software SMAERS* connects to the *CSP* via a *Trusted Channel* provided by the claimed *TC Package*. The TOE is executed on *General Purpose Computing Hardware* in the operational environment (*OE*) of an Electronic Record-keeping System (*ERS*). The TOE uses a storage for storing signed *log messages* and provides an interface to the ERS to record *transaction data*. In addition, Figure 1.1 clearly indicates the *TOE Boundary*, the components of the *Security Module (SM)*, and the components of the *Certified Technical Security System (CTSS)*. Note that the CTSS does not rely on an external *CTSS Interface component* between the security module and the *ERS*. Therefore in the following sections of this document, each time the TOE is mentioned as interacting with the *CTSS Interface*, it directly interacts with the *ERS*.

1.1.1 Usage and Major Security Features

The TOE provides the following TOE security functions:

SF_SecMan Security Management

Security management is concerned with

- maintaining roles and what actions these roles are allowed to perform, and
- managing the internal behavior of the TOE in response to external stimuli.

SF_IdAuth User Identification and Authentication

The TOE supports identification and authentication of external entities.

SF_UserDataProt User Data Protection

This security function focuses on the quality of imported and exported user data. For example, if certain properties in transaction data are missing or unexpected (e.g. an unknown transaction is updated or finished), the TOE rejects the import of transaction data.

SF_TSFProt Protection of the TSF

The TOE provides capabilities for testing external components and its correct functioning.

SF_Audit Security Audit

The TOE creates audit records of auditable events and exports them as system logs.

SF_UcpImp Update Code Package Import

Software updates are supported by the SMAERS platform and the TOE prevents version downgrades.

SF_TrustChan Trusted channel between TOE and CSPLight

The TOE communicates securely with a remote Cryptographic Service Provider Light (CSPLight) with regards to authenticity, integrity and confidentiality.

1.1.2 TOE Type

The Target of Evaluation (TOE) is a *security module application* implemented as software. It is either running on the platform of the CSP (referred to as *platform-architecture*), or running on a separate device communicating with the CSP via a trusted channel (referred to as client-server architecture), cf. [15] [13].

The TOE has to securely store sensitive objects (user data and TSF data, see assets). In case of the platform-architecture, the CSP platform provides suitable mechanisms for this that may be used by the TOE.

In case of the client-server architecture, where the TOE can not directly rely on the CSP platform, a platform with secure storage must be used. The platform that executes the TOE has to provide mechanisms to preserve the integrity, confidentiality (when required), and to prevent rollback of stored sensitive objects, including the TOE software itself. The confirmation of suitability of the chosen platform shall be part of the evaluation.

The TOE relies on the CSP for all cryptographic operations except for the implementation for the trusted channel. In addition the TOE must rely on the platform in case of update code package verification.

***ST application note 1*: This TOE uses client-server architecture.**

1.1.3 Required non-TOE hardware/software/firmware

In the following, the required non-TOE hardware/software/firmware is described on which the security of the TOE relies.

Cryptographic Service Provider Light (CSPLight)

The TOE requires a Cryptographic Service Provider Light (CSPLight or CSPL) with CC EAL2 certification according to *Protection Profile Cryptographic Service Provider Light Time Stamp Service and Audit - Clustering (PPC-CSPLight-TS-Au-Cl) Version 1.0, BSI-CC-PP-0113-2020* [14]. The CSPL used in the TOE is *fiskaly Cloud Crypto Service Provider* [21]. Note: Throughout this document we use the terms CSP (Cryptographic Service Provider [9]) and CSPL[ight] (Cryptographic Service Provider Light [13]) interchangeably, in accordance with the terminology employed in the PP SMAERS [15]. The CSPLight is an external entity (following the client-server model described in PP SMAERS [15]).

The CSPLight must be securely operated in an environment certified according to ISO/IEC 27001. The operator must implement and continuously maintain an information security management system (ISMS) with security level *high*. The hardware platform of the operational environment of the CSPLight satisfies all requirements specified in Appendix of [15] (Operational Requirements for CSPLight).

The interfaces of the CSPLight are defined by the AGD [20] and FSP [18] documentation of the *fiskaly Cloud Crypto Service Provider*.

As the *fiskaly Cloud Crypto Service Provider* is cloud-based, the TOE uses the services of the CSPLight remotely. In the usual case, customers will have a contract with fiskaly GmbH regarding the whole CTSS, which includes (aside from TR-certified components) both the (CC-certified) TOE and the non-TOE CSPLight. That is, in a virtual sense, the required CSPLight services are "delivered" in combination with the TOE, while the CSPLight itself is a remote entity to the TOE.

Execution Device

The TOE is run on general purpose computing hardware, i.e. laptop, personal computer or server hardware.

Operating System

The TOE requires an Operating System, i.e. a system software providing access to the *Execution Device*'s hardware components, in particular storage and network devices.

Table 1.1 shows the supported platform of the TOE.

Platform	Architectures
Alpine Linux Version 3.x	running in Docker 19 on maintained underlying operating system

Table 1.1: Supported platform

Mass Storage Device

The TOE requires a Mass Storage Device for persisting signed transaction logs.

Secure Mass Storage

The TOE requires a secure and trusted Mass Storage Device for persisting internally stored data, including the PACE password, the transaction counter and other configuration settings.

1.1.4 Physical Scope of the TOE

The TOE is a software TOE according to PP SMAERS [15].

This ST is part of the TOE and publicly available. The "Preparative Procedures & Operational User Guidance Documentation – fiskaly Security Module Application for Electronic Record-Keeping Systems" document (cf. [19]) is part of the TOE. This document is to be made available to parties purchasing the TOE, or which have signed a corresponding NDA with fiskaly GmbH. In addition, the Document "Functional Specification – Security Module Application for Electronic Record-Keeping Systems" (cf. [17]) is made available to integrators of the CTSS / TOE, and to parties which have signed a corresponding NDA with fiskaly GmbH. All (other) evaluation evidence is only made available to parties involved in the certification process.

1.1.5 Certificates to verify log messages of the TOE

The TOE creates signed log messages. To verify these, the TOE exports a certificate chain with each TAR file. The certificates are issued by a PKI. The root CA certificate can be obtained by the BMF. The version, being obtained from the Federal Ministry of Finance (BMF) is the root of trust for the verification.

To check the version of the TOE, export a TAR-file from the TOE and consider the file `info.csv` within. The field version in this file contains the version of the TOE and of the CSPL, for example the entry `fiskaly sign cloud-TSE v1.0.5-1.2.0` refers to SMAERS version 1.0.5 and CSPL version 1.2.0.

1.2 TOE Life Cycle

The TOE life cycle is part of the life cycle of the CTSS. This section describes the complete life cycle of the CTSS, including details necessary for understanding the interaction with the CSPLight. In addition, the life cycle of the CTSS is closely coupled to the lifetime of the certificate, which is used to verify the signatures being provided by the CSPLight. Note that the key pair and certificate are constant for the life cycle of the TOE and CTSS. They cannot protect additional transaction data, once the certificate is no longer valid.

Before an ERS can use the TOE, the integrator or ERS manufacturer has to integrate the CTSS into its ERS System. Then the CTSS / TOE has to be prepared on request of a customer, such that a key pair (and account) is created in the CSPLight and a certificate is created in the associated CA.

Then, the CTSS / TOE can be instantiated by providing configuration data to access the CSP-Light.

Now the TOE life cycle begins and the TOE is operational.

The life cycle of the TOE ends once it becomes disabled via the CTSS interface. This state is irreversible. Disabling the TOE via the CTSS interface can be seen as the standard way of ending the life cycle of the TOE by the customer. This way the proper system log that documents the decommissioning of the CTSS gets generated.

1.3 TOE Description

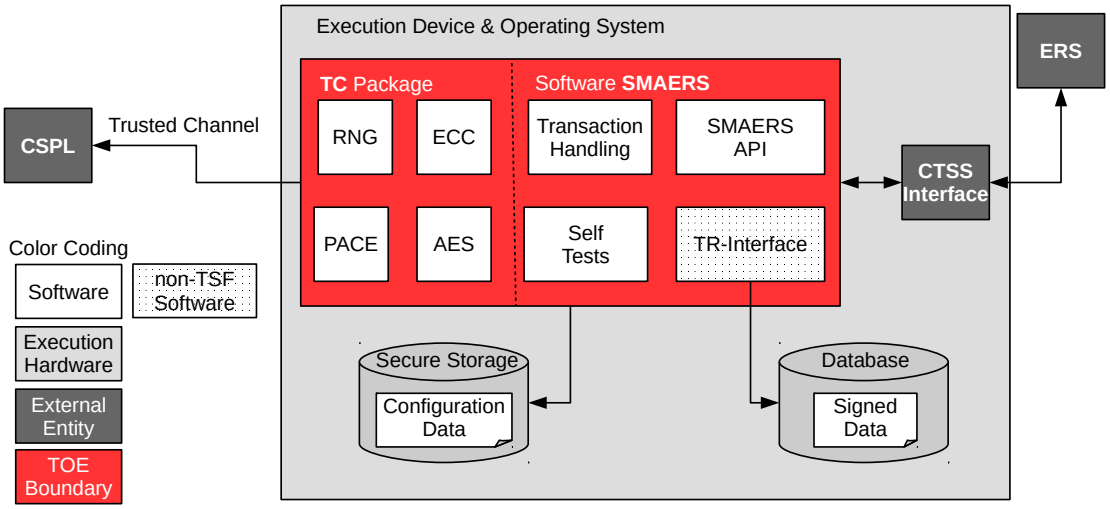


Figure 1.2: Description and interaction between the TOE and the relevant non-TOE components

The TOE is a Security Module Application for Electronic Record-keeping Systems (SMAERS) [15] implemented as software as part of the *Security Module (SM)* [16] of a *Certified Technical Security System (CTSS)* [8] for *Electronic Record-keeping Systems (ERS)*. The TOE is provided as

an application that is to be executed in the operational environment of the ERS (*OE.ERS*).

Figure 1.2 shows the interactions between TOE and relevant non-TOE components. The *Software SMAERS* connects to the *CSPLight* via a *Trusted Channel* provided by the claimed *TC Package*. The *CSPLight* provides cryptographic services (most importantly signature creation with timestamp and signature counter). The *TR-interface* contains non-TSF functionality, which is required by [16] but is not required by [15]. For example filtered exports of log messages are realized here. The TOE is executed on an *Operating System* running on an *Execution Device* in the operational environment of an ERS (Electronic Record-keeping System). The TOE stores its integrity-protected data in a *Database* and verifies the correct functioning of its *Transaction Handling* by *Self Tests*. *Signed Data* are the log information, which are integrity protected by the CSP's digital signatures. In addition, the TOE is dependent on a *Secure Storage*, which allows the TOE to store its *Configuration Data* (e.g. PACE Password, transaction counter, ...) in a confidentiality and integrity protected way. This storage has to be provided by the Operational Environment of the TOE.

The KassenSichV [2] requires the Security Module to provide tamper-proof determination of the point in time when the transaction starts, the transaction number, the point in time when the transaction is completed or terminated, and the check value (i.e. signature value). The Security Module provides the logging of accounts, records and security management activities in form of log messages. Log messages are created by the TOE using the CSPLight over a trusted channel (TC).

The TC is necessary because the TOE and the CSPLight are implemented as separated components. Their interaction through a trusted channel is required in order to protect the integrity of the communication data, and to prevent misuse of the CSPLight signing and time stamping service provided for the TOE. The TC is based on PACE (Password Authenticated Connection Establishment) [22] for key agreement and CMAC AES-256 is used to protect the integrity and authenticity of exchanged messages.

Log messages consist of certified data, protocol data, audit data and a signature. There are three types of *log messages*, namely *transaction logs*, *system logs* and *audit logs* (cf. TR-03151 [16]).

Transaction logs are created to protect the actual transaction data of the electronic record-keeping system as certified data. They are generated whenever a transaction is started, updated or finished. The protocol data of *transaction logs* contain the transaction number of the actual transaction and time stamps. All transaction logs with the same transaction number build together the required data of the fiscal transaction according to [2] Section 2, Sentence 2.

System logs are generated to log the execution of system operations and TSF security events. The certified data of the systems logs provide information about system events.

Audit logs are generated to document management or configuration operations of the CSPLight. The audit data of audit logs provide information for the interpretation of the *transaction logs*, e.g. providing information about setting or readjusting the time source that is used for time stamps.

The TOE

- imports transaction data from the ERS and includes it as certified data in a transaction log,

- increments the internally stored transaction number in case of StartTransaction operations,
- generates part of the protocol data of the transaction log, including the transaction number generated by the TSF, and the serial number as hash value of the public key included by the TSF for verification of the digital signature (keyID),
- includes the digital signature created by the CSPLight over the certified data and the protocol data in the transaction log,
- communicates with the CSPLight for consuming security and cryptographic services via a trusted channel,
- adds the signature created by the CSPLight to log messages together with the corresponding timestamp and signature counter value provided by the CSPLight,
- exports signed transaction logs to the storage,
- imports audit records from the CSPLight and exports them as signed audit logs to the storage,
- generates signed system logs and exports them to the storage,
- provides identification and authentication, access control and security management of the TSF,
- provides self-test and external entity test capabilities to verify its correct functioning, and
- meets the BSI Technical Guidance TR-03153 [8] and uses cryptographic services of the CSPLight compliant with BSI TR-03116-5 [4].

The TOE prevents version downgrades by comparing the version number of the Update Code Package (UCP) with the current version number of the TOE during self-test at start-up. If the version number of the UCP is smaller than the version number of the TOE, the self-test fails.

UCP verification is performed by the platform. The supported platform is given in Table 1.1. For Linux, signed RPM and DEB packages are used.

The TOE maintains several roles of entities. The roles are *unidentified user*, *administrator*, *CTSS interface* and *CSP*. Based on the active role of an entity, the TOE offers a set of possible operations, and prevents certain actions that must be only available to other roles (cf. Chapter 3.1).

The security management of the TOE maintains roles as follows:

- the role *administrator* is assigned to users after successful password authentication,
- the role *CTSS interface* is assigned to entities providing a clientID that is in the set of accepted clientIDs,
- the role *TR administrator* is assigned to entities being *CTSS interface* and additionally authenticating with a password / PUK,
- the role *CSP* is assigned to CSPLight after successful trusted channel establishment.

The CTSS interface role may

- define the keyID to be used for signature creation based on the clientID.[1]
- import transaction data and export (signed) log messages

The TR administrator role may

- dynamically add or remove clientIDs (of electronic record-keeping systems)
- terminate the life cycle of the TOE
- clear the stored log messages after export

The TOE supports identification and authentication of administrators by using locally stored password-hashes.

An authenticated administrator may modify his administrator password.

The configuration options regarding testing of external entities do not exist.

Whenever a client reports a *StartTransaction* operation, the TOE increments the internal transaction counter by 1. In case of *UpdateTransaction* and *FinishTransaction* operations, the transaction number is imported with the transaction data provided by the client.

Whenever a code update is triggered, the TOE uses the CSP to protect the new version number against modification.

[1] Since the serial number of a TSS is equivalent to the hash of the public key used for signature creation of transaction logs, the change of a keyID would change the serial number of the TSS. Consequently, this operation only takes place once at the beginning of the life cycle of a TSS.

Chapter 2

Conformance Claims

2.1 CC Conformance Claims

This ST claims conformance to CC version 3.1 revision 5.

Conformance to CC Part 2 (security functional requirements) [6] is CC Part 2 extended.

Conformance to CC Part 3 (security assurance requirements) [6] is CC Part 3 conformant.

2.2 PP Claims

This ST claims strict conformance to PP Security Module Application for Electronic Record-keeping Systems BSI-CC-PP-0105-V2-2020 (version 1.0) [15].

2.3 Package Claims

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

2.4 Conformance Rationale

The dependencies of security assurance components of the package EAL2 are solved within the package [7]. The components ALC_LCD.1 and ALC_CMS.3 have no dependencies on other components.

Chapter 3

Security Problem Definition

3.1 Introduction

Assets

Asset	Protection
transaction data	authenticity, integrity
transaction number	authenticity, integrity
audit logs/audit records, system logs and transaction logs	authenticity, integrity
update code package	authenticity
UCP version number	integrity
PACE password	integrity, confidentiality

Table 3.1: Assets to be protected by the TOE

The assets of the TOE are

- the transaction data provided by the CTSS interface component, where authenticity and integrity, including completeness of the transaction data, shall be protected, i.e. verification of the transaction log messages shall determine whether the transaction data was received from the CTSS interface component, and modifications and gaps shall be detectable,
- the transaction number (as part of the transaction data) that enumerates transactions, where the transaction number must be continuously increasing without gaps,
- the audit records imported from the CSPLight and exported as audit logs to the CTSS interface component, the system logs and transaction logs,
- the update code package (UCP) and the UCP version number,

- the PACE password to set up the trusted channel to the CSPLight.

To perform mutual authentication using the PACE protocol, both endpoints need to share a static secret (PACE Password). The integrity and confidentiality of the shared secret are preserved by the TOE, using the secure storage of its platform.

The CSPLight protects and enumerates its audit records against undetected modification and gaps.

Users and Subjects

The TOE knows users as external units that actively communicate with the TOE as

- *electronic record-keeping system (ERS)*,
- *CTSS interface component*,
- *CSPLight*,
- *(SMAERS-) administrator*.

The *ERS* is tested by the TOE as an external entity and communicates with the TOE. The TOE uses the *CSPLight* as an external entity providing security services and audit records and is tested as such.

The *(SMAERS-)administrator* is assumed to be the TOE manufacturer or an integrator acting on behalf of the manufacturer and must not be the taxpayer.

The subjects as active entities in the TOE perform operations on objects and obtain their associated security attributes from the authenticated users on whose behalf they are acting, or by default.

Roles

The TOE knows the following roles taken by a user or a subject acting on behalf of a user:

- role *unidentified user*: This role is associated with any user not (successfully) identified by the TOE. This role is assumed for subjects after start-up of the TOE and deactivated *CTSS interface component*. The TOE allows users in this role to run self-tests of the TOE.
- role *administrator*: A user in this role is allowed to perform management functions. The administrator subject is acting on behalf of a human user, after successful authentication as administrator, until logout or restart of the TOE.
- role *CTSS interface*: A subject in this role is allowed to import *Transaction Data* from the *ERS*, to generate *transaction logs* and *system logs*, and to export *transaction logs* and *system logs* to the storage and *ERS*. A subject in this role is started automatically after start-up of the TOE if the *CTSS interface* role is activated and the *ERS* and the *CSPLight* are successfully tested according to FPT_TEE.1. The ERS uses the CTSS role.

- role *TR administrator*: A subject in this role is allowed to manage the list of acceptable client IDs for identification of ERS. In addition a subject in this role can terminate the life cycle of the TOE.
- *CSP role*: A subject in this role is allowed to import audit records from CSPLight and to export *audit logs* to the *CTSS interface component*. In addition the CSP role is allowed to start the update process. A subject in *CSP role* is started automatically after start-up of the TOE if the *CSPLight* is successfully tested according to FPT_TEE.1.

***ST application note 2*: The CSP role is utilized by the TOE to upgrade its internal data to a new TOE version. The software update itself is executed via the platform.**

Objects

The TSF operates the following types of user data objects

- *transaction data* (TD),
- *audit records*,
- *data-to-be-signed* (DTBS),
- *protocolData with signature* containing the time stamp, the signature counter and the digital signature; all generated by the CSPLight (cf. TR-03151 [16] and TR-03153 [8]),
- *log messages* (LM) as *transaction log*, *system log* or *audit log*,
- *update code package* (UCP),
- *commands (type of operation)*.

The formats of *transaction data* and *log messages* meet the BSI TR-03151 [16].

The *ERS* provides *transaction data* as data to be certified by means of *transaction logs* (cf. TR-03151 [16] and TR-03153 [8]).

Audit records are data imported from the CSPLight.

The *data-to-be-signed* compiled by the TSF and sent to the CSPLight for signing and time stamping consists of

- certified data i.e.
  - in case of a *transaction log*: the *transaction data* with type of the certified data *transaction log*, object identifier (id-SE-API-transaction-log): bsi-de (0.4.0.127.0.7) applications (3) sE-API (7) sE-API-dataformats(1) 1 (cf. [16], chapter 2.3.1);
  - in case of a *system log*: the security related events with the type of the certified data *system log*, object identifier (id-SE-API-system-log): bsi-de (0.4.0.127.0.7) applications (3) sE-API (7) sE-API-dataformats(1) 2;

  - in case of an *audit log*: the *audit record* with the type of the certified data *audit log*, object identifier (id-SE-API-audit-log): bsi-de (0.4.0.127.0.7) applications (3) sE-API (7) sE-API-dataformats(1) 3;

- protocol data generated by the TSF
  - the *transaction number*,
  - the *keyID* as a hash value of the signature-verification key,
  - the *type of the operation* as name of the API function whose execution is recorded by the *log message*, i.e. *StartTransaction*, *UpdateTransaction* or *FinishTransaction*,
  - the *optional protocol data* (may be empty).

The CSPLight adds to the *data-to-be-signed*

- the point in *time* when the *log message* was created,
- the *signature counter* that enumerates the signatures created with the signature-creation key.

Refer to TR-03153 [8] for details of the log messages format.

The Update Code Package (UCP) is a complete software package that is managed by the secure platform and its operating system that executes the SMAERS application. The operating system of the secure platform performs an update of the SMAERS application, it is required that the verification of the UCP is performed by the operating system prior to installation. Depending on the update procedure of the operating system either the new TOE alone or the old TOE and the new TOE together perform an upgrade by exporting and importing TSF data into the new TOE.

Security Attributes

Users known to the TOE have the security attributes stored in an *authentication data record (ADR)*.

- *user identity* (User-ID),
- *authentication reference data*,
- *role* with detailed access rights gained after successful authentication.

CTSS interface component and CSPLight known to the TOE have at least the security attributes *identity*, cf. FIA_ATD.1.

Passwords as *authentication reference data* have the security attributes

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

The *transaction data* (TD) have the security attributes

- *clientID* to determine the signature-creation key to be used for signing the *transaction log* and the *keyID* to be included in the protocol data of the *transaction log*,
- *type of the operation* to determine the actual transaction as *StartTransaction*, *UpdateTransaction* or *FinishTransaction*,
- *transaction number* to assign the TD to an ongoing transaction and enumerating the transactions, continuously increasing without gaps.

The TOE accepts *transaction data* only if the clientID is known and mapped to a signature key in the CSPLight (*keyID*).

The TOE manages for each known keyID the last assigned transaction number and the transaction numbers of the ongoing transactions. If the *type of the operation* of imported *transaction data* is *StartTransaction*, then a new transaction is started and the TOE generates a new *transaction number* by addition of 1 to the last assigned *transaction number*, includes this value in the protocol data of the *transaction log* returned to the CTSS interface component, and adds this value to the list of ongoing transactions. If the *type of the operation* is *UpdateTransaction* or *FinishTransaction* and meets the *transaction number* of an ongoing transaction, the *transaction number* in the transaction data is imported and assigned to the protocol data of the transaction log. If the type of the operation is FinishTransaction or the transaction is terminated by the TOE, the transaction number is removed from the list of ongoing transactions (cf. TR-03151 [16]).

***ST application note 3*: Transaction log message operations are always triggered via the CTSS interface role. The TOE does not end transactions independently.**

***ST application note 4*: The TOE manages a single keyID used for determining the cryptographic key for signature creation (according to TR-03153 [8]) by the CSPLight, and one to multiple clientID(s) for ERS identification.**

A UCP has the security attributes

- issuer: identifier of the authorized issuer of the UCP signing the UCP,
- signature: digital signature of the UCP generated by the authorized issuer,
- version number.

Log messages

Log messages include at least the following security attributes and the signature used by the tax inspector of the cash register inspection

- *signature counter* enumerating the *log message* continuously increasing without gaps,
- *time stamp* as time when the *log message* was created,
- *keyID* to determine the certificate to be used for verification of the digital signatures as check value of the transaction data.

The following security attributes are conditional in log messages:

- Transaction logs contain the security attribute *transaction number* assigning the *log message* to the transaction of the electronic record-keeping system and the type of operation, i.e start, update or finish transaction.
- System logs contain the security attribute *event* assigning the *log message* to the security related event of the TSF.
- Audit logs contain the security attribute *audit record* assigning the log message to security related events of the CSPLight.

3.2 Threats

T.EvadTD Evading Transaction Data

The attacker prevents sending to the TOE legally required *transaction data* in order to avoid generation of valid *transaction logs*.

T.ManipTD Manipulation of Transaction Data

The attacker manipulates *transaction data* sent by the electronic record-keeping system though the CTSS interface component to the TOE, or generates forged *transaction data* and sends them to the TOE in order to generate incorrect *transaction logs*.

T.ManipDTBS Manipulation of Data-To-Be-Signed-And-Time-Stamped

The attacker generates forged or manipulates *data-to-be-signed* sent for signing and time stamping to the CSP. A forged *transaction log* may result in forged *transaction data* provided for cash inspection. A forged *audit log* or *system log* may result in faulty interpretation of the *transaction data*.

T.ManipLM Manipulation of a Log Message

The attacker manipulates without detection a *log message* exported to the CTSS interface component. This log message is then used for cash inspection.

T.ManipLMS Manipulation of a Log Message Sequence

The attacker manipulates without detection the *log message sequence* exported to the CTSS interface component. This log message sequence is then used for cash inspection.

T.ManipTN Manipulation of Transaction Number

The attacker manipulates the TOE's internal *transaction number* used in *log messages*.

T.FaUpD Faulty Update Code Package

An attacker deploys an unauthorized manipulated *update code package* or *restores a previous TSF implementation* enabling attacks against integrity of TSF implementation, or confidentiality and integrity of user data or TSF data after installation of the manipulated *update code package*.

Application note 1: The taxpayer is the subject that owns and operates the ERS and CTSS (either directly or indirectly). The taxpayer is assumed to use an ERS equipped with a CTSS, to prevent misuse of the ERS by unauthorized persons, and to correctly tally all transactions with the ERS as required by law (cf. OSP.SecERS and OSP.ProtDev). The TOE does not protect against threats that result from temporarily or permanently not using an ERS as required by law. The taxpayer is however also considered as a potential attacker, who may use a manipulated CTSS or manipulates logs after they were produced by the CTSS.

3.3 Organizational Security Policies

OSP.SecERS Secure use of the Electronic Record-Keeping System

The taxpayer shall use an electronic record-keeping system to generate accounts, records and receipts. The electronic record-keeping system shall record separately, correctly, completely, and in real time accounts and records on all transactions that are legally required (cf. FCG [1], section 146a (1), sentence 1). The receipt shall include, besides the transaction data, the points in time when the transaction is started, completed or terminated, and the transaction number provided by the certified security device (cf. KassenSichV [2], section 6, sentence 1).

OSP.CertSecDev Certified Security Device

The electronic record-keeping system and the accounts and records generated by the electronic record-keeping system shall be protected by a certified security device (cf. FCG [1], section 146a (1), sentence 2). The security module of the certified security device generates time stamps of the start, completion, and termination of a transaction, as well as a transaction number (cf. KassenSichV [2], section 2, sentence 3).

OSP.ProtDev Protection of Electronic Record-Keeping System and Certified Security Device

The taxpayer shall correctly operate the electronic record-keeping system (cf. FCG [1], section 379 (1), sentence 1, number 4), and correctly protect the electronic record-keeping system and the certified security device (cf. FCG [1], section 379 (1), sentence 1, number 5).

OSP.ValidTrans Validation of Transactions

A sequence of transactions is valid if (1) all log messages meet the requirements for content defined in KassenSichV [2] section 2, (2) their check values according to KassenSichV [2], section 2, sentence 2, number 7 are valid digital signatures, (3) the transaction numbers are consecutive increasing without gaps (cf. KassenSichV [2], section 2, sentence 4), and (4) the points in time when the transaction starts are monotonically increasing. The sequence of log messages support detection of incomplete transactions and manipulations.

OSP.Update Authorized Update Code Packages

Update Code Packages are delivered to the TOE from the platform and are signed by the authorized issuer. The platform verifies the authenticity of the received *Update Code Package* before installation.

Application note 2: The update is performed by the platform provided by the operational environment, cf. OE.CSPPlatform for the platform architecture or OE.SMAERSPlatform for the client-server architecture.

***ST author comment to application note 2*: The TOE is based on the client-server architecture. Therefore, the CSP is not the platform of the TOE. Supported platform is one variant of *GNU/Linux* (cf. Table 1.1). The TOE is dependent on the authenticity verification provided by this platform. Please note that while the CSP is not the platform of the TOE, it is still employed by the TOE to keep secure record of the version upgrade carried out by the platform. This information is used by the TOE to prevent version downgrades.**

3.4 Assumptions

A.SMAERSPlatform Secure platform storage

The platform that executes the TOE provides mechanisms to preserve the confidentiality, integrity and to prevent rollback of stored sensitive objects, including the TOE software itself.

***ST application note 5*: The secure storage mechanism utilized by the TOE preserves the confidentiality and integrity of assets according to Table 3.1. Rollbacks (i.e. version downgrades) of the TOE software are prevented by checking the actual version against the expected version in the TOE's self-test.**

A.CSP Cryptographic service provider

A CSP is *either* remotely accessible via trusted channel to the TOE (client-server architecture) and certified as compliant to PPC-CSP-TS-Au [12], PPC-CSP-TS-Au-Cl [11], or PPC-CSPLight-TS-Au-Cl [14] running on hardware that meets the Appendix of [15] (Operational Requirements for CSPLight) as well as the requirements in Chapter 1 Section "TOE Life Cycle". *Or*, the

operational environment provides a cryptographic service provider for the TOE that is certified as compliant to PPC-CSP-TS-Au [12] or PPC-CSP-TS-Au-Cl [11] (platform architecture). The CSP exports audit records in form of audit logs meeting BSI TR-03151 [16]. Also, the CSP must provide a fully defined API description.

***ST application note 6*: The TOE requires a CSPL connected via a trusted channel.**

A.ProtComCSP Protection of Communication between TOE and CSP

The integrity of the communication data between TOE and CSP in the client-server architecture is protected via a trusted channel, and the security target must claim the package *Trusted Channel*, defined in Chapter 7 of [15]. In case of the platform architecture of the CSP, the CSP provides a secure execution environment for the TOE and protects the integrity of communication data with the TOE directly using the security services of the CSP.

A.ProtComERS Protection of Communication between TOE and Electronic Record-keeping System

The electronic record-keeping system provides transaction data whenever a transaction starts, transaction data are updated, or when the transaction is completed or terminated. The ERS and the TOE must be contained in the same physical operational environment that must protect the integrity of communication data between the TOE and the electronic record-keeping system (see Figure 2 in PP SMAERS [15]).

A.VerifLMS Verification of Log Message Sequences

The operational environment verifies the digital signatures, the transaction numbers and the time stamps of the *log messages* in sequence in order to detect forged or missing *log messages*. The certificate of the signature-verification data is securely distributed to the tax inspector. The tax inspector ensures that the transactions are created by a certified security module, e.g. in form of test transactions.

A.Admin Trustworthy Administrator

The administrator acts in a trustworthy way and must be independent of the tax payer (cf. Application note 1).

Chapter 4

Security Objectives

4.1 Security Objectives for the TOE

O.GenLM Generation of Log Messages

The TSF shall generate *transaction logs* containing

- *transaction data*, *transaction number* created by the TSF, and
- time stamps and digital signatures created by the cryptographic service provider.

The TSF shall generate *system logs.*

O.ImpExp Import of Transaction Data from and Export of Log Messages to CTSS Interface Component

The TSF shall import *transaction data* from the electronic record-keeping system through the CTSS interface component, import *audit records* from CSP and export *log messages* to the CTSS interface component.

O.IAA Authentication of Administrators

The TOE shall verify the claimed identity of the administrators by means of password.

O.SecMan Security Management

The TOE shall restrict the security management of TSF and TSF data to authenticated administrators. The TSF prevents management of the *transaction number* generation.

O.TEE Test of External Entities

The TSF shall test the presence and identity of the electronic record-keeping system and cryptographic service provider connected to the TOE, and allow generation of *transaction logs* only if both pass the tests, and must enter a secure state if any test fails.

***ST application note 7*: The TOE requires the ERS to actively report its presence regularly, which becomes documented by a system log message.**

O.TST Self-Test and Secure State

The TSF shall perform self-tests. The TSF enters a secure state if the self-test fails, or the test of the presence and identity of the electronic record-keeping system fails, or the test of the presence and identity of cryptographic service provider fails. It shall also test for new successfully installed update code packages and the correctness of the increased version number.

***ST application note 8*: The authenticity of the UCP is verified by the platform. Version downgrades are prevented by the TOE by the execution of a self-test.**

O.ImpExpUCP Secure Import and Export of User Data

The TSF shall securely export the user data and TSF data to the secure storage of the platform and import the user data and TSF data after the successful update process.

***ST application note 9*: The TOE makes use of a secure storage mechanism for its user data and TSF data. Export occurs regularly during normal operation. Import occurs at least at every start-up of the TOE including start-ups after updates.**

O.SecCommCSP Trusted Channel between TOE and CSP

The TOE shall protect the integrity of the communication between the TOE and the cryptographic service provider by means of a trusted channel.

***ST application note 10*: The ST author added O.SecCommCSP as a security objective for the TOE as a direct implementation of OE.SecCommCSP.**

4.2 Security Objectives for the Operational Environment

OE.ERS Trustworthy Electronic Record-Keeping System

The taxpayer shall correctly use an electronic record-keeping system that provides separately, correctly, completely and in real time all *transaction data* that are legally required for generation of *log messages* to the TOE (cf. Application Note 1). The electronic record-keeping system

shall support testing its presence and identity as external entity by the TOE. The electronic record-keeping system shall produce receipts including not only the transaction data, but also the points in time whenever a transaction is started, completed or terminated, as well as the transaction number provided by the certified security device.

OE.SMAERSPlatform Secure platform storage

The platform that executes the TOE has to ensure the integrity of the TOE itself and to provide secure storage which protects the integrity and confidentiality of stored security relevant objects as required (cf. Chapter 1.1.2 'TOE Type'). The platform verifies and installs the UCP.

OE.CSP Cryptographic Service Provider Component

A CSP must be *either* remotely accessible via a trusted channel to the TOE (client-server architecture) and certified as compliant to PPC-CSP-TS-Au [12], PPC-CSP-TS-Au-Cl [11], or PPC-CSPLight-TS-Au-Cl [14] running on hardware that meets Appendix of [15] (Operational Requirements for CSPLight).

Or, the operational environment shall provide a cryptographic service provider for the TOE that is certified as compliant to PPC-CSP-TS-Au [12] or PPC-CSP-TS-Au-Cl [11], i.e. using the platform architecture.

The CSP shall export audit records in form of audit logs meeting BSI TR-03151 [16].

Application note 3: The Common Criteria Protection Profile Configurations PPC-CSP-TS-Au [12], PPC-CSP-TS-Au-Cl [11], or PPC-CSPLight-TS-Au-Cl [14] require the cryptographic service provider to provide security services to digitally sign *transaction data*, to verify a signature of an update code packages, and for time services. The CSP audit records shall be exported meeting TR-03151 [16] in order to avoid transformation of the audit record into a log message. The vendor of the TOE may provide the TOE together with a certified cryptographic service provider.

***ST author comment to application note 3*: The platform is responsible for the verification of signatures of UCPs. Audit records of the CSPL are exported in the format of audit log messages in accordance with TR-03151 [16]. The TOE requires a CSPL connected via a trusted channel.**

OE.CSPPlatform CSP as Secure Platform of the TOE

In case of the platform architecture, the CSP provides a secure execution environment and security services for the TOE running on top.

Application note 4: In the typical case of a client-server architecture, the TOE and the CSP are physically separated components and the TOE cannot rely on the CSP as secure execution platform. Instead the security target shall claim the package trusted channel (cf. PP SMAERS [15] Chapter 7) to protect the integrity of the communication between the TOE and the CSP.

***ST author comment to application note 4*: The TOE follows the client-server architecture. Consequently, OE.CSPPlatform does not apply. This security target claims the package trusted channel.**

OE.Transaction Verification of Transaction

The operational environment shall verify the validity of *log message sequences* by verification of the corresponding digital signatures, the *transaction numbers* as being consecutive without gaps, and shall verify the points in time when the transaction starts as being consecutively increasing with increasing *transaction numbers* and consider the *log messages*. The taxpayer shall ensure that the cryptographic service provider holds digital signature creation data and a corresponding valid certificate. The certificate shall be securely distributed to the tax inspector.

OE.SecOEnv Secure Operational Environment

The operational environment shall protect the integrity of the communication between the electronic record-keeping system and the TOE. The administrator shall act in a trustworthy way and is assumed to be the manufacturer or integrator. The administrator must be independent of the taxpayer.

OE.SecCommCSP Secure Communication between TOE and CSP

The security target shall claim the package trusted channel (cf. SMAERS PP [15] Chapter 7) to protect the integrity of the communication between the TOE and the CSP in the client-server architecture. In case of the platform architecture, the operational environment shall protect the integrity of the communication between the TOE and the cryptographic service provider.

***ST application note 11*: The TOE follows the client-server architecture and uses a Trusted Channel between the TOE and the CSP. Consequently, OE.SecCommCSP is directly met by O.SecCommCSP.**

OE.SUCP Signed Update Code Packages

The manufacturer shall issue digitally signed *update code packages* together with its security attributes.

OE.SecUCP Secure download and authorized use of Update Code Package

The platform role shall verify the authenticity of received update code packages and install only authentic update code packages.

***ST application note 12*: In accordance with FDP_ACF.1/UCP, the platform verifies the authenticity of received update code packages whereas the CSP is employed to upgrade the stored data. The UCP is used for the creation of a new security module application or for updating an existing application.**

4.3 Security Objectives Rationale

Table 4.1 traces a security objective 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.

	T.EvadTD	T.ManipTD	T.ManipDTBS	T.ManipLM	T.ManipLMS	T.ManipTN	T.FaUpD	OSP.SecERS	OSP.CertSecDev	OSP.ProtDev	OSP.ValidTrans	OSP.Update	A.CSP	A.SMAERSPlatform	A.ProtComCSP	A.ProtComERS	A.VerifLMS	A.Admin
O.GenLM	×			×	×						×							
O.IAA				×							×							
O.ImpExp					×						×							
O.SecCommCSP			×												×			
O.SecMan						×					×							
O.TEE	×	×	×	×	×			×										
O.TST				×			×											
O.ImpExpUCP							×					×						
OE.CSP				×					×				×					
OE.SMAERSPlatform		×	×				×							×				
OE.ERS	×	×						×										
OE.SecUCP							×					×						
OE.SecCommCSP			×												×			
OE.SecOEnv	×			×	×			×		×						×		×
OE.SUCP							×					×						
OE.Transaction											×						×	

Table 4.1: Security Objective Rationale

***ST application note 13*: O.SecCommCSP has been added to meet OE.SecCommCSP**

by supporting a trusted channel between the TOE and the CSP. In addition, OE.CSPPlatform is not relevant since the TOE is based on the client-server architecture.

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 threat T.EvadTD "Evading Transaction Data" is mitigated by:

- The security objective for the TOE O.GenLM requiring the TSF to *transaction logs* containing *transaction data* and a *transaction number* generated by the TSF, and time stamps and digital signatures, therefore allowing to decide whether presented transaction data have corresponding transaction data set in the transaction data set sequence.
- The security objective for the TOE O.TEE requiring the TSF to test the presence and identity of the electronic record-keeping system connected to the TOE.
- The security objective for the operational environment OE.ERS requiring the taxpayer to use an electronic record-keeping system that provides completely and in real time all *transaction data* that are legally required for generation of *log messages* to the TOE.
- The security objective for the operational environment OE.SecOEnv requiring the operational environment to protect the communication between ERS and TOE against manipulation and perturbation.

The threat T.ManipTD "Manipulation of Transaction Data" is mitigated by:

- The security objective for the TOE O.TEE requiring the TSF to test the presence and identity of the CTSS interface component connected to the TOE.
- The security objective for the operational environment OE.ERS requiring the taxpayer to use an electronic record-keeping system that provides correctly, completely and in real time all transaction data that are legally required for generation of *log messages* to the TOE.
- The security objective for the operational environment OE.SMAERSPlatform requiring the operational environment to protect the TOE against manipulation and misuse.

The threat T.ManipDTBS "Manipulation of Data-To-Be-Signed-And-Time-Stamped" is mitigated by:

- The security objective for the TOE O.TEE to test the presence and identity of the CSP connected to the TOE.
- In case of the platform architecture, the OE.CSPPlatform "CSP as Secure Platform of the TOE" requires the CSP to provide a secure execution environment. In case of the client-server architecture, the OE.SMAERSPlatform.
- The security objective for the operational environment OE.SecCommCSP "Secure communication between TOE and CSP" ensures the protection of the integrity of the communication between the TOE and the cryptographic service provider. The operational

environment shall protect the integrity of the communication between the TOE and the cryptographic service provider. In case of the client-server architecture the TOE and the CSP component are physically separated components. The integrity of the communication between the TOE and the CSP shall be protected by means of a trusted channel as provided by the CSP according to PPC-CSP-TS-Au [12], PPC-CSP-TS-Au-Cl [11], PPC-CSPLight-TS-Au-Cl [14] and by the TOE claiming the package *trusted channel* between the TOE and the CSP, cf. PP SMAERS [15] Chapter 7.

***ST application note 14*: The platform architecture is not used by the TOE. The TOE follows the client-server architecture. Consequently, OE.CSPPlatform is not assumed, and OE.SecCommCSP "Secure communication between TOE and CSP" is met by O.SecCommCSP "Trusted channel between TOE and CSP".**

The threat T.ManipLM "Manipulation of Log Messages" is countered by:

- The security objective for the TOE O.GenLM "Generation of Log Messages" by means of digital signatures generated by the CSP, which allows to detect manipulation of transaction data sets according to OE.Transaction.
- The security objective for the TOE O.IAA requiring the TSF to authenticate administrators by means of a password.
- The security objective for the TOE O.TEE "Test of External Entities" requiring the TSF to test the presence and identity of the CSP connected to the TOE.
- The security objective for the TOE O.TST "Self-Test and Secure State" detects failure and prevents generation of transaction data sets if time source is not available or the test of CSP fails.
- The security objectives for the operational environment OE.CSP "Cryptographic Service Provider Component" ensure the availability of a certified CSP for generation of time stamps and digital signatures, and distribution of the certificate linked to the taxpayer for signature verification.
- The security objective for the operational environment OE.SecOEnv "Secure Operational Environment" protecting the communication between ERS and TOE.

The threat T.ManipLMS "Manipulation of a Log Message Sequence" is countered by:

- The security objective for the TOE O.GenLM "Generation of Log Messages" requiring the TSF to generate *log messages* containing *transaction data* imported from the electronic record-keeping system, requiring the TSF to generate time stamps whenever a transaction starts, is completed or aborted, and requiring the TSF to create a *transaction number* and a digital signature of the *transaction data* using the digital signature-creation service of the cryptographic service provider.
- The security objective for the TOE O.ImpExp "Import of Transaction Data from and Export of Log Message to CTSS Interface Component" requiring the TSF to import *transaction data* from the electronic record-keeping system through the CTSS interface component and export *log messages* to the CTSS interface component.

- The security objective for the TOE O.TEE “Test of External Entities” requiring the TSF to test the availability of the CTSS interface component and CSP connected to the TOE.
- The security objective for the operational environment OE.SecOEnv “Secure Operational Environment” protecting the communication between ERS and TOE.

The threat T.ManipTN “Manipulation of Transaction Number” is countered by the security objectives for the TOE O.SecMan TSF preventing management of the *transaction number* generation.

The threat T.FaUpD “Faulty Update Code Package” is countered by:

- The security objectives for the TOE O.ImpExpUCP “Secure Import and Export of User Data” ensuring that user data are exported and imported after successful update process.
- The security objective for the TOE O.TST “Self-Test and Secure State” ensuring a correctly increased version number after installation of an update code package.
- The security objective for the operational environment OE.SUCP ensures that the authentic *update code packages* are signed and distributed with security attributes.
- The OE.SecUCP “Secure download and authorized use of Update Code Package” ensures that only authentic UCPs are installed.
- The OE.SMAERSPlatform ensures verifying the UCP.

The organizational security policy OSP.SecERS “Secure use of the electronic record-keeping system” is directly enforced by:

- The security objective for the TOE O.TEE requiring the TSF to test the presence and identity of the ERS as an external entity.
- The security objective for the operational environment OE.ERS “Trustworthy Electronic Record-Keeping System”.
- The security objective for the operational environment OE.SecOEnv “Secure Operational Environment” protecting the communication of ERS and TOE.

The organizational security policy OSP.CertSecDev “Certified Security Device” is directly enforced by the security objectives for the operational environment OE.CSP “Cryptographic Service Provider Component” and the certification conformant to this protection profile.

The organizational security policy OSP.ProtDev “Protection of ERS and Security Module” is directly ensured by the security objective for the operational environment OE.SecOEnv “Secure Operational Environment”.

The organizational security policy OSP.ValidTrans “Validation of Transactions” is enforced by the security objectives for the TOE

- the security objective for the TOE O.GenLM "Generation of Log Messages" requiring the TSF to generate *log messages* containing *transaction data* imported from the electronic record-keeping system, to generate time stamps whenever a transaction starts, is completed or aborted, and to generate a *transaction number* and a digital signature of the *transaction data* created using the digital signature-creation service of the cryptographic service provider,
- the security objectives for the TOE O.IAA "Authentication of Administrators" requiring the TSF to authenticate the administrators by means of password,
- the security objective for the TOE O.ImpExp "Import of Transaction Data from and Export of Log Message to CTSS Interface Component" requiring the TSF to import *transaction data* from the electronic record-keeping system through the CTSS interface component and export *log messages* to the CTSS interface component,
- the security objective for the TOE O.SecMan "Security Management" preventing manipulation of the *transaction numbers* and limiting the authorized manipulation of the time source to administrators,
- the security objective for the operational environment OE.Transaction "Verification of Transaction" ensures the condition for verification of the digital signature of the transaction data set.

The organizational security policy OSP.Update "Authorized Update Code Packages" is implemented by the security objective for the operational environment OE.SUCP "Signed Update Code Packages" ensuring digital signature of secure *update code packages* together with its security attributes and the security objectives for the operational environment OE.SecUCP "Secure Download and Authorized Use of Update Code Package" ensuring verification of digital signature.

The assumption A.CSP "Cryptographic Service Provider" is directly implemented by the security objective for the operational environment OE.CSP "Cryptographic Service Provider Component".

The assumption A.SMAERSPlatform is directly implemented by the security objective for the operational environment OE.SMAERSPlatform that requires secure storage of sensitive objects.

The assumption A.ProtComCSP "Protection of Communication between TOE and CSP" is directly implemented by the security objective for the operational environment OE.SecCommCSP which requires the protection of the communication between the TOE and the CSP. In case of the platform architecture, the OE.CSPPlatform requires the CSP to provide a secure execution environment. In case of the client-server architecture, the TOE and the CSP component are physically separated components. The integrity of the communication between the TOE and the CSP shall be protected by means of a trusted channel as provided by the CSP according to PPC-CSP-TS-Au [12], PPC-CSP-TS-Au-Cl [11], or PPC-CSPLight-TS-Au-Cl [14] and by the TOE claiming the package *trusted channel*, cf. PP SMAERS [15] Chapter 7.

***ST application note 15*: The client-server architecture is used and the communication between the TOE and the CSP is protected by a trusted channel.**

The assumption A.ProtComERS "Protection of Communication between TOE and Electronic Record-Keeping System" is directly implemented by the security objectives for the operational environment OE.SecOEnv "Secure Operational Environment" protecting the integrity of the communication between the electronic record-keeping system and the TOE.

The assumption A.VerifLMS “Verification of Log Message Sequences” is directly implemented by the security objective for the operational environment OE.Transaction “Verification of Log Message Sequences”.

The assumption A.Admin “Trustworthy Administrator” is directly implemented by the security objective for operational environment OE.SecOEnv “Secure Operational Environment”.

Chapter 5

Extended Components Definition

The extended components FIA_API.1 and FCS_RNG.1 are used only in the package Package Trusted Channel between TOE and CSP, cf. PP SMAERS [15] Chapter 7.

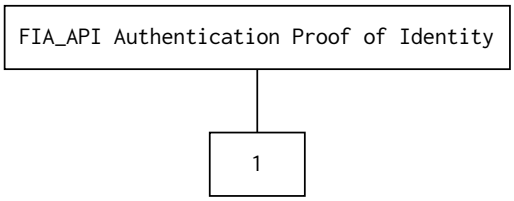
5.1 Authentication Proof of Identity (FIA_API)

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.

Family Behaviour

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

Component Levelling:



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

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.

Audit: FIA_API.1

There are no auditable events foreseen.

FIA_API.1 Authentication Proof of Identity

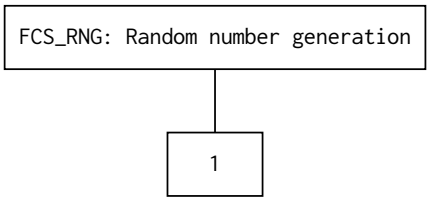
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_API.1.1	The TSF shall provide a [assignment: *authentication mechanism*] to prove the identity of the [assignment: *object, authorized user or role*] to an external entity.

5.2 Generation of Random Numbers (FCS_RNG)

Family Behaviour

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

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.

Management: FCS_RNG.1

There are no management activities foreseen.

Audit: FCS_RNG.1

There are no auditable events foreseen.

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*].

Chapter 6

Security Requirements

Part 2 of the CC defines a set of operations that may be applied to requirements.

The CC allows several operations to be performed on functional and assurance requirements: refinement, selection, assignment, and iteration. Each of these operations is used in this PP.

The **refinement** operation is used to add detail to a requirement, and thus further restricts a requirement. Refinement of security and assurance 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 (example: **addition**). In cases where words from a CC requirement component were deleted, these words are crossed out (example: ~~deletion~~). Addition refinements made by the ST author are bold and colored blue (example: **addition**). Deletion refinements made by the ST author are bold, crossed out and colored blue (example: **~~deletion~~**).

The **selection** operation is used to select one or more options provided by the CC in stating a requirement. Selections that have been made by the PP authors are denoted as italic text and the original text of the component is given by a footnote (example: *option 1*[1]). Selections made by the ST author are italic, colored blue, and have a footnote documenting the selection by the ST author (example: *option 1*[2]).

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 authors are denoted by showing as italic text and the original text of the component is given by a footnote (example: *made assignment*[3]). Assignments made by the ST author are italic, colored blue, and have a footnote documenting the assignment by the ST author (example: *made assignment*[4]).

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.

[1] [selection (by PP author): *option 1, option 2*]
[2] [selection (by ST author): *option 1, option 2*]
[3] [assignment (by PP author): *assignment to be made*]
[4] [assignment (by ST author): *assignment to be made*]

6.1 Security Functional Requirements

6.1.1 Security Management

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, administrator, CTSS interface role and CSP role, the role TR Administrator*[5] [6] .
FMT_SMR.1.2	The TSF shall be able to associate users with roles.

FMT_SMF.1 Specification of Management Functions

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FMT_SMF.1.1	The TSF shall be capable of performing the following management functions: *(1) management of security functions behaviour (cf. FMT_MOF.1),* *(2) management of authentication reference data (cf. FMT_MTD.1/AD, FMT_MTD.3/PW),* *(3) management of security attributes (cf. FMT_MTD.3/PW, FMT_MSA.3, FMT_MSA.4),* *(4) no additional security management functions*[7] [8].

[5] [assignment (by ST author): *other roles*]
[6] [assignment (by PP author): *authorized identified roles*]
[7] [assignment (by ST author): *list additional of security management functions to be provided by the TSF*]
[8] [assignment (by PP author): *list of management functions to be provided by the TSF*]

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 and disable*[9] the functions ~~*password authentication according to FIA_UAU.5.2, clause (2) if defined*~~[10] to *administrator*[11] ,

(2) ***determine the behaviour of and modify the behaviour of***[12] **the function *FDP_ACF.1/LM by definition of a life time limit of ongoing transactions after which the transaction is terminated by the TSF***[13] **to *administrator***[14] ,

(3) ***determine the behaviour of***[15] **the function *FPT_TEE.1 by definition of the identity and features to be tested of ERS***[16] **to *administrator***[17] ,

(4) ***determine the behaviour of***[18] **the function *FPT_TEE.1 by definition of the identity and features to be tested of CSP***[19] **to *administrator***[20] ,

(5) ***determine the behaviour of and modify the behaviour of***[21] **the function *FPT_TEE.1 in case the test of CTSS interface component or CSP fails***[22] **to *administrator***[23] ,

(6) ***determine the behaviour of and modify the behaviour of***[24] **the functions *select the auditable events according to FAU_GEN.1/SYS***[25] **to *administrator***[26] ,

(7) ***determine the behaviour of and modify the behaviour of***[27] **the functions *automatic export of audit trails according to FAU_STG.3.1/SYS clause (1)***[28] **to *administrator***[29]

[9] [selection (by PP author): *determine the behaviour of, disable, enable, modify the behaviour of*]
[10] [assignment (by PP author): *list of functions*]
[11] [assignment (by PP author): *the authorised identified roles*]
[12] [selection (by PP author): *determine the behaviour of, disable, enable, modify the behaviour of*]
[13] [assignment (by PP author): *list of functions*]
[14] [assignment (by PP author): *the authorised identified roles*]
[15] [selection (by PP author): *determine the behaviour of, disable, enable, modify the behaviour of*]
[16] [assignment (by PP author): *list of functions*]
[17] [assignment (by PP author): *the authorised identified roles*]
[18] [selection (by PP author): *determine the behaviour of, disable, enable, modify the behaviour of*]
[19] [assignment (by PP author): *list of functions*]
[20] [assignment (by PP author): *the authorised identified roles*]
[21] [selection (by PP author): *determine the behaviour of, disable, enable, modify the behaviour of*]
[22] [assignment (by PP author): *list of functions*]
[23] [assignment (by PP author): *the authorised identified roles*]
[24] [selection (by PP author): *determine the behaviour of, disable, enable, modify the behaviour of*]
[25] [assignment (by PP author): *list of functions*]
[26] [assignment (by PP author): *the authorised identified roles*]
[27] [selection (by PP author): *determine the behaviour of, disable, enable, modify the behaviour of*]
[28] [assignment (by PP author): *list of functions*]

Application note 5: The refinements of FMT_MOF.1, bullet (2) to (7) are made in order to avoid iterations of the component. The life time of a transaction starts with receiving the *transaction data* with *type of operation* being *StartTransaction*.

***ST application note 16*: Auditable events of the TOE are as soon as possible signed by the CSP, and the system log message is exported as soon as possible to the storage in accordance with FDP_ACF.1/LM and FDP_ETC.2/LM. As the TOE has direct interface to the storage, automatic exports as defined in FAU_STG.3/SYS are carried out in response to the auditable action.**

***ST application note 17*: The TOE does not terminate ongoing transactions independently as this is the responsibility of the ERS.**

[29] [assignment (by PP author): *the authorised identified roles*]

FMT_MSA.1 Management of security attributes

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MSA.1.1	The TSF shall enforce the *log message SFP and update SFP*[30] to restrict the ability to

(1) *define the set of accepted values of*[31] the security attributes *"clientID"*[32] to ~~CTSS interface role~~ TR administrator[33] ,

(2) ***define depending on the clientID***[34] **the *identity of the signature-creation key (keyID) to be used for the transaction log***[35] **to *CTSS interface role***[36] **,**

(3) ***define***[37] **the *identity of the signature-creation key (keyID) to be used for the system log and audit logs***[38] **to *CTSS interface role***[39] **,**

(4) ***increase by 1***[40] **the *internally stored security attribute "transaction number" whenever a transaction is started***[41] **to *subjects in CTSS interface role***[42] **,**

(5) ***modify***[43] **the *TD security attribute "transaction number" imported from the TD***[44] **to *none***[45] **.**

(6) ***increase***[46] **the *security attribute "version number" of UCP after successful installation***[47] **to *CSP role***[48] **.**

Application note 6: The refinements of FMT_MSA.1 are made in order to avoid iteration of the component.

[30] [assignment (by PP author): *access control SFP(s), information flow control SFP(s)*]
[31] [selection (by PP author): *change_default, query, modify, delete [assignment: other operations]*]
[32] [assignment (by PP author): *list of security attributes*]
[33] [assignment (by PP author): *the authorised identified roles*]
[34] [selection (by PP author): *change_default, query, modify, delete [assignment: other operations]*]
[35] [assignment (by PP author): *list of security attributes*]
[36] [assignment (by PP author): *the authorised identified roles*]
[37] [selection (by PP author): *change_default, query, modify, delete [assignment: other operations]*]
[38] [assignment (by PP author): *list of security attributes*]
[39] [assignment (by PP author): *the authorised identified roles*]
[40] [selection (by PP author): *change_default, query, modify, delete [assignment: other operations]*]
[41] [assignment (by PP author): *list of security attributes*]
[42] [assignment (by PP author): *the authorised identified roles*]
[43] [selection (by PP author): *change_default, query, modify, delete [assignment: other operations]*]
[44] [assignment (by PP author): *list of security attributes*]
[45] [assignment (by PP author): *the authorised identified roles*]
[46] [selection (by PP author): *change_default, query, modify, delete [assignment: other operations]*]
[47] [assignment (by PP author): *list of security attributes*]
[48] [assignment (by PP author): *the authorised identified roles*]

***ST application note 18*: A code update results in the generation of a system log message signed by the CSP that documents the increase of the security attribute "version number" of UCP. The new version number is protected against modification and cannot be decreased.**

***ST application note 19*: The role TR administrator alway implies the role CTSS Interface, which makes this restriction a refinement. fiskaly GmbH wants to restrict this administrative function to the TR administrator role to prevent accidantial missconfiguration and harmonize the role understanding between this ST and TR 03153 [8].**

FMT_MSA.3 Static attribute initialisation

Hierarchical to:	No other components.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.3.1	The TSF shall enforce the *log message SFP and update SFP*[49] to provide *restrictive*[50] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow the *none*[51] to specify alternative initial values to override the default values when an object or information is created.

[49] [assignment (by PP author): *access control SFP, information flow control SFP*]
[50] [selection (by PP author): *choose one of: restrictive, permissive,[assignment: other property]*]
[51] [assignment (by PP author): *the authorised identified roles*]

6.1.2 User identification and authentication

FIA_ATD.1 User attribute definition

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_ATD.1.1	The TSF shall maintain the following list of security attributes: belonging to ~~individual users~~ **administrator**: *(1) identity,* *(2) authentication reference data,* *(3) role*[52] and (a) **security attribute *identity*** *clientID*[53] ***belonging to the ERS***[54] (b) **security attribute *identity*** *PACE Password*[55] ***belonging to the CSP.***[56]

Application note 7: The refinements distinguish between the sets of security attributes maintained for authenticated users by an administrator, and the tested users ERS and CSP according to FTP_TEE.1. The security attributes are defined for users by the administrator according to FMT_MSA.1.

[52] [assignment (by PP author): *list of security attributes*]
[53] [assignment (by ST author): *additional security attributes*]
[54] [assignment (by PP author): *list of security attributes*]
[55] [assignment (by ST author): *additional security attributes*]
[56] [assignment (by PP author): *list of security attributes*]

FMT_MTD.1/AD Management of TSF data - Authentication data

Hierarchical to:	No other components.
Dependencies:	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MTD.1.1/AD	The TSF shall restrict the ability to (1) *delete and create*[57][58] the *authentication data record of all authorized users*[59] to *administrator*[60] (2) ***modify***[61] **the *authentication reference data***[62] **to *the corresponding authorized user***[63] **.**

FMT_MTD.3/PW Secure TSF data - Password

Hierarchical to:	No other components.
Dependencies:	FMT_MTD.1 Management of TSF data
FMT_MTD.3.1/PW	The TSF shall ensure that only secure values are accepted for *passwords*[64] **and enforce changing initial passwords after first successful authentication of the user to a different secure operational password**.

[57] (by PP author) "create" denotes initial creation and setting a new value in case a user forgot/lost their authentication data
[58] [selection (by PP author): *change_default, query, modify, delete, clear, [assignment: other operations]*]
[59] [assignment (by PP author): *list of TSF data*]
[60] [assignment (by PP author): *the authorised identified roles*]
[61] [selection (by PP author): *change_default, query, modify, delete, clear, [assignment: other operations]*]
[62] [assignment (by PP author): *list of TSF data*]
[63] [assignment (by PP author): *the authorised identified roles*]
[64] [assignment (by PP author): *list of TSF data*]

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 *3(for administrator) and 5 (for TR administrator)*[65][66] unsuccessful authentication attempts occur related to *password authentication*[67] .
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been *met*[68] , the TSF shall *block password authentication for 2^(number of consecutively failed retries) seconds (for administrator); block password authentication until password gets unblocked with PUK (for TR administrator)*[69] .

[65] [assignment (by ST author): positive integer number]
[66] [selection (by ST author): *[assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]*]
[67] [assignment (by ST author): *list of authentication events*]
[68] [selection (by ST author): *met, surpassed*]
[69] [assignment (by ST author): *list of actions*]

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.*[70]
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*[71] .
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) A subject is associated with attribute 'identity' and 'CTSS interface role' after the ERS is successfully tested according to FPT_TEE.1.* *(2) A subject is associated with attribute 'identity' and 'CSP role' after the CSP is successfully tested according to FPT_TEE.1.* *(3) A subject is associated with attribute 'identity' and 'administrator role' after successful authentication.*[72]

FIA_UID.1 Timing of identification

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_UID.1.1	The TSF shall allow *self test according to FPT_TST.1*[73] 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.

[70] [assignment (by PP author): *list of user security attributes*]
[71] [assignment (by PP author): *rules for the initial association of attributes*]
[72] [assignment (by PP author): *rules for the changing of attributes*]
[73] [assignment (by PP author): *list of TSF mediated actions*]

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) testing of external entity ERS according to FPT_TEE.1 and starting the subject CTSS interface component if testing was successful and the role CTSS interface component is activated,* *(3) testing of external entity CSP according to FPT_TEE.1 and start the subject CSP if testing was successful,*[74] *(4) setting a proxy server and timeout value used in the communication with the CSP and exporting tar archives of log messages to the ERS, which were previously exported to storage*[75] 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.

***ST application note 20*: FIA_UAU.1.1 (5) allows an unidentified user to set a proxy server. This might be required to be done by the tax payer to allow the TOE to connect the CSP. Note that the communication between TOE and CSP is protected by the Trusted Channel, i.e. the proxy server only relays information and is not able to modify or read the exchanged information. When the network settings of the tax payer change, adopting these settings might be required to allow the TOE to pass the selftest.**

[74] [assignment (by PP author): *list of TSF mediated actions*]
[75] [assignment (by ST author): *list of other TSF mediated actions*]

FIA_UAU.5 Multiple authentication mechanisms

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_UAU.5.1	The TSF shall provide *password authentication*[76] to support user authentication.
FIA_UAU.5.2	The TSF shall authenticate any user's claimed identity according to the *rule that*

(1) password authentication shall be used for an administrator,

(2) successful trusted channel establishment via PACE shall be used for CSP role, providing a clientID that is in the list of accepted clientIDs shall be used for ERS connecting via the CTSS interface role, password and PUK shall be used for TR administrator role[77] [78].

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 *power on or reset*[79] .

[76] [assignment (by PP author): *list of multiple authentication mechanisms*]
[77] [assignment (by ST author): *additional rules describing how the multiple authentication mechanisms provide authentication*]
[78] [assignment (by PP author): *rules describing how the multiple authentication mechanisms provide authentication*]
[79] [assignment (by PP author): *list of conditions under which re-authentication is required*]

6.1.3 User Data Protection

FDP_ACC.1/LM Subset access control – Access to Logging

Hierarchical to:	FDP_ACC.1 Subset access control
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1/LM	The TSF shall enforce the *Log Message SFP*[80] on

(1) subjects:

(a) subject acting for CTSS interface component,
(b) subject acting for CSP;

(2) objects:

(a) transaction data,
(b) audit record,
(c) data-to-be-signed,
(d) protocolData with signature,
(e) log message,
(f) commands;

(3) operations:

(a) import,
(b) export.[81]

[80] [assignment (by PP author): *access control SFP*]
[81] [assignment (by PP author): *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

FDP_ACF.1/LM Security attribute based access control – Access to TDS

Hierarchical to:	FDP_ACC.1 Subset access control
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1/LM	The TSF shall enforce the *Log Message SFP*[82] to objects based on the following:

(1) subjects:

(a) subject in CTSS interface role with security attribute activated or deactivated,

(b) subject in CSP role;

(2) objects:

(a) transaction data,

(b) audit record,

(c) data-to-be-signed,

(d) protocolData with signature,

(e) log message,

(f) commands.[83]

[82] [assignment (by PP author): *access control SFP*]

[83] [assignment (by PP author): *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

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

(1) A subject in activated CTSS interface role is allowed to

(a) import the transaction data from the CTSS interface component according to FDP_ITC.2/TD,

(b) import commands from activated CTSS interface component,

(c) export the DTBS of transaction log and system log to the CSP according to FDP_ETC.2/DTBS,

(d) import the protocolData with signature from the CSP according to FDP_ITC.2/TSS,

(e) export the transaction log and system log to the CTSS interface component according to FDP_ETC.2/LM.

(2) A subject in activated CTSS interface role is allowed to terminate the transaction after time limit defined according to FMT_MOF.1.1 clause (2) is reached.

(3) A subject in CSP role is allowed to import audit records from the CSP according to FDP_ITC.2/TSS and to export audit logs to the CTSS interface component according to FDP_ETC.2/LM.[84]

FDP_ACF.1.3/LM The TSF shall explicitly authorise access of subjects to objects based on the following additional rules *The TOE exports Tar Archives of log messages, which were already exported to storage to the ERS, even in secure state*[85] .

FDP_ACF.1.4/LM The TSF shall explicitly deny access of subjects to objects based on the rules

(1) a user in other role than CTSS interface role is not allowed to perform actions listed in FDP_ACF.1.2/LM clause (1) and (2).

(2) a user in other role than CSP role is not allowed to perform actions listed in FDP_ACF.1.2/LM clause (3).[86]

***ST application note 21*: The assigned rule in FDP_ACF.1.3/LM catches the spirit of the original architecture of the PP. In the PP, the export of TAR Archives of log messages from the storage is not part of the TOE, but part of the external CTSS interface component. For this TOE, there is no such external component. Adding this rule makes it possible to export existing data in case, the connection to the CSP fails, which is feasible and was intented by the original design pattern.**

[84] [assignment (by PP author): *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]

[85] [assignment (by ST author): *rules, based on security attributes, that explicitly authorise access of subjects to objects*]

[86] [assignment (by PP author): *rules, based on security attributes, that explicitly deny access of subjects to objects*]

FDP_ITC.2/TD Import of user data with security attributes – Transaction 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/TD	The TSF shall enforce the *log message SFP*[87] when importing ~~user data~~ **transaction data** controlled under the SFP, from outside of the TOE.
FDP_ITC.2.2/TD	The TSF shall use the security attributes associated with the imported ~~user data~~ **transaction data**.
FDP_ITC.2.3/TD	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the ~~user data~~ **transaction data** received.
FDP_ITC.2.4/TD	The TSF shall ensure that interpretation of the security attributes of the imported ~~user data~~ **transaction data** is as intended by the source of the user data.
FDP_ITC.2.5/TD	The TSF shall enforce the following rules when importing ~~user data~~ **transaction data** controlled under the SFP from outside of the TOE: *(1) The TSF shall import the transaction data with the security attribute clientID if the clientID is in the set of accepted values according to FMT_MSA.1. If the clientID is not in the set of accepted values the TSF must not import the transaction data.* *(2) The TSF shall import the transaction data with the security attribute 'type of the operation'.* *(3) The transaction data shall be imported with the security attribute 'transaction number' if the 'type of the operation' is UpdateTransaction or FinishTransaction and the transaction number meets a transaction number of an ongoing transaction.* *(4) The TSF shall import audit records from CSP.*[88]

[87] [assignment (by PP author): *access control SFP(s) and/or information flow control SFP(s)*]
[88] [assignment (by PP author): *additional importation control rules*]

FDP_ETC.2/DTBS 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/DTBS	The TSF shall enforce the *log message SFP*[89] when exporting ~~user data~~ **data-to-be-signed**, controlled under the SFP(s), ~~outside of the TOE~~ **to the CSP**.
FDP_ETC.2.2/DTBS	The TSF shall export the user data with the ~~user data's associated~~ **security attributes associated with the data-to-be-signed**.
FDP_ETC.2.3/DTBS	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported ~~user data~~ **data-to-be-signed**.
FDP_ETC.2.4/DTBS	The TSF shall enforce the following rules when user data is exported from the TOE: *(1) Data-to-be-signed shall be exported for generation of a log message with security attribute identifying the private signature key to be used by FDP_DAU.2/TS according to PPC-CSP-TS-Au [12], PPC-CSP-TS-Au-Cl [11], PPC-CSPLight-TS-Au-Cl [14].*[90]

[89] [assignment (by PP author): *access control SFP*]
[90] [assignment (by PP author): *additional exportation control rules*]

FDP_ITC.2/TSS Import of user data with security attributes – Time stamp and signature

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path]
FDP_ITC.2.1/TSS	The TSF shall enforce the *log message SFP*[91] when importing ~~user data~~ **protocolData with signature and audit records**, controlled under the SFP, from ~~outside of the TOE~~ **the CSP**.
FDP_ITC.2.2/TSS	The TSF shall use the security attributes associated with the imported user data.
FDP_ITC.2.3/TSS	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the ~~user data~~ **protocolData with signature and audit records** received.
FDP_ITC.2.4/TSS	The TSF shall ensure that interpretation of the security attributes of the imported ~~user data~~ **protocolData with signature and audit records** is as intended by the source of the user data.
FDP_ITC.2.5/TSS	The TSF shall enforce the following rules when importing ~~user data~~ **protocolData with signature and audit records** controlled under the SFP from ~~outside of the TOE~~ **the CSP** *no additional importation control rules*[92] .

Application note 8: The CSP shall generate and return to the TOE at least the signature counter of the used signature-creation key, the time stamp and the signatures for the *data-to-be-signed* exported by the TOE according to FDP_ETC.2/DTBS. The CSP shall generate time stamps according to FDP_DAU.2/TS using a time source according to FPT_STM.1 (cf. PPC-CSP-TS-Au [12], PPC-CSP-TS-Au-Cl [11], PPC-CSPLight-TS-Au-Cl [14]. Note, the TOE of this protection profile may use the CSP to provide time stamps by an administrator settable internal clock; cf. selection clause (4) in FPT_STM.1.1). If the CSP meets TR-03151 [16] for the *transaction logs*, then the CSP returns a *log message* to the TOE. If the CSP generates the time stamp and signatures with a signature counter, then the TOE shall compile the *log message* according to TR-03153 [8]. The signature counter and the time stamp of *transaction logs* and of audit data received as *audit logs* may be used to test the CSP according to FPT_TEE.1.

[91] [assignment (by PP author): *access control SFP(s) and/or information flow control SFP(s)*]
[92] [assignment (by ST author): *additional importation control rules*]

FDP_ETC.2/LM Export of user data with security attributes – Log message

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_ETC.2.1/LM	The TSF shall enforce the *log message SFP*[93] when exporting user data **log message**, controlled under the SFP(s), ~~outside of the TOE~~ **to CTSS interface component**.
FDP_ETC.2.2/LM	The TSF shall export the user data with the user data's associated security attributes.
FDP_ETC.2.3/LM	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/LM	The TSF shall enforce the following rules when user data is exported from the TOE: *Log messages shall be exported with security attribute*

(1) transaction logs:

(a) transaction number of the transaction identifying the log messages which belongs to the transaction,

(b) signature counter of the private signature key used by FDP_DAU.2/TS according to PPC-CSP-TS-Au [12], PPC-CSP-TS-Au-Cl [11], PPC-CSPLight-TS-Au-Cl [14] enumerating all log messages,

(c) type of the operation

(d) time stamp when the log message was signed,

(e) keyID as hash value of the public key for verification of the signature,

(f) signature for verification of the authenticity of the certified data and protocol data.

(2) system logs:

(a) type of the operation or TSF security event,

(b) signature counter of the private signature key used by FDP_DAU.2/TS according to PPC-CSP-TS-Au [12], PPC-CSP-TS-Au-Cl [11], PPC-CSPLight-TS-Au-Cl [14]

(c) time stamp when the log message was signed,

(d) keyID as hash value of the public key for verification of the signature,

(e) signature for verification of the authenticity of the certified data and protocol data.

(3) audit records of the CSP shall be exported unchanged as audit logs to the CTSS interface component.[94]

[93] [assignment (by PP author): *access control SFP*]

Application note 9: The CTSS interface component does not implement any security functionality addressed in this PP and imports and stores log message received from the TOE as user data.

FPT_TDC.1 Inter-TSF basic TSF data consistency

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_TDC.1.1	The TSF shall provide the capability to consistently interpret *(1) clientID,* *(2) type of the operation,* *(3) transaction number,* *(4) signature counter,* *(5) time stamp,* *(6) keyID as hash value of the public key,* *(7) signature*[95] when shared between the TSF and another trusted IT product.
FPT_TDC.1.2	The TSF shall use *BSI TR-03151 [16] and BSI TR-03153 [8]*[96] when interpreting the TSF data from another trusted IT product.

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 *(1) transaction numbers building a strong increasing sequence without gaps,* *(2) Time stamps of the log messages building a non-decreasing sequence with consideration of adjustments of the CSP's time source.*[97]

[94] [assignment (by PP author): *additional exportation control rules*]
[95] [assignment (by PP author): *list of TSF data types*]
[96] [assignment (by PP author): *list of interpretation rules to be applied by the TSF*]
[97] [assignment (by PP author): *list of security attributes*]

Application note 10: The rules may be enforced by internally storing of the *transaction number* and last time stamp provided by the CSP in the log messages.

***ST author comment to application note 10*: The TOE stores the last signature counter value and time stamp provided by the CSP. The transaction number is managed and stored by the TOE.**

FMT_MSA.4 Security attribute value inheritance

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

FMT_MSA.4.1 The TSF shall use the following rules to set the value of security attributes:

(1) The TSF uses the security attribute clientID imported with transaction data to determine the signature-creation key be used by FDP_DAU.2/TS with ECDSA in PPC-CSP-TS-Au [12], PPC-CSP-TS-Au-Cl [11], PPC-CSPLight-TS-Au-Cl [14]) to sign the corresponding log message as defined according to FMT_MSA.1.

(2) If the type of the operation of imported transaction data is StartTransaction, then the last internally generated transaction number of the respective keyID shall be increased by 1, and this value shall be assigned to the ongoing transaction and the transaction log of imported transaction data.

(3) If the type of the operation of imported transaction data is UpdateTransaction or FinishTransaction and meets the transaction number of an ongoing transaction, then the transaction number of the imported transaction data shall be assigned to the protocol data of the transaction log.[98]

[98] [assignment (by PP author): *rules for setting the values of security attributes*]

6.1.4 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 according to FPT_TST.1 fails,* *(2) test of ERS according to FPT_TEE.1 fails,* *(3) test of CSP according to FPT_TEE.1 fails.*[99] **The TSF shall exit the secure state only if the self-test, the test of the ERS and the test of the CSP are passed.**

Application note 11: The self-test according to FPT_TST.1 and test of external entities according to FPT_TEE.1 cause the TOE to enter a secure state if the self-test or the tests of the ERS or CSP fail. The exit of the secure state requires all conditions listed in the refinement being fulfilled.

[99] [assignment (by PP author): *list of types of failures in the TSF*]

FPT_TEE.1 Testing of external entities

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_TEE.1.1	The TSF shall run a suite of tests *during start-up, periodically during normal operation, user initiated shutdown and before exiting the secure state according to FPT_FLS.1*[100] to check the fulfilment of *(1) ERS identity clientID*[101] *and* *(2) CSP identity successful PACE execution, time stamp, and signature counter value*[102] .[103] **The tests include the identification of the TOE to the tested device.**
FPT_TEE.1.2	If the test fails, the TSF shall enter the secure state according to FPT_FLS.1 *none additional action*[104] [105] .

Application note 12: The administrator may by able to define the actions in FPT_TEE.1 according to FMT_MOF.1.1 (5). In case of a failure additional actions may e.g. include reading the stored audit logs. The suite of tests determine whether the configured CSP is available for the TOE and log messages can be signed. The TOE may use signature counter and time stamps received from the CSP to test it. The signature counter shall increase strong monotonically without gaps because any gap may indicate unauthorized signature-creation. The tests of the CSP should allow the CSP to identify the TOE as user of the CSP, cf. FIA_UID.1.1 clause (2) in PP CSP [9], PP CSPLight [13]. Please refer for further explanations to the user notes and evaluator notes in CC part 2 [6], chapter J.12.

[100] [selection (by PP author): *during initial start-up, periodically during normal operation, at the request of an authorised user, [assignment: other conditions]*]
[101] [assignment (by ST author): *list of properties of the ERS*]
[102] [assignment (by ST author): *list of properties of the CSP*]
[103] [assignment (by PP author): *list of properties of the external entities*]
[104] [selection (by ST author): *none additional action, [assignment: additional action(s)]*]
[105] [assignment (by PP author): *action(s)*]

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, at the request of the authorised user, periodically during normal operation and before exiting the secure state according to FPT_FLS.1*[106] to demonstrate the correct operation of *parts of TSF*[107] .
FPT_TST.1.2	The TSF shall provide authorised users with the capability to verify the integrity of *TSF data*[108] .
FPT_TST.1.3	The TSF shall provide authorised users with the capability to verify the integrity of *TSF* ***implementation***[109] .

Application note 13: The security attribute "version number" of the UCP is part of the TSF data. During TSF testing, the consistency of the version number has to be checked to detect upgrades or attempted downgrades of the installed code of the TOE. In case of a detected change of the version number, the TOE must follow the UCP SFP and log the events according to FAU_GEN.1/SYS.

***ST application note 22*: The test of external entities (FPT_TEE.1) and the self-test (FPT_TST.1) is triggered externally by the ERS in regular intervals.**

[106] [selection (by PP author): *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]*]
[107] [selection (by PP author): *[assignment: parts of TSF], the TSF*]
[108] [selection (by PP author): *[assignment: parts of TSF data], TSF data*]
[109] [selection (by PP author): *[assignment: parts of TSF], TSF*]

6.1.5 Security Audit

FAU_GEN.1/SYS Audit data generation – System Log

Hierarchical to:	No other components.
Dependencies:	FPT_STM.1 Reliable time stamps
FAU_GEN.1.1/SYS	The TSF shall be able to generate an audit record of the following auditable events: a) start-up and shutdown of the audit functions; b) all auditable events for the *not specified*[110] level of audit; and c) other auditable events *(1) system operation commands as specified in TR-03151 [16], Appendix A* *(2) authentication failure handling (FIA_AFL.1): the reaching of the threshold for the unsuccessful authentication attempts with claimed Identity of the user,* *(3) failure with preservation of secure state (FPT_FLS.1): entering and exiting secure state,* *(4) setting of the version number of the UCP and upgrade of stored data,* *(5) the events selftest, exitSecureState, enterSecureState, configureLogging, startAudit, updateDeviceCompleted, authenticateSmaersAdmin, registerClient, deregisterClient, and deleteStoredData as specified by Klarstellungen und Anwendungshinweise zu BSI TR-03153 und BSI-CC-PP-0105-V2-2020 [10]* [111] [112]
FAU_GEN.1.2/SYS	The TSF shall record within each audit record at least the following information: a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, *no other audit relevant information*[113] .

[110] [selection (by PP author): *choose one of: minimum, basic, detailed, not specified*]
[111] [assignment (by ST author): *additional specifically defined auditable events*]
[112] [assignment (by PP author): *other specifically defined auditable events*]
[113] [assignment (by ST author): *other audit relevant information*]

Application note 14: The security relevant events that have to be logged according to FAU_GEN.1/SYS are part of the system log.

FMT_MTD.1/SYSCTSS Management of TSF data – System log – CTSS Interface Component

Hierarchical to:	No other components.
Dependencies:	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MTD.1.1/SYSCTSS	The TSF shall restrict the ability to (1) manual export, (2) clear after manual export, the *system logs*[114] to *CTSS Interface Component*[115] .

***ST application note 23*: fiskaly GmbH additionally restricts the ability to clear after manual export to TR administrator. Note that TR 03153 [8] already restricts this to TR administrator, so this added restriction harmonizes the role understandings between this ST and the TR requirements. In addition, the role TR administrator alway implies the role CTSS Interface, which makes this restriction a refinement.**

FMT_MTD.1/SYSAdmin Management of TSF data – System log – Administrator

Hierarchical to:	No other components.
Dependencies:	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MTD.1.1/SYSAdmin	The TSF shall restrict the ability to *(1) select audited events in FAU_GEN.1/SYS* *(2) define the number of audit records causing automatic export and clearing of exported audit records according to FAU_STG.3.1/SYS clause (1)* *(3) define the percentage of storage capacity of audit records if actions are assigned in FAU_STG.3.1/SYS clause (2)*[116] the *system logs*[117] to *Administrator*[118] .

***ST application note 24*: The TOE does not allow to deselect audit events by any user, In addition, it does not offer other configuration options with respect**

[114] [assignment (by PP author): *list of TSF data*]
[115] [assignment (by PP author): *the authorised identified roles*]
[116] [selection (by PP author): *change_default, query, modify, delete, clear,[assignment: other operations]*]
[117] [assignment (by PP author): *list of TSF data*]
[118] [assignment (by PP author): *the authorised identified roles*]

to FMT_MTD.1/SYSAdmin, which implicitly restricts the available configuration options (namely none) to Administrator.

FAU_STG.1/SYS Protected audit trail storage – System log

Hierarchical to:	No other components.
Dependencies:	FAU_GEN.1 Audit data generation
FAU_STG.1.1/SYS	The TSF shall protect the stored audit records in the audit trail from unauthorised deletion.
FAU_STG.1.2/SYS	The TSF shall be able to *prevent*[119] unauthorised modifications to the stored audit records in the audit trail.

FAU_STG.3/SYS Action in Case of Possible Audit Data Loss – System log

Hierarchical to:	No other components.
Dependencies:	FAU_STG.1 Protected audit trail storage
FAU_STG.3.1/SYS	The TSF shall (1) *automatically export audit trails and clear automatically exported audit records*[120] if the audit trail exceeds an *Administrator defined number of audit records within the pre-defined range [1, 1]*[121] [122] (2) *no action*[123] **if the audit trail exceeds an *Administrator settable percentage of storage capacity.*[124]**

Application note 15: The ST writer shall perform the open operations in FAU_STG.3.1/SYS 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 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).

***ST author comment to application note 15*: Auditable events of the TOE are as soon as possible signed by the CSP and the corresponding system log message is exported to the storage in accordance with FDP_ACF.1/LM and FDP_ETC.2/LM. This is equivalent to audit logs as defined in FDP_ACF.1.2/LM (3), which are also exported to the storage directly.**

[119] [selection (by PP author): *choose one of: prevent, detect*]
[120] [assignment (by PP author): *actions to be taken in case of possible audit storage failure*]
[121] [assignment (by ST author): *pre-defined range*]
[122] [assignment (by PP author): *pre-defined limit*]
[123] [assignment (by ST author): *actions to be taken in case of possible audit storage failure*]
[124] [assignment (by PP author): *pre-defined limit*]

Application note 16: The automatic export shall prevent loss of internal audit data due to storage constraints, by protecting the audit data and storing the signed and timestamped data in the CTSS interface component, i.e. outside the TOE.

***ST author comment to application note 16*: Loss of internal audit data is prevented by automatic exports carried out.**

6.1.6 Code Update Package Import

FDP_ACC.1/UCP Subset access control – Use of *Update Code Package*

Hierarchical to:	FDP_ACC.1 Subset access control
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1/UCP	The TSF shall enforce the *update SFP*[125] on • *subjects: CSP role;* • *objects: stored data;* • *operations: upgrade.*[126]

[125] [assignment (by PP author): *access control SFP*]
[126] [assignment (by PP author): *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

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

Hierarchical to:	FDP_ACC.1 Subset access control
Dependencies:	FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1/UCP	The TSF shall enforce the *Update SFP*[127] to objects based on the following: *(1) subjects: CSP role;* *(2) objects: update code package with security attributes version number.*[128]
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) CSP role is allowed to upgrade the stored data if* *(a) the digital signature of the UCP generated by the issuer is successfully verified by the SMAERS platform.*[129]
FDP_ACF.1.3/UCP	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *no rules*[130] .
FDP_ACF.1.4/UCP	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *(1) a CSP role is not allowed to upgrade the stored data if the verification of digital signature of the UCP by means of the SMAERS platform fails;* *(2) no other rules*[131] [132]

Application note 17: The CSP role should be allowed to apply the stored update code package if the version number of the update code package is higher than the version number of the TSF. The execution of UCP is outside the TSF-mediated functionality of the PP on hand.

[127] [assignment (by PP author): *access control SFP*]
[128] [assignment (by PP author): *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*]
[129] [assignment (by PP author): *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
[130] [assignment (by ST author): *rules, based on security attributes, that explicitly authorise access of subjects to objects*]
[131] [assignment (by ST author): *other rules, based on security attributes, that explicitly deny access of subjects to objects*]
[132] [assignment (by PP author): *rules, based on security attributes, that explicitly deny access of subjects to objects*]

FDP_ETC.2/UCP_UD Export 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]
FDP_ETC.2.1/UCP_UD	The TSF shall enforce the *log message SFP*[133] when exporting user data, controlled under the SFP(s), ~~outside of the TOE~~ **to the storage of the platform**.
FDP_ETC.2.2/UCP_UD	The TSF shall export the user data with the user data's associated security attributes.
FDP_ETC.2.3/UCP_UD	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/UCP_UD	The TSF shall enforce the following rules when user data is exported from the TOE: *no additional exportation control rules.*[134]

FDP_ITC.2/UCP_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/UCP_UD	The TSF shall enforce the *log message SFP*[135] when importing user data, controlled under the SFP, from ~~outside of the TOE~~ **the storage of the platform**.
FDP_ITC.2.2/UCP_UD	The TSF shall use the security attributes associated with the imported user data.
FDP_ITC.2.3/UCP_UD	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received. are unambiguously associated with the exported user data.
FDP_ITC.2.4/UCP_UD	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_UD	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: *no additional importation control rules.*[136]

[133] [assignment (by PP author): *access control SFP*]
[134] [assignment (by ST author): *additional exportation control rules*]
[135] [assignment (by PP author): *access control SFP(s) and/or information flow control SFP(s)*]
[136] [assignment (by ST author): *additional importation control rules*]

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 successful upgrade of the stored data***[137] the following objects: *previous code and data*[138] .

6.1.7 Trusted channel between TOE and CSP

FTP_ITC.1/TC Inter-TSF trusted channel

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FTP_ITC.1.1/TC	The TSF shall provide a communication channel between itself and ~~another trusted IT product~~ **the CSP** that is ~~logically distinct from other communication channels~~ *logically distinct from other communication channels*[139] and provides assured identification of its end points **TOE and CSP** and protection of the channel data from modification ~~or disclosure~~.
FTP_ITC.1.2/TC	The TSF shall permit the *TSF*[140] to initiate communication via the trusted channel.
FTP_ITC.1.3/TC	The TSF shall initiate communication via the trusted channel *for communication with the CSP*[141] .

Application note 18: Protection against modification is required for the trusted channel. If sensitive data is transferred over the trusted channel, the ST writer shall provide additional cryptographic operations to protect the exchanged data against disclosure.

***ST author comment to application note 18*: In addition to the protection against modification, the trusted channel established between the TOE and the CSP protects the exchanged data against disclosure.**

[137] [selection (by PP author): *allocation of the resource to, deallocation of the resource from*]
[138] [assignment (by PP author): *list of objects*]
[139] [selection (by ST author): *logically distinct from other communication channels, using physical separated ports*]
[140] [selection (by PP author): *the TSF, the remote trusted IT product*]
[141] [assignment (by PP author): *list of functions for which a trusted channel is required*]

FIA_UAU.5/TC Multiple authentication mechanisms

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_UAU.5.1/TC	The TSF shall provide *(1) PACE with Generic Mapping with user in PCD role with establishment of trusted channel according to FTP_ITC.1/TC,* *(2) no other method of mutual authentication with key establishment*[142] , *(3) message authentication by MAC verification of received messages*[143] to support user authentication.
FIA_UAU.5.2/TC	The TSF shall authenticate any user's claimed identity according to ~~the~~ *(1) PACE may be used for authentication of CSP with establishment of trusted channel according to FTP_ITC.1/TC,* *(2) message authentication by MAC verification of received messages shall be used after initial authentication of a remote entity according to clause (1) for trusted channel according to FTP_ITC.1/TC.*[144]

Application note 19: The ST writer may assign another method of mutual authentication with key establishment in FIA_UAU.5.1/TC clause (2) if this method is supported by the certified CSP and therefore meets the OSP.SecCryM "Secure Cryptographic Mechanisms" in PP CSP [9], PP CSPLight [13].

***ST author comment to application note 19*: No other method of mutual authentication with key establishment is assigned in FIA_UAU.5.1/TC.**

[142] [selection (by ST author): *other method of mutual authentication with key establishment*]
[143] [assignment (by PP author): *list of multiple authentication mechanisms*]
[144] [assignment (by PP author): *rules describing how the multiple authentication mechanisms provide authentication*]

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

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_API.1.1	The TSF shall provide a *PACE in PCD role*[145] to prove the identity of the *TOE*[146] to ~~an external entity~~ **a CSP and establishing a trusted channel according to FTP_ITC.1/TC**.

FCS_CKM.1 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	The TSF shall generate cryptographic keys for **FCS_COP.1** in accordance with a specified cryptographic generation algorithm *PACE with Curve P-256 (FIPS 186-4) [24]*[147] *and Generic Mapping in PCD role*[148] and specified cryptographic key sizes *256 bits*[149] that meet the following: *ICAO [22], section 4.4*[150] .

Application note 20: PACE is used to authenticate the TOE and the CSP. It establishes a trusted channel with MAC integrity protection of the following communication through the trusted channel.

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 *by overwriting the memory data with logical zeros (zeroization), random data or a new key value*[151] that meets the following: *none*[152] .

[145] [assignment (by PP author): *authentication mechanism*]
[146] [assignment (by PP author): *object, authorized user or role*]
[147] [selection (by ST author): *elliptic curves in table 5 (of [15])*]
[148] [assignment (by PP author): *cryptographic key generation algorithm*]
[149] [assignment (by PP author): *cryptographic key sizes*]
[150] [assignment (by PP author): *list of standards*]
[151] [assignment (by ST author): *cryptographic key destruction method*]
[152] [assignment (by ST author): *list of standards*]

FCS_COP.1 Cryptographic Operation

Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1	The TSF shall perform *MAC calculation and MAC verification*[153] in accordance with a specified cryptographic algorithm *according to AES-256 in CMAC [25]*[154] [155] and cryptographic key sizes *256 bits*[156] that meet the following: *the referenced standards above according to the chosen selection.*[157]

FCS_RNG.1 Random Number Generation

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FCS_RNG.1.1	The TSF shall provide a *deterministic*[158] random number generator that implements: *(DRG.3.1) If initialized with a random seed, the internal state of the RNG shall have at least 125 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.* [159]
FCS_RNG.1.2	The TSF shall provide random numbers that meet *(DRG.3.4) The RNG gets initialized with a random seed during intialisation of the TOE and persists the internal state of the RNG. It generates output for which* $\mathbf{2^{14}}$ *strings of bit length 128 are mutually different with probability* $\mathbf{1 - 2^{-8}}$*. (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.* [160] .

Application note 21: The TOE may use internal source or external source or more than one source of randomness providing seeds of at least 125 bits entropy. The deterministic part of the RNG shall meet BSI TR3116-5 [4] and therefore of class DRG.3 or higher according to [3].

***ST application note 25*: The TOE is of software type, so it has to rely on its environment to receive proper seed-input. Therefore fiskaly GmbH decided to rely on (SMAERS) Administrator to provide a proper seed of the required quality in the personalization phase of the TOE.**

[153] [assignment (by PP author): *list of cryptographic operations*]
[154] [selection (by ST author): *CMAC [25], GMAC [26], HMAC [27]*]
[155] [assignment (by PP author): *cryptographic algorithm*]
[156] [assignment (by PP author): *cryptographic key sizes*]
[157] [assignment (by PP author): *list of standards*]
[158] [selection (by ST author): *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*]
[159] [assignment (by ST author): *list of security capabilities*]
[160] [assignment (by ST author): *a defined quality metric*]

6.2 Security Assurance Requirements

The PP SMAERS [15] requires the TOE to be evaluated to EAL2 augmented with ALC_LCD.1 (Developer-Defined Lifecycle Model) and ALC_CMS.3 (Implementation representation CM coverage), and with specific refinements on ALC_CMS.3, ADV_ARC.1 and ATE_IND.2 (cf. PP SMAERS [15]).

6.3 Security Requirements Rationale

6.3.1 Dependency Rationale

No additional SFRs were introduced in this Security Target. So this section is equivalent to the corresponding section in [15] (including the package trusted channel, which is specified in chapter 7 of [15]).

6.3.2 Security Functional Requirements Rationale

This document introduces no additional SFRs, which are not present in the Protection Profile [15]. So this section is equivalent to the corresponding section in [15].

6.3.3 Security Assurance Requirements Rationale

This section is equivalent to the corresponding section of [15].

Chapter 7

TOE Summary Specification

The following section describes how the TOE meets each SFR. For that reason, the SFRs are assigned to Security Functions (SFs) provided by the TOE.

Keep in mind that the TOE is a software. It comes in the form of one executable binary, which starts a service upon execution. This service uses a Remote Procedure Call Framework (RPC), which is available for various programming languages, so the interface can be easily implemented by ERS vendors / integrators. The mapping from SRFs to SFs is shown in Table 7.1.

	SF_SecMan	SF_IdAuth	SF_UserDataProt	SF_TSFProt	SF_Audit	SF_UcpImp	SF_TrustChan
FMT_SMR.1	×						
FMT_SMF.1	×						
FMT_MOF.1	×						
FMT_MSA.1	×						
FMT_MSA.3	×						
FIA_ATD.1		×					
FMT_MTD.1/AD		×					
FMT_MTD.3/PW		×					
FIA_AFL.1		×					
FIA_USB.1		×					
FIA_UID.1		×					
FIA_UAU.1		×					

FIA_UAU.5		×					
FIA_UAU.6		×					
FDP_ACC.1/LM			×				
FDP_ACF.1/LM			×				
FDP_ITC.2/TD			×				
FDP_ETC.2/DTBS			×				
FDP_ITC.2/TSS			×				
FDP_ETC.2/LM			×				
FPT_TDC.1			×				
FMT_MSA.2			×				
FMT_MSA.4			×				
FPT_FLS.1				×			
FPT_TEE.1				×			
FPT_TST.1				×			
FAU_GEN.1/SYS					×		
FMT_MTD.1/SYSCTSS					×		
FMT_MTD.1/SYSAdmin					×		
FAU_STG.1/SYS					×		
FAU_STG.3/SYS					×		
FDP_ACC.1/UCP						×	
FDP_ACF.1/UCP						×	
FDP_ETC.2/UCP_UD						×	
FDP_ITC.2/UCP_UD						×	
FDP_RIP.1/UCP						×	
FTP_ITC.1/TC							×
FIA_UAU.5/TC							×
FIA_API.1							×
FCS_CKM.1							×
FCS_CKM.4							×
FCS_COP.1							×
FCS_RNG.1							×

Table 7.1: Mapping of security functional requirements (SFR) to security functions (SF) of the TOE

7.1 SF_SecMan Security Management

The TOE maintains several roles of entities. The roles are *unidentified user*, *administrator*, *CTSS interface*, *TR administrator*, and *CSP*. These roles are associated with entities after successful authentication. Based on the active role of an entity, the TOE offers a set of possible operations associated with that role, and prevents those actions that must be only available to other roles. This part of SF_SecMan satisfies FMT_SMR.1

The security management of the TOE maintains roles as follows:

- the role *administrator* is assigned to users after successful password authentication,
- the role *CTSS interface* is assigned to entities providing a clientID that is in the set of accepted clientIDs,
- the role *TR administrator* is assigned to entities providing with role CTSS interface, which authenticate by password or puk,
- the role *CSP* is assigned to CSPLight after successful trusted channel establishment.

The TR administrator role may

- dynamically add or remove clientIDs (of electronic record-keeping systems)
- deactivate the CTSS interface to end the TOE's life cycle

Whenever an (authenticated) client reports a *StartTransaction* operation, the TOE increments the internal transaction counter by 1. In case of *UpdateTransaction* and *FinishTransaction*, the transaction number is imported with the transaction data provided by the (authenticated) client.

Whenever a code update is triggered, the TOE uses the CSP to protect the new version number against modification. The CSP role is used by the TOE to record the increase of the version number after UCP installation by a digital signature with timestamp and signature counter value.

This part of SF_SecMan satisfies FMT_MSA.1, FMT_SMF.1.1 (3).

The TOE securely manages the authentication reference data of the administrator using the security storage in its operational environment. This part of SF_SecMan satisfies FMT_SMF.1.1 (2), FMT_SMF.1.1 (3).

The administrator is not expected to perform any actions in the life time of the TOE. The role does not have any management options except the change of its own authentication reference value.

This part of SF_SecMan satisfies FMT_MOF.1, FMT_SMF.1.1 (1).

The TOE uses restrictive default values for security attributes (for example per default all audit events are logged) and has no means to change these default values. This part of SF_SecMan satisfies FMT_MSA.3.

7.2 SF_IdAuth User Identification and Authentication

The TOE stores persistently and securely a list of supported clientIDs and the PACE Password used in the identification of the CSPLight, besides the hashed password of the administrator. This part of SF_IdAuth satisfies FIA_ATD.1.

The TOE supports the authentication of administrators by password. An authenticated administrator may modify the administrator password.

The TOE uses a password, which is stored as salted hash and failed authentication tries, stored in the secure storage in its operational environment to authenticate and manage the administrator. This part of SF_IdAuth satisfies FMT_MTD.1/AD, FMT_MTD.3/PW, FIA_AFL.1, FIA_USB.1.3 (3), FIA_UAU.5.2 (1).

The TOE authenticates clients (electronic record-keeping systems) by their provided clientID, that must be in the list of accepted clientIDs (which is dynamically settable). This list of accepted clientIDs is persisted in the secure storage in the operational environment of the TOE. This part of SF_IdAuth satisfies FIA_USB.1.3 (1), FIA_UAU.5.2 (2). To do so, the TOE manages a list of (at most 200) acceptable clientIds.

The TOE authenticates the CSP via trusted channel establishment. After successful trusted channel establishment, the TOE tests the CSP further. The TOE allows signature creation by the CSP only after this successful testing of the CSP. This part of SF_IdAuth satisfies FIA_USB.1.3 (2), FIA_UAU.5.2 (2).

The TOE associates users with roles as the result of successful authentication, whereas the initial role of all users is *unidentified user*. This part of SF_IdAuth satisfies FIA_USB.1.1, FIA_USB.1.1. The TOE allows the execution of self-tests even before a user is identified. This part of SF_IdAuth satisfies FIA_UID.1. The TOE allows the execution of self-tests even before a user is authenticated. Any other action requires authentication. This part of SF_IdAuth satisfies FIA_UAU.1.

Any authentication expires as soon as the TOE execution is terminated. This part of SF_IdAuth satisfies FIA_UAU.6.

7.3 SF_UserDataProt User Data Protection

User data refers to data imported to, and exported from the TOE, as well as data handled by the TOE internally in the meantime. This includes, for example, transaction data sent by the client (electronic record-keeping system) and audit records sent by the CSP.

The TOE is used by the ERS for

- importing transaction data, which immediately results in a transaction log signature creation request to the CSP, then to the import of the protocol data (which includes timestamp, signature, signature counter) from the CSP, and finally to the export of a signed transaction log message to the storage.
- commands (e.g., triggering a self-test), which results in a system log signature creation request to the CSP, then to the import of import of the signed system log, and finally to the export of the signed system log message to the storage.

Audit records of the CSP are imported by the TOE and exported to the storage as signed audit logs. No other role or entity may import audit records and export audit logs.

This part of SF_UserDataProt satisfies FDP_ACC.1/LM, FDP_ACF.1/LM.

Transaction data is only imported by the TOE if

- the clientID has been previously registered (i.e. it is in the set of acceptable clientIDs),
- the 'type of operation' property is present,
- the transaction number is in the set of ongoing transaction (in case of *UpdateTransaction* and *FinishTransaction* operations).

Otherwise an error is reported back to the ERS. This part of SF_UserDataProt satisfies FDP_ITC.2/TD (with the exception of FDP_ITC.2.5/TD (4)).

Based on the transaction data provided by the ERS, the TOE assembles the *data-to-be-signed* and exports it to the CSP. In addition, the TOE specifies the keyID (which is constant for each TOE) of the signature creation key to be used for the cryptographic signing operation. This part of SF_UserDataProt satisfies FDP_ETC.2/DTBS, FDP_ITC.2.5/TD (4).

The TOE imports signed log messages from the CSPLight, including signature value and timestamp. Audit records of the CSPLight are imported as signed audit logs. This part of SF_UserDataProt satisfies FDP_ITC.2/TSS.

Log messages are exported with the contents and in the structure as specified by BSI TR-03151 [16]. Audit records imported from the CSP are not altered by the TOE. This part of SF_UserDataProt satisfies FDP_ETC.2/LM, FPT_TDC.1.

Transaction numbers generated by the TOE build a strongly increasing sequence without gaps in normal operation. In normal operation, time stamps build a non-decreasing sequence, except possibly for time adjustments due to clock updates in the CSP. This part of SF_UserDataProt satisfies FMT_MSA.2.

The TOE increments the transaction number by 1 in case of *StartTransaction* and assigns this number to the list of ongoing transactions and to the imported transaction data. In case of *UpdateTransaction* and *FinishTransaction* operations, and if the imported transaction number is in the list of ongoing transactions, the TOE uses the imported transaction number for the creation of the new transaction log message. In case of *FinishTransaction*, the TOE removes

the transaction number from the list of ongoing transactions. This part of SF_UserDataProt satisfies FMT_MSA.4.

7.4 SF_TSFProt Protection of the TSF

The TOE implements self-test capabilities to ensure a consistent internal state by detecting unauthorized modifications of its protected data. The self-test includes the prevention of version downgrades of the TOE. The self-test is executed automatically at start-up of the TOE and during its normal operation, but can also be manually triggered. If the self-test fails, the TOE maintains its secure state, which means that all further operations that would change security attributes are rejected. The TOE leaves this state only after successfully passing the self-test.

In addition, the TOE uses snapshots, signed and verified by the CSPL, to protect internal data, like the transaction counter, against unwanted modification. The verification of these snapshots is part of the self-test of the TOE. The TOE makes here additional use of the cryptographic capabilities of the CSPL.

This part of SF_TSFProt satisfies FPT_FLS.1, FPT_TST.1.1.

In addition, the TOE tests external entities:

- The TOE tests the identity and availability of clients (electronic record-keeping systems). For this purpose, the clients of the TOE are required to regularly trigger the TOE's function to identify the external components it is connected to (cf. FPT_TEE.1).
- The TOE tests the identity and availability of the CSP. In addition, the TOE tests the plausibility of timestamps and signature counter values provided by the CSP. The plausibility of timestamp values is considered as given if the points in time when the transaction starts are monotonically increasing, or there exists an audit log message that reports a time update carried out by the CSP that explains any inconsistency. The plausibility of signature counter values is considered as given if the values are increasing strongly monotonically.

This part of SF_TSFProt satisfies FPT_TEE.1.

While the purpose of testing of the clients is mainly to keep a record (in the form of system logs) of their availability, the test of the CSP may result in the detection of problems that cause a failure, leading to the TOE entering a secure state. In this state, the TOE will no longer accept any transaction data nor create signed log messages. This part of SF_TSFProt satisfies FPT_FLS.1.

The TOE checks its internal state using the cryptographic services of the CSPLight to prevent operation in a corrupted state. The TOE prevents version downgrades by comparing the version number of the UCP (attribute of the TOE binary) with the current version number of the TOE (attribute in the stored data) during self-test at start-up. If the version number of the UCP is smaller than the version number of the TOE, the self-test fails. This part of SF_TSFProt satisfies FPT_TST.1.2.

The integrity of the TOE is verified by the platform. This part of SF_TSFProt satisfies FPT_TST.1.3.

7.5 SF_Audit Security Audit

The TOE generates audit records for auditable events regarding system operation commands as specified in BSI TR-03151 [16], failures with preservation of secure state (cf. Section 7.4) and version upgrades triggered by importing an authentic Update Code Package (UCP). For each auditable event, the TOE records the date and time, the type of event and the outcome of the event and the identity of the involved entity. This part of SF_Audit satisfies FAU_GEN.1/SYS.

Audit records generated by the TOE are as soon as possible sent to the CSP for signature creation and exported as signed system logs to the storage. Since system logs are only generated in association with imported commands, the TOE exports as soon as possible the signed system log as a result of the command to the storage. This part of SF_Audit satisfies FMT_MTD.1/SYSCTSS, FAU_STG.1/SYS, FAU_STG.3/SYS.

The TOE does not provide an option for the administrator to select which events are actually audited. All logs are always created. Since each system log is exported as soon as possible as a result of an imported command, defining the number of audit records causing an export (always just one in accordance with FAU_STG.3/SYS (1)) and storage limits (*no action* assigned in FAU_STG.3/SYS (2)) is not supported by the TOE. This part of SF_Audit satisfies FMT_MTD.1/SYSAdmin.

7.6 SF_UcpImp Code Update Package Import

An authentic Update Code Package (UCP) must be used for an update of the software (i.e., with a version number equal to or greater than the currently installed TOE sample) via the platform of the TOE.

Supported platform is one variant of *GNU/Linux*, provided in Table 1.1. The TOE is dependent on the authenticity verification provided by this platform. For Linux, signed RPM or DEB packages are used. The CSPLight is used for securely recording the version upgrade by a digital signature with timestamp and signature counter. This part of SF_UcpImp satisfies FDP_ACC.1/UCP, FDP_ACF.1/UCP.

In addition the TOE compares checks the version number of the binary and uses a service of the platform to verify the integrity of the implementation of the TOE. This is done each time the selftest gets executed.

The TOE makes use of a secure storage mechanism for its user data and TSF data. Export occurs regularly during normal operation. Import occurs at least at every start-up of the TOE including start-ups after updates. This part of SF_UcpImp satisfies FDP_ETC.2/UCP_UD, FDP_ITC.2/UCP_UD.

7.7 SF_TrustChan Trusted channel between TOE and CSPLight

The TOE and the CSPLight communicate via a trusted channel established with Password Authenticated Connection Establishment (PACE). The trusted channel protects the integrity and authenticity of exchanged messages by Message Authentication Code and the confidentiality of the transmitted data.

The trusted channel uses HTTP(S) as non-TOE transport protocol for the connection to the CSPLight. All exchanged messages are encrypted and integrity protected using PACE. This part of SF_TrustChan satisfies FTP_ITC.1/TC.

The TOE establishes the trusted channel in PCD role with the CSPLight using NIST Curve P-256 (cf. [24]). No other means for trusted channel establishment are supported by the TOE. MAC verification is performed for each exchanged message with the CSPLight after trusted channel establishment (if successful) followed by decryption using AES-256 in CMAC. This part of SF_TrustChan satisfies FIA_UAU.5/TC, FIA_API.1, FCS_CKM.1, FCS_COP.1.

The session keys used for encryption/decryption and MAC are destroyed once a session expires or the TOE is shut down. This part of SF_TrustChan satisfies FCS_CKM.4.

The key derivation used for establishing the trusted channel uses a random number generator (RNG) of class DRG.3 provided by the TOE. This RNG is based on an SHA-256 generator as defined in chapter 10.1.1 of [23] (FCS_RNG.1). This part of SF_TrustChan satisfies FCS_RNG.1.

The TOE will overwrite all cryptographic key material with zeros as soon as the material is no more needed. This part of SF_TrustChan satisfies FCS_CKM.4.

Bibliography

[1] 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).

[2] Verordnung zur Bestimmung der technischen Anforderungen an elektronische Aufzeichnungs- und Sicherungssysteme im Geschäftsverkehr (Kassensicherungsverordnung – KassenSichV), Bundesgesetzblatt Jahrgang 2017 Teil I Nr. 66, ausgegeben zu Bonn am 6. Oktober 2017. https://www.bundesfinanzministerium.de/Content/DE/Gesetzestexte/Gesetze_Verordnungen/2017-10-06-KassenSichV.html.

[3] BSI. A proposal for: Functionality classes for random number generators, Version 2.0.

[4] Bundesamt für Sicherheit in der Informationstechnik. Technische Richtlinie BSI TR-03116 Kryptographische Vorgaben für Projekte der Bundesregierung Teil 5: Anwendungen der Secure Element API, Stand 2019, Datum: 1. Februar 2019. https://www.bsi.bund.de/DE/Publikationen/TechnischeRichtlinien/tr03116/index_htm.html.

[5] Common Criteria. Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, CCMB-2017-04-001, Version 3.1, Revision 5, April 2017. https://www.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R5.pdf.

[6] Common Criteria. Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, CCMB-2017-04-002, Version 3.1, Revision 5, April 2017. https://www.commoncriteriaportal.org/files/ccfiles/CCPART2V3.1R5.pdf.

[7] Common Criteria. Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components, CCMB-2017-04-003, Version 3.1, Revision 5, April 2017. https://www.commoncriteriaportal.org/files/ccfiles/CCPART3V3.1R5.pdf.

[8] Federal Office for Information Security. BSI TR-03153 Technische Sicherheitseinrichtung für elektronische Aufzeichnungssysteme Version 1.0.1. https://www.bsi.bund.de/DE/Publikationen/TechnischeRichtlinien/tr03153/tr03153_node.html.

[9] Federal Office for Information Security. Cryptographic Service Provider (CSP) Version 0.9.8, BSI-CC-PP-0104-2019. https://www.bsi.bund.de/SharedDocs/Zertifikate_CC/PP/aktuell/PP_0104.html.

[10] Federal Office for Information Security. Klarstellungen und Anwendungshinweise zu BSI TR-03153-TS und BSI-CC-PP-0105-V2-2020, 17. November 2020. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03153/TR-03153-TS-Ergaenzungen.pdf?__blob=publicationFile&v=2.

[11] Federal Office for Information Security. Protection Profile Configuration Cryptographic Service Provider – Time Stamp Service, Audit and Clustering (PPC-CSP-TS-Au-Cl) Version 0.9.4, BSI-CC-PP-0108-2019. https://www.bsi.bund.de/SharedDocs/Zertifikate_CC/PP/aktuell/PP_0108.html.

[12] Federal Office for Information Security. Protection Profile Cryptographic Service Provider Light - Time Stamp Service and Audit (PPC-CSPLight-TS-Au), Version 1.0. https://www.bsi.bund.de/SharedDocs/Zertifikate_CC/PP/aktuell/PP_0112.html.

[13] Federal Office for Information Security. Protection Profile Cryptographic Service Provider Light (CSPL) Version 1.0, BSI-CC-PP-0111-2019. https://www.bsi.bund.de/SharedDocs/Zertifikate_CC/PP/aktuell/PP_0111.html.

[14] Federal Office for Information Security. Protection Profile Cryptographic Service Provider Light Time Stamp Service and Audit - Clustering (PPC-CSPLight-TS-Au-Cl) Version 1.0, BSI-CC-PP-0113-2019. https://www.bsi.bund.de/SharedDocs/Zertifikate_CC/PP/aktuell/PP_0113.html.

[15] Federal Office for Information Security. Security Module Application for Electronic Record-keeping Systems (SMAERS) Version 1.0, BSI-CC-PP-0105-V2-2020. https://www.bsi.bund.de/SharedDocs/Zertifikate_CC/PP/aktuell/PP_0105_0105_V2.html.

[16] Federal Office for Information Security. Technical Guideline BSI TR-03151 Secure Element API (SE API) Version 1.0.1. https://www.bsi.bund.de/DE/Publikationen/TechnischeRichtlinien/tr03151/index_htm.html.

[17] fiskaly GmbH. Functional Specification – Security Module Application for Electronic Record-Keeping Systems, Version 1.1.5, 2021.

[18] fiskaly GmbH. Functional Specification, fiskaly Cloud Crypto Service Provider Version 1.2.10, 2021.

[19] fiskaly GmbH. Preparative Procedures & Operational User Guidance Documentation – fiskaly Security Module Application for Electronic Record-Keeping Systems, Version 1.2.0, 2021.

[20] fiskaly GmbH. Preparative Procedures & Operational User Guidance Documentation fiskaly Cloud Crypto Service Provider Version 1.3.3, 2021.

[21] fiskaly GmbH. Securit Target, fiskaly Cloud Crypto Service Provider Version 1.2.3, 2021.

[22] ICAO: Machine Readable Travel Documents. ICAO Doc9303, Part 11: Security Mechanisms for MRTDSs, seventh edition, 2015 .

[23] NIST.] Recommendation for Random Number Generation Using Deterministic Random Bit Generators, NIST 800-90A Revision 1, https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-90a.pdf.

[24] NIST. Digital Signature Standard (DSS), 2013.

[25] NIST. SP800-38B Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, May 2005 .

[26] NIST. SP800-38D Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007.

[27] NIST. The Keyed-Hash Message Authentication Code (HMAC), July 2008.

Keywords

Term	Description
authentication reference data	data used by the TOE to verify the authentication attempt of a user
authentication verification data	data used by the user to authenticate themselves to the TOE
authenticity	the property that ensures that the identity of a subject or resource is the one claimed (cf. ISO/IEC 7498-2:1989)
cluster	a system of TOE samples initialized by an administrator and communication through trusted channels in order to manage known users and to share the cryptographic keys
cryptographic key	a variable parameter which is used in a cryptographic algorithm or protocol
data integrity	the property that data has not been altered or destroyed in an unauthorized manner (cf. ISO/IEC 7498-2:1989)
firmware	executable code that is stored in hardware and cannot be dynamically written or modified during execution while operating on a non-modifiable or limited execution platform, cf. ISO/IEC 19790
hardware	physical equipment or comprises the physical components used to process programs and data or to protect physically the processing components, cf. ISO/IEC 19790
Issuer of update code package	Trusted authority issuing an update code package (UCP) and holding the signature private key for signing the UCP and corresponding to the public key implemented in the TOE for verification of the UCP. The issuer is typically the TOE manufacturer. The issuer of an UCP is identified by the security attribute Issuer of the UCP.
Platform guidance	All documentation provided by the hardware manufacturer, or software platform manufacturer, that provides information on how to securely implement functionality.

private key	confidential key used for asymmetric cryptographic mechanisms like decryption of cipher text, signature-creation or authentication proof, where it is difficult for the adversary to derive the confidential private key from the known public key
public key	public known used for asymmetric cryptographic mechanisms like encryption of cipher text, signature-verification or authentication verification, where it is difficult for the adversary to derive the confidential private key from the known public key
secret key	key of symmetric cryptographic mechanisms, using two identical keys with the same secret value or two different values, where one may be easy calculated from the other one, for complementary operations like encryption / decryption, signature-creation / signature-verification, or authentication proof / authentication verification
secure channel	a trusted channel which is physically protected and logical separated communication channel between the TOE and the user, or is protected by means of cryptographic mechanisms
software	executable code that is stored on erasable media which can be dynamically written and modified during execution while operating on a modifiable execution platform, cf. ISO/IEC 19790
trusted channel	a means by which a TSF and another trusted IT product can communicate with necessary confidence (cf. CC part 1 [5], paragraph 97)
update code package	code if implemented changing the TOE implementation at the end of the TOE life time

Table 7.2: Glossary (Table 8 in Base-PP [13])

Abbreviations

Acronym	**Term**
A.xxx	Assumption
CC	Common Criteria
CSP	Cryptographic Service Provider
CSPLight	Cryptographic Service Provider Light
ECC	Elliptic curve cryptography
HMAC	Keyed-Hash Message Authentication Code
KDF	Key derivation function
MAC	Message Authentication Code
n. a.	Not applicable
O.xxx	Security objective for the TOE
OE.xxx	Security objective for the TOE environment
OSP.xxx	Organisational security policy
PACE	Password Authenticated Connection Establishment
PKI	Public key infrastructure
PP	Protection profile
SAR	Security assurance requirements
SFR	Security functional requirement
T.xxx	Threat
TOE	Target of Evaluation
TSF	TOE security functionality
UCP	update code package

Table 7.3: Abbreviations (Table 9 in Base-PP [13])

Code and Document Signing

For verifying signed artifacts, the following public key shall be used:

```
RWRUupxsFLBrtFlb7g2ZWVFSQE23BvMEtPszqNu2E8Q32U4L99AKexpl
```

fiskaly GmbH uses an `Ed25519` key following OpenBSD's `signify` [1] approach for digitally signing all published artifacts (i.e., software and documents). In particular, fiskaly GmbH uses the `minisign` [2] tool that is fully compatible with the `signify` verification tool. As an example, the following command can be used for verification:

```
$ minisign -P {PUBKEY} -V -m {FILENAME}
Signature and comment signature verified
Trusted comment: timestamp:1603971010 file:{FILENAME}
```

[1] `https://www.openbsd.org/papers/bsdcan-signify.html`
[2] `https://jedisct1.github.io/minisign/`