cryptovision CSP – Java Card applet providing Cryptographic Service Provider

Security Target Lite

BSI-DSZ-CC-1119

Common Criteria / ISO 15408
EAL 4+

Document Version 1.16 • 2022-11-21

cv cryptovision GmbH • Munscheidstr. 14 • 45886 Gelsenkirchen • Germany
www.cryptovision.com • info@cryptovision.com • +49-209-167-2450

Content

Version Control

Version	Date	Author	Changes to Previous Version
1.16	2022-11-21	Thomas Zeggel	Security Target Lite based on version 1.16 of the Security Target.

1 Introduction

1.1 ST/TOE Identification

Title:	cryptovision CSP – Java Card applet providing Cryptographic Service Provider – Security Target Lite
Document Version:	v1.16
Origin:	cv cryptovision GmbH
Compliant to:	Common Criteria Protection Profile – Cryptographic Service Provider, BSI-CC-PP-0104-2019, in the configuration "Cryptographic Service Provider – Time Stamp Service and Audit (PPC-CSP-TS-Au)" according to PP-module [PP0107]
Product identification:	cryptovision CSP – Java Card applet providing Cryptographic Service Provider, version 2.0
Short TOE name:	cryptovision CSP
Javacard OS platform:	NXP JCOP 4.7 SE051, NSCIB-CC-0095534, [Zert_OS]
Security controller:	NXP N7121, BSI-DSZ-CC-1136, [Zert_IC]
TOE documentation:	Administration and user guide ([Guidance_PRE], [Guidance_OPE])

1.2 ST overview

This document contains the security target for the product cryptovision CSP – Java Card applet providing Cryptographic Service Provider to be used exclusively on the NXP JCOP 4.7 SE051 Javacard OS platform, which is certified according to CC EAL 6+ [ZertOS].

The product cryptovision CSP as well as the JCOP 4.7 operating system are provided on a smart card chip based on the NXP N7121 security controller, which is itself certified according to CC EAL 6+ [ZertIC].

This Security Target defines the security objectives and requirements for the cryptovision CSP.

This security target claims strict conformance to the Protection Profile *Common Criteria Protection Profile – Cryptographic Service Provider*, BSI-CC-PP-0104-2019 [PP0104], in the configuration "Cryptographic Service Provider – Time Stamp Service and Audit (PPC-CSP-TS-Au)" according to PP-module [PP0107].

The main objectives of this ST are:

- to introduce TOE and the CSP application,
- to define the scope of the TOE and its security features,
- to describe the security environment of the TOE, including the assets to be protected and the threats to be countered by the TOE and its environment during the product development, production and usage,
- to describe the security objectives of the TOE and its environment supporting in terms of integrity and confidentiality of application data and programs and of protection of the TOE,
- to specify the security requirements which includes the TOE security functional requirements, the TOE assurance requirements and TOE security functionalities.

The assurance level for the TOE is CC EAL4 augmented with ALC_DVS.2 and AVA_VAN.5.

1.3 TOE overview

The TOE overview follows the description in the protection profile [PP0104].

1.3.1 TOE type

The Target of Evaluation (TOE) is a cryptographic service provider (CSP) component. The TOE is dedicated to provide cryptographic services for the protection of the confidentiality and the integrity of user data, and for entity authentication.

1.3.2 TOE definition

The TOE is physically defined as a device consisting of hardware, firmware and software. The TOE is implemented as a security integrated circuit based on a Java Card with the NXP JCOP 4.7 operating system. The cryptovision CSP applet layer consists of the CSD applet, the CSI applet and the SIF library.

It provides a Java Card interface for any application which is loaded to the chip. The CSP functionality can be used together with the basic JCOP 4.7 Java Card functionality by an application loaded on the CSP.

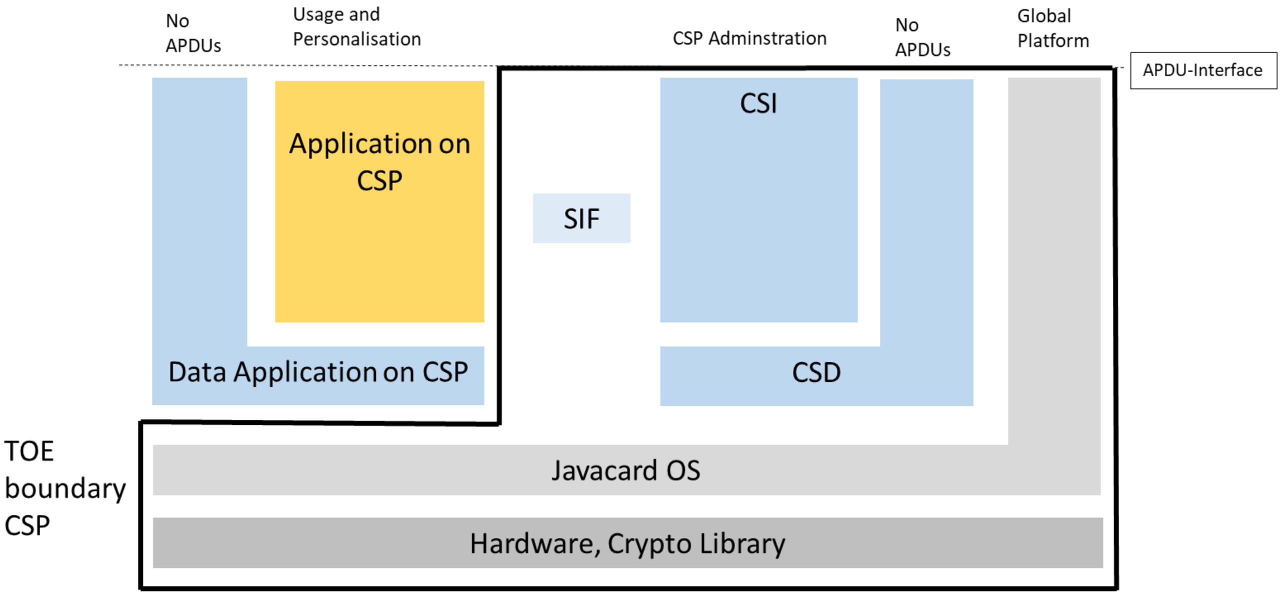


Figure 3: Structure of the TOE (CSP) and TOE boundary. The CSP applet layer consists of the components CSI, CSD and SFI.

While the CSI applet implements the main security functionality and the administration interface of the TOE, the CSD applet is responsible for the the secure storage of keys. Thus, an update process can exchange CSI without changing the key set of a CSP. The SIF library contains the shareable interfaces usable by an application residing on the CSP (e.g. a SMAERS application or an interface adaption application for master-slave usage of the CSP).

Note that the cryptovision CSP can also be used for the client-server architecture. In this case, a simple interface applet must be present as application which links external APDU calls to the SIF interface. This interface applet is not part of the TOE.

1.3.3 TOE security functionality

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

- authentication of users,

- authentication and attestation of the TOE to entities,
- data authentication and non-repudiation including time stamps,
- encryption and decryption of user data,
- trusted channel including mutual authentication of the communicating entities, encryption and message authentication proof for the sent data, decryption and message authentication verification for received data,
- management of cryptographic keys with security attributes including key generation, key derivation and key agreement, internal storage of keys, import and export of keys with protection of their confidentiality and integrity,
- generation of random bits which may be used for security services outside the TOE.

The TSF provides a non-cryptographic real time service.

The TOE uses memory encryption for protection of internally stored data.

The TOE is dedicated for composed IT products comprising the TOE and one or more application components. The TOE provides the security services for these application components.

The protection profile [PP0104] considers two different architecture of the composed IT product:

- Platform architecture: The TOE is a platform consisting of hardware and an operating system providing a secure execution environment and security services for the application component running on top.
- Client-server architecture: The TOE and the application component are physically separated components interacting through a trusted channel. The application component (in client role) uses the security services of the TOE (in server role).

The TOE targets **the platform architecture as well as the client-server architecture**. If the TOE is used in a client-server architecture, an interface applet must be present as application which links external APDU calls to the internal SIF interface. This interface applet is not part of the TOE.

The communication between the TOE and the application is protected by means of secure channel. A secure channel is a trusted channel (cf. for definition CC part 1 [CC_1], paragraph 97) which is physically protected and logical separated communication channel between the TOE and the user, or is protected by means of cryptographic mechanisms.

The TOE supports cryptographically protected trusted channel between the TOE and the external entities. This has to be used if the TOE is used in a client-server architecture. In case of usage of the TOE in a platform architecture the TOE protects the communication with the application physically and by logical separated communication channel. In this case, the communication between the TOE and the application is protected by the JCOP 4.7 platform (secure inter-applet communication).

The internal cryptographic TSF is used for

- TSF data import including certificates and cryptographic keys,
- confidentiality protection of stored user data and TSF data,

The non-cryptographic TSF provides human user authentication, access control on cryptographic TSF and cryptographic keys, security audit and TSF protection.

The TOE supports download, authenticity verification and decryption of Update Code Packages for the CSP.

The TOE provides a time service, time stamp service and security audit.

The time service allows the user to query the internal time of the TSF.

The time stamp service provides evidence that user data were presented to the TSF and exported audit data were generated at certain point in time and in a verifiable sequence. The validity of these user data and audit records can be verified.

The audit functionality generates audit records on selected user activities controlled by the TSF and security events of the TOE defined by the Base-PP [PP0104] and the PP-Module [PP0107]. The Administrator role may be split in an additional role Auditor and Timekeeper.

- The Auditor is allowed to configure the audit functionality, review audit data and export audit trails.
- The Timekeeper is allowed to adjust the internal clock.

Neither of those roles is allowed to manage cryptographic keys, users and update code packages.

1.3.4 Method of use

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

The TOE provides time service and time stamp service as additional method of use compared with those of the TOE defined in the Base-PPs. The time service provides users with reliable time as known to the TOE.

The time stamp service provides evidence some user data are provided to the TOE at given point in time. The security audit can be used to make the user responsible for their actions including those described in the Base-PP [PP0104]. The audit records can be exported in a signed and time stamped form.

1.3.5 TOE identification

The configuration of the SE051 is: 0x045A (cf. table 4.10 in [AGD_PRE]).

The module configuration is: 0x0815 (cf. table 5.3 in [AGD_PRE]).

Identification of the TOE platform, configuration and module configuration is performed according to [AGD_PRE].

Once the platform is identified correctly, the cryptovision CSP can be verified as descibed in [Guidance_PRE].

1.3.6 Major security features of the TOE

The TOE provides the following TOE security functionalities:

- TSF_Access manages the access to objects (files, directories, data and secrets) stored in the TOE. Access is granted (or denied) in accordance to access rights that depend on appropriate identification and authentication mechanisms.
- TSF_Admin manages the security functional policies as well as the timer and audit storage.
- TSF_Secret ensures secure management of secrets such as cryptographic keys. This covers secure key storage, access to keys as well as secure key deletion.
- TSF_Crypto performs high level cryptographic operations. The implementation is mainly based on the Security Functionalities provided by TSF_OS.
- TSF_SecureMessaging realizes a secure communication channel after successful authentication. Please note that SFRs of the FCS_COP group are realized within TSF_Crypto, even if they are used by TSF_SecureMessaging.
- TSF_Auth realizes different authentication mechanisms.

- TSF_Integrity protects the integrity of internal data. This function makes use of the underlying Java Card OS.
- TSF_OS contains all security functionalities provided by the certified platform (IC, Javacard operation system). Besides some minor additions, the cryptographic operations are provided by this platform.

1.4 TOE life cycle

The platform of the TOE (hardware, IC embedded software and the Java Card OS) has been developed by

- NXP (hardware, IC dedicated software, IC embedded software, Java Card OS).

The TOE comprises of the NXP SE051 product, which itself is a composite product based on the certified hardware, certified crypto library and the certified Java card operating system layer. The development and certification of this platform is in the hands of NXP.

Cryptovision uses the guidance documentation for relevant parts of the IC Dedicated Software and the IC Embedded Software (operating system) and develops the CSP application layer and the according guidance documentation.

The CSP application layer is a set of Java Card applets and a library (CSI, CSD, SIF) developed by cryptovision and adds the specific security functionality to fulfill the requirements of protection profile [PP0104] with [PPC-CSP-TS-Au].

After completion of the development, the CSP application layer is delivered from cryptovision to NXP (standard high volume production) or a third party (small volume production) in a secure way (encrypted and digitally signed). In the case of the standard high volume production, the CSP application layer is installed on the SE051 platform during the production at NXP. In this case, the delivery of the CSP chips by NXP is the delivery of the TOE according to Common Criteria. In the other case (small volume production), the delivery of the CSP application layer to the third party is the delivery of the TOE according to Common Criteria. After this, the CSP application layer is loaded to the SE051 following [AGD_PRE].

1.5 Production and delivery of the TOE

1.5.1 Standard high volume production

The CSP applet layer and the according guidance documentation are securely delivered to the IC manufacturer (NXP) who integrates CSP layer in the production software images (integrating the Java Card OS with the CSP layer).

The production at NXP includes also the integration of a set of cryptographic keys generated and owned by cryptovision.

The TOE is then produced and delivered to customers. The security measures during production and delivery process resemble the certified processes for the Java Card OS. The general guidance documentation of the Javacard OS is delivered by the NXP Docstore (the standard process for the certified Javacard OS). The CSP guideance documentation is delivered by cryptovision by encrypted and signed email.

Product-specific keys (attestation key, authentication keys if applicable) are generated at cryptovision and securely delivered to NXP by encrypted email together with the CSP code. They are stored on the TOE as part of the production process. Depending on the planned usage of the product, the production process may include a further software product (e.g., a SMAERS applet) and keys and certificates from bulk data that have been securely delivered by a PKI provider.

The TOE is securely delivered from NXP to a device manufacturer, using the process that is also used for standard Java Cards. **With this delivery the life cycle phase ends which is subject of CC evaluation according to the assurance life cycle (ALC).**

The device manufacturer embedds the CSP (optionally with SMAERS) chip in an end device (e.g., a USB token or a SD card). Afterwards, the TOE is delivered to the customer.

1.5.2 Small volume production

The chips with the Java Card OS are produced by NXP. The production at NXP includes also the integration of a set of cryptographic keys generated and owned by cryptovision.

The chip is then delivered to a third party. The CSP applet layer (CSI, CSD, SIF), the attestation key and the according guidance documentation are securely delivered to this third party (embedded in protected APDUs). **With this delivery the life cycle phase ends which is subject of CC evaluation according to the assurance life cycle (ALC).**

The application layer of the TOE (CSI, CSD, SIF) is loaded on the NXP SE051 chip in the third party environment (encrypted and digitally signed) using standard Global platform mechanisms with delegated management. The attestation key is loaded using the secure key import mechanism of the CSP.

Further steps may be added if the CSP is directly loaded with an application (like cryptovision SMAERS) at the third party. It should be noted that delivery of the CSP guidance documentation may only be necessary for the third party, since the end customer ususally only gets the CSP combined with an application.

2 Conformance claims

2.1 CC conformance claims

The security target claims conformance to CC version 3.1 revision 5.

Conformance of this security target with respect to CC Part 2 [CC_2] (security functional components) is CC Part 2 extended.

Conformance of this security target with respect to CC Part 3 [CC_3] (security assurance components) is CC Part 3 conformant.

2.2 Package claim

This security target claims package-augmented conformance to EAL4. The minimum assurance level for this protection profile is EAL4 augmented with AVA_VAN.5 and ALC_DVS.2.

2.3 PP claim

This security target claims strict conformance to

- Common Criteria Protection Profile – Cryptographic Service Provider, BSI-CC-PP-0104-2019 [PP0104],

in the configuration

- Cryptographic Service Provider – Time Stamp Service and Audit (PPC-CSP-TS-Au)

according to the protection profile module [PP0107]

2.4 Statement of Compatibility concerning Composite Security Target

2.4.1 Assessment of the Platform TSFs

The following table lists all Security Functionalities of the underlying Platform ST and shows, which Security Functionalities of the Platform ST are relevant for this Composite ST and which are irrelevant. The first column addresses specific Security Functionality of the underlying platform, which is assigned to Security Functionalities of the Composite ST in the second column. The last column provides additional information on the correspondence if necessary.

Platform TSF-group	Correspondence in this ST	References/Remarks
SF.JCVM	-	Java Card Virtual Machine
SF.CONFIG	-	Configuration Management
SF.OPEN	-	Card Content Management
SF.CRYPTO	TSF_Crypto	Cryptographic Functionality
SF.RNG	TSF_Crypto	Random Number Generator Part of TSF.Crypto
SF.DATA_STORAGE	TSF_Secret	Secure Data Storage
SF.PUF	-	User Data Protection using PUF PUF functionality is not used in the TOE
SF.EXT_MEM	-	External Memory Not used in the TOE.
SF.OM	-	Java Object Management
SF.MM	TSF_Secret	Memory Management
SF.PIN	TSF_Access	PIN Management
SF.PERS_MEM	-	Persistent Memory Management
SF.EDC	TSF_Integrity	Error Detection Code API
SF.HW_EXC	TSF_Integrity	Hardware Exception Handling
SF.RM	-	Restricted Mode
SF.PID	TSF_Admin	Platform Identification SF.PID provides a platform identifier. This platform identifier is generated during the card image generation. The platform identifier contains IDs for: • NVM content (stored during romizing) • Patch Level (stored during romizing, can be changed during personalization if patch is loaded) • ROM code (stored during romizing) • ROM code checksum (stored during romizing or during first TOE boot). It identifies unambiguously the NVM and ROM part of the TOE.
SF.SMG_NSC	TSF_Crypto, TSF_Secret	No Side-Channel
SF.ACC_SBX	-	Secure Box The functionality is not used for the TOE.
SF.MOD_INVOC	-	Module Invocation

SF.RENS_RES	-	Sensitive Result

Table 1: Relevant platform TSF-groups and their correspondence

2.4.2 Assessment of the Platform SFRs

The following table provides an assessment of all Platform SFRs. The Platform SFRs are listed in the order used within the security target of the platform [ST_JCOP].

Platform SFR	**Correspondence in this ST**	**References/Remarks**
COREG_LC Security Functional Requirements (chapter 7.2.1 in platform ST)		
Firewall Policy (chapter 7.2.1.1 in platform ST)		
FDP_ACC.2/FIREWALL	No correspondence	Out of scope (internal Java Card Firewall). The resulting requirements for applets are reflected in the User Guidance of the TOE. No contradiction to this ST.
FDP_ACF.1/FIREWALL	No correspondence	Out of scope (internal Java Card Firewall). The resulting requirements for applets are reflected in the User Guidance of the TOE. No contradiction to this ST.
FDP_IFC.1/JCVM	No correspondence	Out of scope (internal Java Virtual Machine). No contradiction to this ST.
FDP_IFF.1[JCVM]	No correspondence	Out of scope (internal Java Virtual Machine). No contradiction to this ST.
FDP_RIP.1/OBJECTS	No correspondence.	Out of scope (internal Java Card Firewall). No contradiction to this ST.
FMT_MSA.1/JCRE	No correspondence	Out of scope (internal Java Card Firewall). No contradiction to this ST.
FMT_MSA.1/JCVM	No correspondence	Out of scope (internal Java Card Firewall). No contradiction to this ST.
FMT_MSA.2/FIREWALL-JCVM	No correspondence	Out of scope (internal Java Card Firewall). The resulting requirements for applets are reflected in the User Guidance of the TOE. No contradiction to this ST.
FMT_MSA.3/FIREWALL	No correspondence	Out of scope (internal Java Card Firewall). The resulting requirements for

Platform SFR	Correspondence in this ST	References/Remarks
		applets are reflected in the User Guidance of the TOE. No contradiction to this ST.
FMT_MSA.3/JCVM	No correspondence	Out of scope (internal Java Card Firewall). No contradiction to this ST.
FMT_SMF.1	No correspondence	Out of scope (internal Java Card Firewall). No contradiction to this ST.
FMT_SMR.1	No correspondence	Out of scope (internal Java Card Firewall). No contradiction to this ST.
Application Programming Interface (chapter 7.2.1.2 in platform ST)		
FCS_CKM.1 (FCS_CKM.1.1, FCS_CKM.1.1[RSA][, FCS_CKM.1.1[ECDSA], FCS_CKM.1.1[PUF])	FCS_CKM.1/AES FCS_CKM.1/ECC FCS_CKM.1/RSA FCS_CKM.1/ECKA-EG FCS_CKM.1/AES_RSA FCS_CKM.1/PACE FCS_CKM.1/TCAP FCS_COP.1/TCE FCS_CKM.1/SDEK	The Java Card platform fulfills the requirements directly or provides the necessary cryptographic algorithms to fulfill the requirements. No contradiction to this ST.
FCS_CKM.2	No correspondence.	Relevant for “setkey” Java Card method. No contradiction to this ST.
FCS_CKM.3	No correspondence.	Relevant for “getkey” Java Card method. No contradiction to this ST.
FCS_CKM.4 (FCS_CKM.4.1, FCS_CKM.4.1[PUF])	FCS_CKM.4	The Java Card platform fulfills the requirement that all keys are physically overwritten in a randomized manner]. This ST requires that keys are physically overwritten (independent of the values). Thus, all internal Java Card key objects fulfill the requirement of this ST. No contradiction to this ST.

Platform SFR	Correspondence in this ST	References/Remarks
FCS_COP.1 (FCS_COP.1.1[PUF_AES] FCS_COP.1.1[PUF_MAC] FCS_COP.1.1[TripleDES] FCS_COP.1.1[AES] FCS_COP.1.1[RSACipher] FCS_COP.1.1[ECDH_P1363] FCS_COP.1.1[DESMAC] FCS_COP.1.1[AESMAC] FCS_COP.1.1[RSASignature] FCS_COP.1.1[ECSignature] FCS_COP.1.1[ECAdd] FCS_COP.1.1[SHA] FCS_COP.1.1[AES_CMAC] FCS_COP.1.1[DAP])	FCS_COP.1/Hash FCS_COP.1/KW FCS_COP.1/KU FCS_COP.1/ED FCS_COP.1/HEM FCS_COP.1/HDM FCS_COP.1/MAC FCS_COP.1/CDS-ECDSA FCS_COP.1/VDS-ECDSA FCS_COP.1/TCE FCS_COP.1/TCM FCS_COP.1/SDE FCS_COP.1/VDSUCP FCS_COP.1/DecUCP FIA_API.1/PACE FIA_API.1/CA FCS_CKM.1/AES_RSA FCS_CKM.5/AES_RSA	The requirements of this ST are equivalent or fulfilled based on the platform requirements. FCS_COP.1/Hash of this ST corresponds to the platform SFR FCS_COP.1.1[SHA]. FCS_COP.1/KW and FCS_COP.1/KU are fulfilled using SFR FCS_COP.1.1[AES] of the platform. FCS_COP.1/ED is fulfilled using SFR FCS_COP.1.1[AES] of the platform. FCS_COP.1/HEM and FCS_COP.1/HDM are partly fulfilled using SFR FCS_COP.1.1[AES] and FCS_COP.1[AES_CMAC] of the platform. FCS_COP.1/MAC is fulfilled using SFR FCS_COP.1.1[AESMAC] and FCS_COP.1[AES_CMAC] of the platform. FCS_COP.1/CDS-RSA and FCS_COP.1/VDS-RSA are fulfilled using SFR FCS_COP.1.1[RSASignature] of the platform. FCS_COP.1/CDS-ECDSA and FCS_COP.1/VDS-ECDSA are fulfilled

Platform SFR	Correspondence in this ST	References/Remarks
		using SFR FCS_COP.1.1[ECSignature] of the platform. FCS_COP.1/TCE is fulfilled using SFR FCS_COP.1.1[AES] of the platform. FCS_COP.1/TCM is fulfilled using SFR FCS_COP.1.1[AES_CMAC] of the platform. FCS_COP.1/SDE is fulfilled using SFR FCS_COP.1.1[AES] of the platform. FCS_COP.1/VDSUCP is fulfilled using SFR FCS_COP.1.1[DAP] of the platform. FCS_COP.1/DecUCP is fulfilled using SFR FCS_COP.1.1[AES] of the platform. FIA_API.1/CA is fulfilled using SFR FCS_COP.1.1[ECDH_P1363] of the platform. FCS_CKM.1/AES_RSA and FCS_CKM.5/AES_RSA are partly fulfilled using the platform SFR FCS_COP.1[RSAcipher]. No contradictions to this ST.
FCS_RNG.1	FCS_RNG.1	Deterministic random number generator. No contradiction to this ST.
FCS_RNG.1[HDT]	No correspondence	Hybrid deterministic random number generator. No contradiction to this ST.
FDP_RIP.1/ABORT	No correspondence.	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_RIP.1/APDU	No correspondence.	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_RIP.1/GlobalArray_Refined	No correspondence.	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_RIP.1/bArray	No correspondence.	Out of scope (internal Java Card functionality). No contradiction to this ST.

Platform SFR	Correspondence in this ST	References/Remarks
FDP_RIP.1/KEYS	No correspondence.	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_RIP.1/TRANSIENT	No correspondence.	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_ROL.1/FIREWALL	No correspondence.	Out of scope (internal Java Card Firewall). The resulting requirements for applets are reflected in the User Guidance of the TOE. No contradiction to this ST.
Card Security Management (chapter 7.2.1.3 in platform ST)		
FAU_ARP.1	FPT_FLS.1, FPT_PHP.3	Not directly corresponding, but platform SFR is basis of fulfillment of FPT_FLS.1 and FPT_PHP.3. No contradiction to this ST.
FDP_SDI.2[DATA]	FPT_FLS.1, FPT_PHP.3	Not directly corresponding, but platform SFR is basis of fulfillment of FPT_FLS.1 and FPT_PHP.3. No contradiction to this ST.
FPR_UNO.1	No correspondence.	No direct correspondence, but relevant for the security of all cryptographic mechanisms. No contradiction to this ST.
FPT_FLS.1	FPT_FLS.1	The fulfillment of the platform SFR is part of the basis of the fulfillment of the SFR of this ST. No contradiction to this ST.
FPT_TDC.1	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
AID Management (chapter 7.2.1.4 in platform ST)		
FIA_ATD.1/AID	No correspondence.	Out of scope (internal Java Card functionality). No contradiction to this ST.
FIA_UID.2/AID	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FIA_USB.1[AID]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.

Platform SFR	Correspondence in this ST	References/Remarks
FMT_MTD.1/JCRE	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MTD.3/JCRE	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
INSTG Security Functional Requirements (chapter 7.2.2 in platform ST) This group consists of the SFRs related to the installation of the applets, which addresses security aspects outside the runtime.		
FDP_ITC.2/Installer	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_SMR.1/INSTALLER	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FPT_FLS.1/INSTALLER	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FPT_RCV.3[INSTALLER]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
ADELG Security Functional Requirements (chapter 7.2.3 in platform ST) This group consists of the SFRs related to the deletion of applets and/or packages, enforcing the applet deletion manager (ADEL) policy on security aspects outside the runtime.		
FDP_ACC.2[ADEL]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_ACF.1[ADEL]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_RIP.1[ADEL]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.1[ADEL]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.3[ADEL]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.

Platform SFR	Correspondence in this ST	References/Remarks
FMT_SMF.1[ADEL]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_SMR.1[ADEL]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FPT_FLS.1[ADEL]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
RMIG Security Functional Requirements (chapter 7.2.4 in platform ST) This group specifies the policies that control the access to the remote objects and the flow of information that takes place when the RMI service is used. Optional, not used in the platform ST.		
ODELG Security Functional Requirements (chapter 7.2.5 in platform ST) The following requirements concern the object deletion mechanism. This mechanism is triggered by the applet that owns the deleted objects by invoking a specific API method.		
FDP_RIP.1/ODEL	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FPT_FLS.1/ODEL	FPT_FLS.1	The fulfillment of the platform SFR is part of the basis of the fulfillment of the SFR of this ST. No contradiction to this ST.
CARG Security Functional Requirements (chapter 7.2.6 in platform ST) This group includes requirements for preventing the installation of packages that has not been bytecode verified, or that has been modified after bytecode verification.		
FDP_UIT.1[CCM]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_ROL.1[CCM]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_ITC.2[CCM]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FPT_FLS.1[CCM]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_ACC.1[SD]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.

Platform SFR	Correspondence in this ST	References/Remarks
FDP_ACF.1[SD]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.1[SD]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.3[SD]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_SMF.1[SD]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_SMR.1[SD]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FCO_NRO.2[SC]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_IFC.2[SC]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_IFF.1[SC]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.1[SC]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.3[SC]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_SMF.1[SC]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FIA_UID.1[SC]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FIA_UAU.1[SC]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.

Platform SFR	Correspondence in this ST	References/Remarks
FIA_UAU.4[SC]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FTP_ITC.1[SC]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
ConfG Security Functional Requirements (chapter 7.2.7 in platform ST)		
FDP_IFC.2[CFG]	No correspondence	Complete information flow control (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_IFF.1[CFG]	No correspondence	Complete information flow control (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_IFF.2[CFG]	No correspondence	Complete information flow control (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_IFF.3[CFG]	No correspondence	Complete information flow control (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_IFF.4[CFG]	No correspondence	Complete information flow control (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_IFF.5[CFG]	No correspondence	Simple security attributes (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.3[CFG]	No correspondence	Static attribute initialisation (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.1[CFG]	No correspondence	Management of security attributes (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_SMR.1[CFG]	No correspondence	Security roles (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.

Platform SFR	Correspondence in this ST	References/Remarks
FMT_SMF.1[CFG]	No correspondence	Specification of management Functions (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.
FIA_UID.1[CFG]	No correspondence	Timing of identification (CFG). Out of scope (internal Java Card functionality). No contradiction to this ST.
SecBoxG Security Functional Requirements (chapter 7.2.8 in platform ST)		
FDP_ACC.2[SecureBox]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_ACF.1[SecureBox]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.1[SecureBox]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.3[SecureBox]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_SMF.1[SecureBox]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
ModDesG Security Functional Requirements (chapter 7.2.9 in platform ST)		
FDP_IFC.1[MODULAR-DESIGN]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FDP_IFF.1[MODULAR-DESIGN]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FIA_ATD.1[MODULAR-DESIGN]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FIA_USB.1[MODULAR-DESIGN]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_MSA.1[MODULAR-DESIGN]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.

Platform SFR	Correspondence in this ST	References/Remarks
FMT_MSA.3[MODULAR-DESIGN]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_SMF.1[MODULAR-DESIGN]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FMT_SMR.1[MODULAR-DESIGN]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FPT_FLS.1[MODULAR-DESIGN]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FIA_UID.1[MODULAR-DESIGN]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
Further Security Functional Requirements (chapter 7.2.10 in platform ST)		
FAU_SAS.1[SCP]	No correspondence	Out of scope (internal Java Card functionality). No contradiction to this ST.
FIA_AFL.1[PIN]	FIA_AFL.1	Authentication Failure Handling (PIN). The fulfillment of the requirement is based on the platform requirement. No contradiction to this ST.
FPT_EMSEC.1	No correspondence	TOE emanation. No direct correspondence, but platform requirement leads to protection of cryptographic keys, PINs and user data. No contradiction to this ST.
FPT_PHP.3	FPT_PHP.3	Resistance to physical attack. The fulfillment of the requirement is based on the platform requirement. No contradiction to this ST.
FCS_CKM.2	No correspondence.	Relevant for “setkey” Java Card method. No contradiction to this ST.
FCS_CKM.3	No correspondence.	Relevant for “getkey” Java Card method. No contradiction to this ST.
FDP_SDI.2[SENSITIVE_RESULT]	FPT_FLS.1, FPT_PHP.3	Not directly corresponding, but platform SFR is basis of fulfillment of FPT_FLS.1 and FPT_PHP.3. No contradiction to this ST.

Table 2: Assessment of the platform SFRs.

2.4.3 Assessment of the Platform Objectives

The following table provides an assessment of all relevant Platform objectives.

Platform Objective	Correspondence in this ST	References/Remarks
OT.SID	No correspondence	Out of scope. No contradiction to this ST.
OT.SID_MODULE	No correspondence	Out of scope. No contradiction to this ST.
OT.FIREWALL	No correspondence	Out of scope. No contradiction to this ST.
OT.GLOBAL_ARRAYS_CONFID	No correspondence	Out of scope. No contradiction to this ST.
OT.GLOBAL_ARRAYS_INTEG	No correspondence	Out of scope. No contradiction to this ST.
OT.NATIVE	No correspondence	Out of scope. No contradiction to this ST.
OT.OPERATE	No correspondence	Out of scope. No contradiction to this ST.
OT.REALLOCATION	No correspondence	Out of scope. No contradiction to this ST.
OT.RESOURCES	No correspondence	Out of scope. No contradiction to this ST.
OT.SENSITIVE_RESULTS_INTEG	No correspondence	Indirectly relevant for the correct function of the TOE of this ST, but no corresponding objectives for the TOE of this ST. No contradiction to this ST.
OT.ALARM	No correspondence	Out of scope. No contradiction to this ST.
OT.CIPHER	No correspondence	Indirectly relevant for the correct function of the TOE of this ST, i.e. O.AuthentTOE, O.Enc, O.DataAuth, O.TChann, O.SecMan. No contradiction to this ST.
OT.RNG	O.RBGS	The objective regarding random number generation is related. No contradiction to this ST.
OT.KEY-MNGT	No correspondence	Indirectly relevant for the correct function of the TOE of this ST, i.e. O.AuthentTOE, O.Enc, O.DataAuth, O.TChann, O.SecMan.

Platform Objective	Correspondence in this ST	References/Remarks
		No contradiction to this ST.
OT.PIN-MNGT	No correspondence	Indirectly relevant for the correct function of the TOE of this ST, i.e. O.I&A. No contradiction to this ST.
OT.TRANSACTION	No correspondence	Out of scope. No contradiction to this ST.
OT.OBJ-DELETION	No correspondence	Out of scope. No contradiction to this ST.
OT.APPLI-AUTH	No correspondence	Out of scope. No contradiction to this ST.
OT.DOMAIN-RIGHTS	No correspondence	Out of scope. No contradiction to this ST.
OT.COMM_AUTH	No correspondence	Out of scope. No contradiction to this ST.
OT.COMM_INTEGRITY	No correspondence	Out of scope. No contradiction to this ST.
OT.COMM_CONFIDENTIALITY	No correspondence	Out of scope. No contradiction to this ST.
OT.EXT-MEM	No correspondence	Out of scope. No contradiction to this ST.
OT.CARD-MANAGEMENT	No correspondence	Out of scope. No contradiction to this ST.
OT.SCP.IC	O.PhysProt	The objectives are related. No contradiction to this ST.
OT.SCP.RECOVERY	O.PhysProt	The objectives are related. No contradiction to this ST.
OT.SCP.SUPPORT	No correspondence	Out of scope. No contradiction to this ST.
OT.IDENTIFICATION	No correspondence	Out of scope. No contradiction to this ST.
OT.SEC_BOX_FW	No correspondence	Out of scope. No contradiction to this ST.
OT.RND	O.RBGS	The objective regarding random number generation is related. No contradiction to this ST.
OT.CARD-CONFIGURATION	No correspondence	Out of scope. No contradiction to this ST.
OT.ATTACK-COUNTER	No correspondence	Out of scope.

Platform Objective	Correspondence in this ST	References/Remarks
		No contradiction to this ST.
OT.RESTRICTED-MODE	No correspondence	Out of scope. No contradiction to this ST.

Table 3: Assessment of the platform objectives.

2.4.4 Assessment of Platform Threats

The following table provides an assessment of all relevant Platform threats.

Platform Threat	Correspondence in this ST	References/Remarks
T.CONFID-APPLI-DATA	No correspondence	Out of scope. No contradiction to this ST.
T.CONFID-JCS-CODE	No correspondence	Out of scope. No contradiction to this ST.
T.CONFID-JCS-DATA	No correspondence	Out of scope. No contradiction to this ST.
T.INTEG-APPLI-CODE	No correspondence	Out of scope. No contradiction to this ST.
T.INTEG-APPLI-CODE.LOAD	No correspondence	Out of scope. No contradiction to this ST.
T.INTEG-APPLI-DATA[REFINED]	No correspondence	Out of scope. No contradiction to this ST.
T.INTEG-APPLI-DATA.LOAD	No correspondence	Out of scope. No contradiction to this ST.
T.INTEG-JCS-CODE	No correspondence	Out of scope. No contradiction to this ST.
T.INTEG-JCS-DATA	No correspondence	Out of scope. No contradiction to this ST.
T.SID.1	No correspondence	Out of scope. No contradiction to this ST.
T.SID.2	No correspondence	Out of scope. No contradiction to this ST.
T.EXE-CODE.1	No correspondence	Out of scope. No contradiction to this ST.
T.EXE-CODE.2	No correspondence	Out of scope. No contradiction to this ST.
T.NATIVE	No correspondence	Out of scope. No contradiction to this ST.
T.MODULE_EXEC	No correspondence	Out of scope. No contradiction to this ST.

Platform Threat	Correspondence in this ST	References/Remarks
T.RESOURCES	No correspondence	Out of scope. No contradiction to this ST.
T.UNAUTHORIZED_CARD_MNGT	T.FaUpD	No direct correspondence, but related to T.FaUpD of this ST. No contradiction to this ST.
T.COM_EXPLOIT	No correspondence	Out of scope. No contradiction to this ST.
T.LIFE_CYCLE	No correspondence	Out of scope. No contradiction to this ST.
T.OBJ-DELETION	No correspondence	Out of scope. No contradiction to this ST.
T.PHYSICAL	T.PhysAttack	No contradiction to this ST.
T.OS_OPERATE	No correspondence	Out of scope. No contradiction to this ST.
T.RND	No correspondence	Out of scope. No contradiction to this ST.
T.CONFIG	No correspondence	Out of scope. No contradiction to this ST.
T.SEC_BOX_BORDER	No correspondence	Out of scope. No contradiction to this ST.
T.MODULE_REPLACEMENT	No correspondence	Out of scope. No contradiction to this ST.
T.ATTACK-COUNTER	No correspondence	Out of scope. No contradiction to this ST.

Table 4: Threats of the platform ST.

2.4.5 Assessment of Platform Organisational Security Policies

The Organisational Security Policy “OSP.VERIFICATION” focuses on the integrity of loaded applets, which is fulfilled by the TOE of this ST since only applets digitally signed by cryptovision can be loaded. This policy does not contradict to the policies of this ST.

The platform ST contains the Organisational Security Policy “OSP.PROCESS-TOE” referring to accurate identification of each TOE instance. This policy will be fulfilled by a distinct product code for the platform and for the composite TOE each. This policy does not contradict to the policies of this ST.

The Organisational Security Policy “OSP.KEY-CHANGE” states that initial security domain keys (APSD) shall be changed before any operation on its Security Domain. This policy does not contradict to the policies of this ST.

The Organisational Security Policy “OSP.SECURITY-DOMAINS” states that security domains can be dynamically created, deleted and blocked during usage phase in post-issuance mode. This policy does not contradict to the policies of this ST.

The Organisational Security Policy “OSP.SECURE-BOX” focuses on the secure box mechanism, which is not used by the TOE. This policy does not contradict to the policies of this ST.

2.4.6 Assessment of Platform Operational Environment

2.4.6.1 Assessment of Platform Assumptions

In the first column, the following table lists all assumptions of the Platform ST. The last column provides an explanation of relevance for the Composite TOE.

Platform Assumption	Relevance for Composite ST
A.APPLET	A.APPLET states that applets loaded post-issuance do not contain native methods. This assumption leads to appropriate directives in the user guidance [Guidance_PRE].
A.VERIFICATION	This assumption targets the applet code verification. In the context of this ST the TOE guarantees that only code digitally signed by cryptovisionand can be loaded and that this code was verified before production. Regarding post-issuance loading of applets, this assumption leads to appropriate directives in the user guidance [Guidance_PRE].
A.USE_DIAG	A.USE_DIAG is required in the platform ST to cover secure communication during packaging, finishing and personalisation. This is reflected by appropriate measures in the production and delivery of the TOE of this ST.
A.USE_KEYS	A.USE_KEYS assumes that that the keys which are stored outside the TOE and which are used for secure communication and authentication between smart card and terminals are protected for confidentiality and integrity in their own storage environment. This assumption leads to appropriate directives in the user guidance [Guidance_PRE].
A.PROCESS-SEC-IC	A.PPROCESS-SEC-IC of the platform ST states that it is assumed that security procedures are used after delivery of the TOE by the TOE Manufacturer up to delivery to the end consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use). This means that the phases after TOE delivery are assumed to be protected appropriately. This is reflected by appropriate measures in the production and delivery of the TOE of this ST.
A.APPS-PROVIDER	A.APPS-PROVIDER assumes that the application provider is a trusted actor that provides basic or secure applications, and that the applicatrion provider is resposible for his security domain keys. This leads to appropriate directives in the user guidance [Guidance_PRE].
A.VERIFICATION-AUTHORITY	A.VERIFICATION-AUTHORITY assumes that the verification authority is a trusted actor and able to guarantee and check the digital signature attached to a basic or secure application. This is reflected by appropriate directives in the user guidance [Guidance_PRE].

Table 5: Assumptions of the Platform ST.

2.4.6.2 Assessment of Platform Security Objectives for the Operational Environment

There are the following Platform Security Objectives for the Operational Environment that have to be considered.

Platform Objective for the Environment	Relevance for Composite ST
OE.APPLET	The platform objective for the environment states that applets loaded post-issuance do not contain native methods. This objective for the environment leads to appropriate directives in the user guidance [Guidance_PRE].
OE.VERIFICATION	The platform objective for the environment targets the applet code verification. In the context of this ST the TOE guarantees that only code digitally signed by cryptovisionand can be loaded and that this code was verified before production. Regarding post-issuance loading of applets, this objective for the environment leads to appropriate directives in the user guidance [Guidance_PRE]. There it is stated that all applets loaded to the TOE have to be verified.
OE.CODE-EVIDENCE	The platform objective for the environment focusses on application code loaded post-issuance. It has to be ensured that the loaded application has not been changed since the code verification. This objective for the environment leads to appropriate directives in the user guidance [Guidance_PRE].
OE.APPS-PROVIDER	The application provider (AP) shall be a trusted actor that provides applications. The AP is responsible for its security domain keys. This objective for the environment leads to appropriate directives in the user guidance [Guidance_PRE].
OE.VERIFICATION-AUTHORITY	The platform objective for the environment targets the verification authority for post-issuance loading. This entity should be a trusted actor who is able to guarantee and check the digital signature attached to an application. This objective for the environment leads to appropriate directives in the user guidance [Guidance_PRE].
OE.KEY-CHANGE	The platform objective for the environment focusses on the change of the security domain initial keys before any operation on it. This objective for the environment leads to appropriate directives in the user guidance [Guidance_PRE].
OE.SECURITY-DOMAINS	The platform objective for the environment states that security domains can be dynamically created, deleted and blocked during usage phase in post-issuance mode. This objective for the environment

	leads to appropriate directives in the user guidance [Guidance_PRE].
OE.USE_DIAG	The platform objective for the environment covers secure communication during packaging, finishing and personalisation. This is corresponding to O.Data_Conf of this composite ST.
OE.USE_KEYS	This platform objective for the environment states that the keys which are stored outside the TOE and which are used for secure communication and authentication between Smart Card and terminals are protected for confidentiality and integrity in their own storage environment. This is reflected by appropriate measures in the production and delivery of the TOE of this ST.
OE.PROCESS_SEC_IC	OE. PROCESS_SEC_IC states that security procedures shall be used after TOE Delivery up to delivery to the end consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use). This is reflected by appropriate measures in the production and delivery of the TOE of this ST.

Table 6: Platform Security Objectives and SFRs for the Operational Environment

3 Security problem definition

This chapter has been taken from [PP0104] and [PP0107] with only minor modifications.

3.1 Introduction

3.1.1 Assets

The assets of the TOE are

- user data which integrity and confidentiality shall be protected,
- cryptographic services and keys which shall be protected against unauthorized use or misuse,
- Update Code Packages (UCP).

The cryptographic keys are TSF data because they are used for cryptographic operations protecting user data and the enforcement of the SFR relies on these data for the operation of the TOE.

Additional assets are:

- user data and time stamps shall be integrity protected,
- time services which time base shall be protected against manipulation.

The cryptographic keys are TSF data because they are used for cryptographic time stamp operations protecting user data and audit records, and the enforcement of the SFR relies on these data for the operation of the TOE. The audit records are TSF data generated by the TSF and exported to the user.

3.1.2 Users and subjects

The TOE knows external entities (users) as

- human user communicating with the TOE for security management of the TOE,
- application component using the cryptographic and other security services of the TOE and supporting the communication with remote entities,
- remote entity exchanging user data and TSF data with the TOE over insecure media.

The TOE communicates with

- human user through a secure channel,
- application component through a secure channel,
- remote entities over a trusted channel using cryptographic mechanisms including mutual authentication.

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

3.1.3 Objects

The TSF operates user data objects and TSF data objects (i. e. passive entities, that contain or receive information, and upon which subjects perform operations). User data objects are imported, used in cryptographic operation, temporarily stored, exported and destroyed after use. The Update Code Packages are user data objects imported and stored in the TOE until use for creation of an updated CSP. TSF data objects are created, temporarily or permanently stored, imported, exported and destroyed as objects of the security management. They may contain e. g. cryptographic keys with their security attributes, certificates, Authentication Data Records with authentication reference data of a user. Cryptographic keys are objects of the key management.

User data objects of the time stamp service are imported, used in time stamp operation, exported and destroyed after use. TSF data objects time and time stamps are created, temporarily or permanently stored, imported, exported and destroyed as objects of the security management. Cryptographic keys used by the time stamp service are TSF data objects of the key management as described above.

3.1.4 Security attributes

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

- User Identity (User-ID),
- Authentication Reference Data,
- Role with detailed access rights.

Passwords as Authentication Reference Data have the security attributes

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

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

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

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

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

The TOE knows at least the following roles taken by a user or a subject acting on behalf of a user[1]:

- Unidentified User: this role is associated with any user not (successfully) identified by the TOE. This role is assumed after start-up of the TOE. The TSF associated actions allowed for the Unidentified User are defined in SFR FIA_UID.1.
- Unauthenticated User: this role is associated with an identified user but not (successfully) authenticated user. The TSF associated actions allowed for the Unauthenticated User are defined in SFR FIA_UAU.1.

[1] This paragraph is taken from the protection profiles [PP0104] and [PP0107] and defines the minimum set of roles. Besides these roles, the TOE uses additional roles (e.g. SMA Manager) that are specified in detail in the Guidance documents [AGD_PRE] and [AGD_OPE].

- Administrator: successful authenticated user allowed to access the TOE in order to perform management functions. It is taken by a human user or a subject acting on behalf of a human user after successful authentication as Administrator.

  The Administrator role is split in more detailed roles:
  - Crypto-Officer: role that is allowed to access the TOE in order to perform management of a cryptographic TSF.
  - User Administrator: role that is allowed to access the TOE in order to perform user management.
  - The Timekeeper is allowed to adjust the internal time.
  - Auditor: role that is allowed to configure the audit functionality, review audit data and export audit trails.
  - Update Agent: authorized user for installation of imported and verified as authentic Update Code Package.

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

- *Key Owner:* successful authenticated user allowed to perform cryptographic operation with their own keys. This role may be claimed by human user or an entity.
- *Application Component:* subjects in this role are allowed to use assigned security services of the TOE without authenticated human user session (e. g. export and import of wrapped keys). This role may be assigned to an entity communicating through a physically separated secure channel or through a trusted channel (which requires assured identification of its end points).

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

General cryptographic keys have at least the security attributes

- Key identity that uniquely identifies the key,
- Key entity, i. e. the identity of the entity this key is assigned to,
- Key type, i. e. as secret key, private key, public key,
- Key usage type, identifying the cryptographic mechanism or service the key can be used for, e. g. a private signature key may be used by a digital signature-creation mechanism (cf. FCS_COP.1/CDS-ECDSA.1 or FCS_COP.1/CDS-RSA), and depending on the certificate for data authentication with identity of guarantor (cf. FDP_DAU.2/Sig) by key usage type “DigSign”, or time stamp service (cf. FDP_DAU.2/TS) by key usage type “TimeStamp”, or attestation (cf. FDP_DAU.2/Att) by key usage type “Attestation”.
- Key access control attributes, i. e. list of combinations of the identity of the user, the role for which the user is authenticated and the allowed key management function or cryptographic operation, including
  - Import of the key is allowed or forbidden,
  - Export of the key is allowed or forbidden,

and may have the security attribute

- Key validity time period, i. e. the time period for operational use of the key; the key must not be used before or after this time slot,
- Key usage counter, i. e. the number of operations performed with this key e. g. number of signature created with a private signature key.

Cryptographic keys used for the time stamp service and the export of audit records have at least the security attributes

- Key identity that uniquely identifies the key,
- Key entity, i. e. the identity of the entity this key is assigned to,
- Key type, i. e. as secret key, private key, public key,
- Key usage type, identifying the cryptographic mechanism or service the key can be used for, where the keys for time stamp service (cf. FDP_DAU.2/TS) have the key usage type "TimeStamp",

and may have the security attribute

- Key usage counter, i. e. the number of operations performed with this key, where the key usage counter of the private key used for time stamp service counts the number of created signature
- Key validity time period, i. e. the time period for operational use of the key; the key must not be used before or after this time slot.

UCP have at least the security attributes

- Issuer of the Update Code Package,
- Version Number of the Update Code Package.

3.2 Threats

3.2.1 T.DataCompr Compromise of communication data

An unauthorized entity gets knowledge of the information contained in data stored on TSF controlled media or transferred between the TOE and authenticated external entities.

3.2.2 T.DataMani Unauthorized generation or manipulation of communication data

An unauthorized entity generates or manipulates user data stored on TSF controlled media or transferred between the TOE and authenticated external entities and accepted as valid data by the recipient.

3.2.3 T.Masqu Masquerade authorized user

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

3.2.4 T.ServAcc Unauthorized access to TOE security services

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

3.2.5 T.PhysAttack Physical attacks

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

3.2.6 T.FaUpD Faulty Update Code Package

An unauthorized entity provides an unauthorized faulty Update Code Package enabling attacks against integrity of TSF implementation, confidentiality and integrity of user data and TSF data after installation of the faulty Update Code Package.

3.3 Organisational security policies

3.3.1 OSP.SecCryM Secure cryptographic mechanisms

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

3.3.2 OSP.SecService Security services of the TOE

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

3.3.3 OSP.KeyMan Key Management

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

3.3.4 OSP.TC Trust center

The trust centers provide secure certificates for trustworthy certificate holder with correct security attributes. The TOE uses certificates for identification and authentication of users, access control and secure use of security services of the TOE including key management and attestation.

3.3.5 OSP.Update Authorized Update Code Packages

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

The following organisational security policies are added due to the PP module [PP0107]:

3.3.6 OSP.Audit Audit for selected security activities and events

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

3.3.7 OSP.TimeService Time Service and Time stamp service

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

3.4 Assumptions

3.4.1 A.SecComm Secure communication

Remote entities support trusted channel using cryptographic mechanisms. The operational environment shall protect the local communication channels by trusted channels using cryptographic mechanisms or by secure channel using non-cryptographic security measures.

4 Security Objectives

This chapter describes the security objectives for the TOE and the security objectives for the TOE environment.

4.1 Security Objectives for the TOE

This section describes the security objectives for the TOE addressing the aspects of identified threats to be countered by the TOE and organizational security policies to be met by the TOE.

4.1.1 O.AuthentTOE Authentication of the TOE to external entities

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

4.1.2 O.Enc Confidentiality of user data by means of encryption and decryption

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

4.1.3 O.DataAuth Data authentication by cryptographic mechanisms

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

4.1.4 O.RBGS Random bit generation service

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

4.1.5 O.TChann Trusted channel

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

Note the TSF can establish the trusted channel by means of secure cryptographic mechanisms only if the other endpoint supports these secure cryptographic mechanisms as well. If trusted channel cannot be established by means of secure cryptographic mechanisms due to missing security functionality of the user then the operational environment shall provide a secure channel protecting the communication by non-cryptographic security measures, cf. A.SecComm and OE.SecComm.

4.1.6 O.I&A Identification and authentication of users

The TOE shall uniquely identify users and verify the claimed identity of the user before providing access to any controlled resources with the exception of self-test, identification of the TOE and authentication of the TOE. The TOE shall authenticate IT entities using secure cryptographic mechanisms.

4.1.7 O.AccCtrl Access control

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

4.1.8 O.SecMan Security management

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

4.1.9 O.TST Self-test

The TSF performs self-tests during initial start-up, at the request of the authorised user and after power-on. The TSF enters secure state if self-test fails or attacks are detected.

4.1.10 O.PhysProt Physical protection

The TSF protects the confidentiality and integrity of user data, TSF data and its correct operation against physical attacks and environmental stress. In case of platform architecture the TSF protects the secure execution environment for and the communication with the application component running on the TOE.

4.1.11 O.SecUpCP Secure import of Update Code Package

The TSF verifies the authenticity of received encrypted Update Code Package, decrypts authentic Update Code Package and allows authorized users to store decrypted Update Code Package.

4.1.12 O.Audit Audit

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

4.1.13 O.TimeService Time services

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

4.2 Security Objectives for the Operational Environment

4.2.1 OE.CommInf Communication infrastructure

The operational environment shall provide public key infrastructure for entities in the communication networks. The trust centers generate secure certificates for trustworthy certificate holder with correct security attributes. They distribute securely their certificate signing public key for verification of digital signature of the certificates and run a directory service for dissemination of certificates and provision of revocation status information of certificates.

4.2.2 OE.AppComp Support of the Application component

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

4.2.3 OE.SecManag Security management

The operational environment shall implement appropriate security management for secure use of the TOE including user management, key management. It ensures secure key management outside the TOE and uses the trust center services to determine the validity of certificates. The cryptographic keys and cryptographic key components shall be assigned to the secure cryptographic mechanisms they are intended to be used-with and to the entities authorized for their use.

4.2.4 OE.SecComm Protection of communication channel

Remote entities shall support trusted channels with the TOE using cryptographic mechanisms. The operational environment shall protect the local communication channels by trusted channels using cryptographic mechanisms or by secure channel using non-cryptographic security measures.

4.2.5 OE.SUCP Signed Update Code Packages

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

4.2.6 OE.Audit Review and availability of audit records

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

4.2.7 OE.TimeSource External time source

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

4.3 Security Objective Rationale

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

	T.DataCompr	T.DataMani	T.Masqu	T.ServAcc	T.PhysAttack	T.FaUpD	OSP.SecCryM	OSP.SecService	OSP.KeyMan	OSP.TC	OSP.Update	A.SecComm	OSP.Audit	OSP.TimeService
O.AccCtrl				x										
O.AuthentTOE							x	x						
O.DataAuth		x					x	x						
O.Enc	x						x	x						
O.I&A			x	x			x	x						
O.PhysProt					x									

	T.DataCompr	T.DataMani	T.Masqu	T.ServAcc	T.PhysAttack	T.FaUpD	OSP.SecCryM	OSP.SecService	OSP.KeyMan	OSP.TC	OSP.Update	A.SecComm	OSP.Audit	OSP.TimeService
O.RBGS							x	x						
O.SecMan			x				x		x	x				
O.SecUpCP						x					x			
O.Tchann	x	x	x	x			x	x						
O.TST					x									
OE.AppComp	x	x		x						x				
OE.CommInf	x	x		x				x	x	x				
OE.SecComm	x	x		x								x		
OE.SecManag			x					x	x					
OE.SUCP						x					x			
O.Audit													x	
O.TimeService														x
OE.Audit													x	
OE.TimeSource														x

Table 7: Overview of the security objectives coverage

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 following text was taken from [PP0104] and [PP0107], respectively.

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

- O.Enc requires the TOE to provide encryption and decryption as security service for the users to protect the confidentiality of user data,
- O.TChann requires the TOE to support trusted channel between TSF and the application component, and between TSF and other users, and the application component and other users with authentication of all communication end points, protected communication ensuring the confidentiality and integrity of the communication and to prevent misuse of the session of authorized users.
- OE.AppComp requires the application component to support the TOE for communication with users and trust center.
- OE.CommInf requires the operational environment to provide the communication infrastructure especially trust center services.

- OE.SecComm requires the operational environment to protect the confidentiality and integrity of communication over local communication channel by physical security measures and remote entities to support trusted channels by means of cryptographic mechanisms. If a trusted channel cannot be established due to missing security functionality of the application component or human user communication channel the operational environment shall protect the communication, cf. A.SecComm and OE.SecComm.

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

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

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

- O.I&A requires the TSF to identify uniquely users and verify the claimed identity of the user before providing access to any controlled resources with the exception of self-test, identification of the TOE and authentication of the TOE.
- O.TChann requires the TSF to provide authentication of all communication end points of the trusted channel.
- O.SecManag requiring the TSF to provide security management of users, TSF, TSF data and cryptographic keys by means of secure cryptographic mechanisms and using certificates.
- OE.SecMan requiring the operational environment to implement appropriate security management for secure use of the TOE including user management.

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

- O.I&A requires the TSF to uniquely identify users and to authenticate users before providing access to any controlled resources with the exception of self-test, identification of the TOE and authentication of the TOE. Note an unauthenticated user is allowed to request authentication of the TOE.
- O.AccCtrl requires the TSF to control access on security services, operations on user data, management of TSF and TSF data.

- O.Tchann requires mutual authentication of the external entity and the TOE and the authentication of communicated data to prevent misuse of the communication with external entities. The operational environment is required by OE.SecComm to ensure secure channels if trusted channel cannot be established.
- The operational environment OE.CommInf requires provision of a public key infrastructure for entity authentication and OE.AppComp requires the application to support communication with trust centers.

The threat T.PhysAttack "Physical attacks" is directly countered by the security objectives

- O.PhysProt requires the TSF to protect the confidentiality and integrity of user data, TSF data and its correct operation against physical attacks and environmental stress.
- O.TST requires the TSF to perform self-tests and to enter secure state if self-test fails or attacks are detected as means to ensure robustness against perturbation.

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

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

The organizational security policy OSP.SecCryM "Secure cryptographic mechanisms" is implemented by means of secure cryptographic mechanisms required in

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

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

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

The organizational security policy OSP.TC “Trust center” is implemented by security objectives for the TOE and the operational environment:

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

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

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

The organizational security policy OSP.Audit “Audit for selected security events” is directly implemented by

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

The organizational security policy OSP.TimeService “Time services” is directly implemented by

- the security objective for the TOE O.TimeService “Time services” requiring the TOE to provide an internal time service and time stamp service for the user, and
- the security objective for the operational environment OE.TimeSource “External time source” requiring the operational environment to provide reliable external time stamps for adjustment of TOE internal time source.

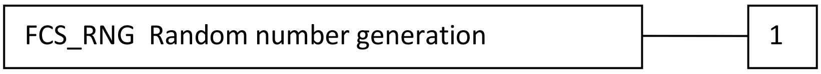
5 Extended Component Definition

5.1 Generation of random numbers (FCS_RNG)

Family behavior

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

Component leveling:



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: There are no management activities foreseen.

Audit: 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].

5.2 Cryptographic key derivation (FCS_CKM.5)

This chapter describes a component of the family Cryptographic key management (FCS_CKM) for key derivation as process by which one or more keys are calculated from either a pre-shared key or a shared secret and other information. Key derivation is the deterministic repeatable process by which one or more keys are calculated from both a pre-shared key or shared secret, and other information, while key generation required by FCS_CKM.1 uses internal random numbers.

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

Management: There are no management activities foreseen.

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

a) Minimal: Success and failure of the activity.

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

FCS_CKM.5 Requires the TOE to provide key derivation.

FCS_CKM.5 Cryptographic key derivation

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]

FCS_CKM.4 Cryptographic key destruction

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

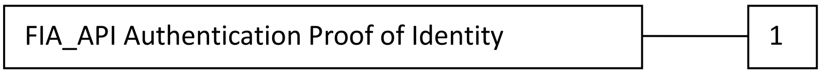
5.3 Authentication Proof of Identity (FIA_API)

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

Family behavior

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

Component leveling:


FIA_API Authentication Proof of Identity

1

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

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

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

Audit: 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.

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

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

Family behavior

This family requires confidentiality protection of exchanged TSF data.

Component leveling:

FPT_TCT Inter-TSF TSF data confidentiality transfer protection

1

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

Management: There are no management activities foreseen.

Audit: There are no auditable events foreseen.

FPT_TCT.1 TSF data confidentiality transfer protection

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FMT_MTD.1 Management of TSF data or
FMT_MTD.3 Secure TSF data]

FPT_TCT.1.1 The TSF shall enforce the [assignment: access control SFP, information flow control SFP] by providing the ability to [selection: transmit, receive, transmit and receive] TSF data in a manner protected from unauthorised disclosure.

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

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

Family behavior

This family requires integrity protection of exchanged TSF data.

Component leveling:

FPT_TIT Inter-TSF TSF data integrity transfer protection

1

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

Management: There are no management activities foreseen.

Audit: There are no auditable events foreseen.

FPT_TIT.1 TSF data integrity transfer protection

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FMT_MTD.1 Management of TSF data or
FMT_MTD.3 Secure TSF data]

FPT_TIT.1.1 The TSF shall enforce the [assignment: access control SFP, information flow control SFP] to [selection: transmit, receive, transmit and receive] TSF data in a manner protected from [selection: modification, deletion, insertion, replay] errors.

FPT_TIT.1.2 The TSF shall be able to determine on receipt of TSF data, whether [selection: modification, deletion, insertion, replay] has occurred.

5.6 TSF data import with security attributes (FPT_ISA)

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

Family behavior

This family requires TSF data import with security attributes.

Component leveling:

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

Management: There are no management activities foreseen.

Audit: There are no auditable events foreseen.

FPT_ISA.1 Import of TSF data with security attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FMT_MTD.1 Management of TSF data or
FMT_MTD.3 Secure TSF data]
[FMT_MSA.1 Management of security attributes, or
FMT_MSA.4 Security attribute value inheritance]
FPT_TDC.1 Inter-TSF basic TSF data consistency

FPT_ISA.1.1 The TSF shall enforce the [assignment: access control SFP, information flow control SFP] when importing TSF data, controlled under the SFP, from outside of the TOE.

FPT_ISA.1.2 The TSF shall use the security attributes associated with the imported TSF data.

FPT_ISA.1.3 The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the TSF data received.

FPT_ISA.1.4 The TSF shall ensure that interpretation of the security attributes of the imported TSF data is as intended by the source of the TSF data.

FPT_ISA.1.5 The TSF shall enforce the following rules when importing TSF data controlled under the SFP from outside the TOE: [assignment: additional importation control rules].

5.7 TSF data export with security attributes (FPT_ESA)

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

Family behavior

This family requires TSF data export with security attributes.

Component leveling:

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

Management: There are no management activities foreseen.

Audit: There are no auditable events foreseen.

FPT_ESA.1 Export of TSF data with security attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FMT_MTD.1 Management of TSF data or
FMT_MTD.3 Secure TSF data]
[FMT_MSA.1 Management of security attributes, or
FMT_MSA.4 Security attribute value inheritance]
FPT_TDC.1 Inter-TSF basic TSF data consistency

FPT_ESA.1.1 The TSF shall enforce the [assignment: access control SFP, information flow control SFP] when exporting TSF data, controlled under the SFP(s), outside of the TOE.

FPT_ESA.1.2 The TSF shall export the TSF data with the TSF data's associated security attributes.

FPT_ESA.1.3 The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported TSF data.

FPT_ESA.1.4 The TSF shall enforce the following rules when TSF data is exported from the TOE: [assignment: additional exportation control rules].

5.8 Stored data confidentiality (FDP_SDC)

To define the security functional requirements of the TOE an additional family (FDP_SDC.1) of the Class FDP (User data protection) is defined here.

The family “Stored data confidentiality (FDP_SDC)” is specified as follows.

Family behavior

This family provides requirements that address protection of user data confidentiality while these data are stored within memory areas protected by the TSF. The TSF provides access to the data in the memory through the specified interfaces only and prevents compromise of their information bypassing these interfaces. It complements the family Stored data integrity (FDP_SDI) which protects the user data from integrity errors while being stored in the memory.

Component leveling:

FDP_SDC Stored data confidentiality

1

FDP_SDC.1 Requires the TOE to protect the confidentiality of information of the user data in specified memory areas.

Management: There are no management activities foreseen.

Audit: There are no auditable events foreseen.

FDP_SDC.1 Stored data confidentiality

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP_SDC.1.1 The TSF shall ensure the confidentiality of the information of the user data while it is stored in the [assignment: memory area].

6 IT Security Requirements

The CC allows several operations to be performed on functional requirements; refinement, selection, assignment, and iteration are defined in paragraph 8.1 of Part 1 [CC_1] of the CC. Each of these operations is used in this ST and the underlying PP.

Operations already performed in the underlying Protection Profiles ([PP0104], [PP0107]) are uniformly marked by ***bold italic*** font style except in cases where text was deleted; for further information on details of the operation, please refer to [PP0104] and [PP0107].

Operations performed within this Security Target are marked by **bold underlined** font style; further information on details of the operation is provided in foot notes.

6.1 Security Functional Requirements for the TOE

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

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

The TOE provides hybrid encryption and decryption combined with data integrity mechanisms for the cipher text as cryptographic security service of the TOE. The encryption FCS_COP.1/HEM combines the generation of a data encryption key and message authentication code (MAC) key, the asymmetric encryption of the data encryption key with an asymmetric key encryption key, cf. FCS_CKM.1/ECKA-EG, FCS_CKM.1/RSA, and the symmetric encryption of the data with the data encryption key and data integrity mechanism with MAC calculation for the cipher text. The receiver reconstructs the data encryption key and the MAC key, cf. FCS_CKM.5/ECKA-EG, calculates the MAC for the cipher text and compares it with the received MAC. If the integrity of the cipher text is determined than the receiver decrypts the cipher text with the data decryption key, cf. FCS_COP.1/HDM.

In general, authentication is the provision of assurance of the claimed identity of an entity. The TOE authenticates human users by password, cf. FIA_UAU.5.1 clause 1. But a human user may authenticate themselves to a token and the token authenticates to the TOE. Cryptographic authentication mechanisms allow an entity to prove its identity or the origin of its data to a verifying entity by demonstrating its knowledge of a secret. The entity authentication is required by FIA_UAU.5.1 clauses (2) to (6). The chapter 5.3 describes SFR for the authentication of the TOE to external entities required by the SFR FIA_API.1. This authentication may include attestation of the TOE as genuine TOE sample, cf. 6.1.4. The authentication may be mutual as required for trusted channels in chapter 6.1.5.

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

Trusted channel requires mutual authentication of endpoints with key exchange of key agreement, protection of confidentiality by means of encryption and cryptographic data integrity protection.

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

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

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

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

Elliptic curve	**Key size**	**Standard**
brainpoolP256r1	256 bit	RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR03111]
brainpoolP384r1	384 bit	RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR03111]
brainpoolP512r1	512 bit	RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR03111]
Curve P-256	256 bit	FIPS PUB 186-4 B.4 and D.1.2.3 [FIPS186-4]
Curve P-384	384 bit	FIPS PUB 186-4 B.4 and D.1.2.4 [FIPS186-4]
Curve P-521	521 bit	FIPS PUB 186-4 B.4 and D.1.2.5 [FIPS186-4]

Table 8: Elliptic curves, key sizes and standards

For Diffie-Hellman key exchange refer to the following groups:

Name	**IANA no.**	**Specified in**
256-bit random ECP group	19	[RFC5903]
384-bit random ECP group	20	[RFC5903]
521-bit random ECP group	21	[RFC5903]
brainpoolP256r1	28	[RFC6954]
brainpoolP384r1	29	[RFC6954]
brainpoolP512r1	30	[RFC6954]

Table 9: Recommended groups for the Diffie-Hellman key exchange

6.1.1 Key management

6.1.1.1 Management of security attributes

6.1.1.1.1 FDP_ACC.1/KM Subset access control – Cryptographic operation

Hierarchical to: No other components

Dependencies: FDP_ACF.1 Security attribute based access control

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

(1) subjects: Crypto-Officer[2]***, Key Owner;***

(2) objects: operational cryptographic keys;

(3) operations: key generation, key derivation, key import, key export, key destruction.

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

6.1.1.1.2 FMT_MSA.1/KM Management of security attributes – Key security attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MSA.1.1/KM The TSF shall enforce the ***Key Management SFP*** and ***Cryptographic Operation SFP*** to restrict the ability to

(***1) change_default*** the security attributes ***Identity of the key, Key entity of the key, Key type, Key usage type, Key access control attributes, Key validity time period*** to **Crypto-Officer**[3]***,***

(2) modify or delete the security attributes Identity of the key, Key entity, Key type, Key usage type, Key validity time period of an existing key to none,

(3) modify independent on key usage the security attributes Key usage counter of an existing key to none.

(4) modify the security attributes Key access control attribute of an existing key to **Crypto-Officer**[4]***,***

(5) query the security attributes Key type, Key usage type, Key access control attributes, Key validity time period and Key usage counter of an identified key to **Crypto-Officer and Key Owner**[5]***.***

PP application note 1: The refinements repeats parts of the SFR component in order to avoid iteration of the component.

6.1.1.1.3 FMT_MSA.3/KM Static attribute initialisation – Key management

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MSA.3.1/KM The TSF shall enforce ***the Key Management SFP, Cryptographic Operation SFP and Update SFP*** to provide ***restrictive*** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/KM The TSF shall allow the **Crypto-Officer**[6] to specify alternative initial values to override the default values when a ***cryptographic key*** object or information is created.

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

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles

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

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

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

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

FMT_SMF.1 Specification of Management Functions

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

(1) create according to FCS_CKM.1 the ***cryptographic keys*** to ***the*** Crypto-Officer[7]***,***

(2) import according to FPT_TCT.1/CK, FPT_TIT.1/CK and FPT_ISA.1/CK the cryptographic keys to Crypto-Officer[8]***,***

(3) export according to FPT_TCT.1/CK, FPT_TIT.1/CK and FPT_ESA.1/CK the cryptographic keys to Crypto-Officer[9] ***if security attribute of the key allows export,***

(4) delete according to FCS_CKM.4 the cryptographic keys to Crypto-Officer[10]***.***

PP application note 2: The bullets (2) to (4) are refinements to avoid an iteration of component and therefore printed in bold in the original protection profile.

6.1.1.2 Hash based functions

6.1.1.2.1 FCS_COP.1/Hash Cryptographic operation – Hash

Hierarchical to: No other components.

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

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

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/Hash The TSF shall perform ***hash generation*** in accordance with a specified cryptographic algorithm ***SHA-256, SHA-384, SHA-512*** and cryptographic key sizes ***none*** that meet the following: ***FIPS 180-4 [FIPS180-4].***

PP application note 3: The hash function is a cryptographic primitive used for HMAC, cf. FCS_COP.1/HMAC, digital signature creation, cf. FCS_COP.1/CDS-*, digital signature verification, cf. FCS_COP.1/VDS-*, and key derivation, cf. FCS_CKM.5.

Developer note: The implementation is based on the functionality of the platform [ST_Javacard]. Due to platform restrictions only resistant against AVA_VAN.5 for temporary data (e.g. as used for generating session keys), but not if repeatedly applied to the same input data.

6.1.1.3 Management of Certificates

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

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

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

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

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

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

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

(1) create[11]***, modify, clear and delete*** the ***root key pair*** to <u>Crypto-Officer</u>[12]***.***

(2) import and delete a known as authentic public key of a certification authority in a PKI to <u>Crypto-Officer</u>[13]***.***

PP application note 4: The root key is defined here with respect to the key hierarchy known to the TOE. In case of clause (1), i. e. may be a key pair of an TOE internal key hierarchy. In clause (2) it may be a root public key of a PKI or a public key of another certification authority in a PKI known as authentic certificate signing key. The PKI may be used for user authentication, key management and signature-verification. The second bullet is a refinement to avoid an iteration of component and therefore printed in bold in the original protection profile.

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

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FMT_MTD.1 Management of TSF data or
FMT_MTD.3 Secure TSF data]

FPT_TIT.1.1/Cert The TSF shall enforce the ***Key Management SFP*** to ***receive certificate*** in a manner protected from ***modification and insertion*** errors.

FPT_TIT.1.2/Cert The TSF shall be able to determine on receipt ***of certificate***, whether ***modification and insertion*** has occurred.

Developer note: The security functionality according to FPT_TIT.1/Cert supports card-verifiable (cv) certificates according to TR-03110 v2.10 [TR-03110v2.10].

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

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FMT_MTD.1 Management of TSF data or
FMT_MTD.3 Secure TSF data]
[FMT_MSA.1 Management of security attributes, or
FMT_MSA.4 Security attribute value inheritance]
FPT_TDC.1 Inter-TSF basic TSF data consistency

FPT_ISA.1.1/Cert The TSF shall enforce the ***Key management SFP*** when importing ***certificates*** , controlled under the SFP, from outside of the TOE.

FPT_ISA.1.2/Cert The TSF shall use the security attributes associated with the imported ***certificate***.

FPT_ISA.1.3/Cert The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the ***certificates*** received.

FPT_ISA.1.4/Cert The TSF shall ensure that interpretation of the security attributes of the imported ***certificates*** is as intended by the source of the ***certificates***.

[11] “create” denotes initial setting a root key

[12] [selection: Administrator, Crypto-Officer].

[13] [selection: Administrator, Crypto-Officer].

FPT_ISA.1.5/Cert The TSF shall enforce the following rules when importing ***certificates*** controlled under the SFP from outside the TOE:

(1) The TSF imports the TSF data in certificates only after successful verification of the validity of the certificate in the certificate chain until known as authentic certificate according to FMT_MTD.1/RK.

(2) The validity verification of the certificate shall include

(a) the verification of the digital signature of the certificate issuer except for root certificates,

(b) the security attributes in the certificate pass the interpretation according to FPT_TDC.1.

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

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_TDC.1.1/Cert The TSF shall provide the capability to consistently interpret ***security attributes of cryptographic keys in the certificate and identity of the certificate issuer*** when shared between the TSF and another trusted IT product.

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

(1) the TOE reports about conflicts between the Key identity of stored cryptographic keys and cryptographic keys to be imported,

(2) the TOE does not change the security attributes Key identity, Key entity, Key type, Key usage type and Key validity time period of public key being imported from the certificate,

(3) the identity of the certificate issuer shall meet the identity of the signer of the certificate

when interpreting ***the certificate from a trust center.***

PP application note 5: The security attributes assigned to certificate holder and cryptographic key in the certificate are used as TSF data of the TOE. The certificate is imported from trust center directory service or any other source but verified by the TSF (i.e. if verified successfully the source is the trusted IT product trust center directory server).

Developer note: The TOE only accepts certificates in the context of Terminal Authentication.

6.1.1.4 Key generation, agreement and destruction

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

Hybrid cryptosystems (FCS_CKM.1/ECKA-EG, FCS_CKM.1/AES_RSA) are a combination of a public key cryptosystem with an efficient symmetric key cryptosystem.

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

6.1.1.4.1 FCS_RNG.1 Random number generation

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1 The TSF shall provide a **deterministic**[14] random number generator that implements:

- **(DRG.3.1) If initialized with a random seed using a PTRNG of class PTG.2 (as defined in [AIS20]) as random source, the internal state of the RNG shall have at least 256 bit of entropy.**
- **(DRG.3.2)The RNG provides forward secrecy (as defined in [AIS20]).**
- **(DRG.3.3) The RNG provides backward secrecy even if the current internal state is known (as defined in [AIS20]).**[15]

FCS_RNG.1.2 The TSF shall provide random numbers that meet

- **(DRG.3.4) The RNG, initialized with a random seed using a PTRNG of class PTG.2 (as defined in [AIS20]) as random source, generates output for which 2^{48} strings of bit length 128 are mutually different with probability at least $1\text{-}2^{-24}$.**
- **(DRG.3.5) Statistical tests cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A in [AIS20].**[16]

PP application note 6: The random bit generation shall be used for key generation and key agreement according to all instantiations of FCS_CKM.1, challenges in cryptographic protocols and cryptographic operations using random values according to FCS_COP.1/HEM and FCS_COP.1/TCE. The TOE provides the random number generation as security service for the user.

Developer note: The implementation is based on the functionality of the platform [ST_Javacard].

6.1.1.4.2 FCS_CKM.1/AES Cryptographic key generation – AES key

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/AES The TSF shall generate cryptographic ***AES*** key in accordance with a specified cryptographic key generation algorithm ***AES*** and key size ***128 bits***, **256 bits**[17] that meet the following: ***ISO 18033-3 [ISO18033-3]***.

PP application note 7: The cryptographic key may be used with FCS_COP.1/ED, e. g. for internal purposes.

Developer note: The implementation is based on the functionality of the platform [ST_Javacard] and uses the DRG.3 random generator of the platform.

[14] [selection: physical, non-physical true, deterministic, hybrid physical, hybrid deterministic]

[15] [assignment: list of security capabilities]

[16] [assignment: a defined quality metric]

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

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

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.5.1/AES The TSF shall derive cryptographic ***AES key*** from **a byte array of variable length**[18] in accordance with a specified cryptographic key derivation algorithms ***AES key generation using bit string derived from input parameters with KDF*** and specified cryptographic key sizes ***128 bits***, **no other key length**[19] that meet the following: ***NIST SP 800-56C [NIST-SP800-56C]***.

Developer note: The implementation is based on the functionality of the platform [ST_Javacard].

Developer note: The length of the counter can be either 16, 24 or 32 bit.

Developer note: Please note that sufficient entropy (>100 bit) must be used in the input provided for the key derivation.

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

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/ECC The TSF shall generate cryptographic ***elliptic curve*** key ***pair*** in accordance with a specified cryptographic key generation algorithm ***ECC key pair generation with*** **brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384, Curve P-521**[20] and specified cryptographic key sizes **256, 384, 512 and 521 bit, respectively**[21] that meet the following: **RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR03111], or FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 and D.1.2.5 [FIPS186-4], respectively**[22].

Developer note: The implementation is based on the functionality of the platform [ST_Javacard]. The EC key pair generation is implemented according to [ISO/IEC 14888-3], [ANSI X9.62-2005] and [FIPS PUB 186-4].

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

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

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.5.1/ECC The TSF shall derive cryptographic ***elliptic curve*** key ***pair*** from **a byte array of variable length**[23] in accordance with a specified cryptographic key derivation algorithm

[18] [assignment: input parameters]

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

[20] [selection: elliptic curves in the table 2]

[21] [selection: key size in the table 2]

[22] [selection: standards in the table 2]

[23] [assignment: input parameters]

ECC key pair generation with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384[24] ***using bit string derived from input parameters with*** **X9.63 Key Derivation Function according to [TR03111], page 27**[25] and specified cryptographic key sizes **256, 384, 512 bit**[26] that meet the following: **RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR03111], or FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS186-4], respectively**[27]**, and** ***[TR03111].***

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

Developer note: Since the SHA-256 of the Javacard platform is used, the function must only be used once for secret data.

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

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/RSA The TSF shall generate cryptographic ***RSA*** key ***pair*** in accordance with a specified cryptographic key generation algorithm ***RSA*** and specified cryptographic key sizes **from 2000 bit to 4096 bit in one bit steps**[28] that meet the following: ***PKCS #1 v2.2 [PKCS1].***

PP application note 10: The cryptographic key sizes assigned in FCS_CKM.1/RSA must be at least 2000 bits. Cryptographic key sizes of at least 3000 bits are recommended. The FCS_CKM.1/RSA assigns given security attributes Key identity and Key entity. The security attribute *Key usage type* is DS-RSA for the private signature-creation key and public signature-verification key, RSA_ENC for public RSA encryption key and private RSA decryption key.

Developer note: The RSA key generation is based on the functionality of the platform [ST_Javacard] and implemented according to [FIPS186-4].

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

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.5.1/ECDHE The TSF shall derive cryptographic ***ephemeral keys for data encryption and MAC with AES-128,*** AES-256[29] from an ***agreed shared secret*** in accordance with a specified cryptographic key derivation algorithm ***Elliptic Curve Diffie-Hellman ephemeral key agreement*** brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-

[24] [selection: elliptic curves in table 2]

[25] [assignment: KDF]

[26] [selection: key size in the table 2]

[27] [selection: standards in the table 2]

[28] [assignment: cryptographic key sizes]

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

256, Curve P-384[30] ***and*** **256-bit random ECP group, 384-bit random ECP group, brainpoolP256r1, brainpoolP384r1, brainpoolP512r1**[31] ***with a key derivation from the shared secret*** **Key Derivation Function for Session Keys**[32] and specified cryptographic key sizes ***128 bits,*** **256 bits**[33] that meet the following: ***TR-03111 [TR-03111]***.

PP application note 11: The input parameters for key derivation is an agreed shared secret established by means of Elliptic Curve Diffie-Hellman. The table 2 lists elliptic curves and table 3 lists the Diffie-Hellman Groups for agreement of the shared secret. The SHA-1 shall be supported for generation of 128 bits AES keys. The SHA-256 shall be selected and used to generate 256 bit AES keys.

6.1.1.4.8 FCS_CKM.1/ECKA-EG Cryptographic key generation – ECKA-EG key generation

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/ECKA-EG The TSF shall generate ***an ephemeral*** cryptographic ***elliptic curve*** key ***pair for ECKGA-EG [TR-03111], sender role*** in accordance with a specified cryptographic key generation algorithm ***ECC key pair generation with*** **brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384**[34] and specified cryptographic key sizes **256 bit, 384 bit, 512 bit**[35] that meet the following: **RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 and D.1.2.5 [FIPS PUB 186-4].** [36].

Developer note: The implementation is based on the functionality of the platform [ST_Javacard]. The EC key pair generation is implemented according to [ISO/IEC 14888-3], [ANSI X9.62-2005] and [FIPS PUB 186-4].

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

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.5.1/ECKA-EG The TSF shall derive cryptographic ***data encryption key and MAC keys for AES 128,*** **AES-256**[37] from ***a private and a public ECC key*** in accordance with a specified cryptographic key derivation algorithms ***ECKGA-EG [TR-03111] with*** **brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384**[38] ***and X9.63 Key Derivation Function*** and specified cryptographic ***symmetric*** key sizes ***128 bits,*** **256 bits**[39] that meet the following: ***TR-03111 [TR-03111], chapter 4.3.2.2.***

[30] [selection: elliptic curves in table 2]

[31] [selection: group in table 3]

[32] [assignment: key derivation function]

[33] [selection:256 bits, none other]

[34] [selection: elliptic curves in the table 2]

[35] [selection: key size in the table 2]

[36] [selection: standards in the table 2]

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

[38] [selection: elliptic curves in table 2]

[39] [selection:256 bits, none other]

PP application note 12: FCS_CKM.5/ECKA-EG is used by both the sender (encryption) and the recipient (decryption) to compute a secret point S_{AB} on an elliptic curve and the derived shared secret Z_{AB}. The shared secret is then used as input to the key derivation function to derive two symmetric keys, the encryption key and the MAC key which are used to encrypt or decrypt the message according to FCS_COP.1/HEM or FCS_COP.1/HDM, respectively. Sender and recipient use however different inputs to FCS_CKM.5/ECKA-EG. The sender first generates an ephemeral ECC key pair according to FCS_CKM.1/ECKA-EG and uses the generated ephemeral private key and the static public key of the recipient as input. The recipient first extracts the ephemeral public key from the encrypted message and uses the ephemeral public key and the static private key (cf. FCS_CKM.1/ECC for key generation) as input. The selection of elliptic curve, the ECC key size and length of the shared secret shall correspond to the selection of the AES key size, e. g. brainpoolP256r1 and 256 bits seed, ECC key and AES keys. FCS_CKM.1/ECKA-EG and FCS_CKM.5/ECKA-EG do not provide self-contained security services for the user but are necessary steps for FCS_COP.1/HEM and FCS_COP.1/HDM (refer to the next section 6.1.3).

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

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/AES_RSA The TSF shall generate ***and encrypt seed, derive*** cryptographic keys ***from seed for data encryption and MAC with AES-128,*** AES-256[40] in accordance with a specified cryptographic key generation algorithm ***X9.63 Key Derivation Function [ANSI-X9.63] and RSA EME-OAEP [PKCS#1]*** and specified cryptographic ***symmetric*** key sizes ***128 bits*, 256 bits**[41] that meet the following***: ISO/IEC 18033-3 [ISO/IEC 18033-3], PKCS #1 v2.2 [PKCS#1].***

PP application note 13: The asymmetric cryptographic key sizes used in FCS_CKM.1/AES_RSA must be at least 2000 bits. Cryptographic key sizes of at least 3000 bits are recommended. FCS_CKM.1/AES_RSA and FCS_CKM.1.1/AES_RSA do not provide self-contained security services for the user but they are only necessary steps for FCS_COP.1/HEM respective FCS_COP.1/HDM (refer to the next section 6.1.3).

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

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.5.1/AES_RSA The TSF shall derive cryptographic ***data encryption key and MAC key for AES-128,*** **AES-256** [42]from ***decrypted RSA encrypted seed*** in accordance with a specified cryptographic key derivation algorithm ***RSA EME-OAEP [PKCS#1] and X9.63 [ANSI-X9.63] Key Derivation Function*** and specified cryptographic ***symmetric*** key sizes ***128 bits*, 256 bits** [43]that meet the following: ***ISO/IEC 14888-2 [ISO/IEC 14888-2].***

6.1.1.4.12 FCS_CKM.4 Cryptographic key destruction

Hierarchical to: No other components.

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

[41] [selection:256 bits, none other]

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

[43] [selection:256 bits, none other]

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

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method **overwriting the keys**[44] that meets the following: **none**[45].

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

6.1.1.5 Key import and export

6.1.1.5.1 FCS_COP.1/KW Cryptographic operation – Key wrap

Hierarchical to: No other components.

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

FCS_COP.1.1/KW The TSF shall perform ***key wrap*** in accordance with a specified cryptographic algorithm ***AES-Keywrap*** **KWP**[46] and cryptographic key sizes ***of the key encryption key 128 bits*****, *none other***[47] that meet the following: ***NIST SP800-38F [NIST-SP800-38F]***.

PP application note 14: The selection of the length of the key encryption key shall be equal or greater than the security bits of the wrapped key for its cryptographic algorithm.

Developer note: For further information, please refer to the guidance [Guidance_OPE].

6.1.1.5.2 FCS_COP.1/KU Cryptographic operation – Key unwrap

Hierarchical to: No other components.

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

FCS_COP.1.1/KU The TSF shall perform ***key unwrap*** in accordance with a specified cryptographic algorithm ***AES-Keywrap*** **KWP**[48] and cryptographic key sizes ***of the key encryption key 128 bits*****, *none other***[49] that meet the following: ***NIST SP800-38F [NIST-SP800-38F]***.

Developer note: For further information, please refer to the guidance [Guidance_OPE].

[44] [assignment: cryptographic key destruction method]

[45] [assignment: list of standards]

[46] [selection: KW, KWP]

[47] [selection:256 bits, none other]

[48] [selection: KW, KWP]

[49] [selection:256 bits, none other]

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

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FMT_MTD.1 Management of TSF data or
FMT_MTD.3 Secure TSF data]

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

Developer note: For further information, please refer to the guidance [Guidance_OPE].

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

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FMT_MTD.1 Management of TSF data or
FMT_MTD.3 Secure TSF data]

FPT_TIT.1.1/CK The TSF shall enforce the ***Key Management SFP*** to ***transmit and receive cryptographic keys*** in a manner protected from ***modification and insertion*** errors ***according to FCS_COP.1/KW***.

FPT_TIT.1.2/CK The TSF shall be able to determine on receipt of ***cryptographic keys***, whether ***modification and insertion*** has occurred ***according to FCS_COP.1/KU***.

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

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FMT_MTD.1 Management of TSF data or
FMT_MTD.3 Secure TSF data]
[FMT_MSA.1 Management of security attributes, or
FMT_MSA.4 Security attribute value inheritance]
FPT_TDC.1 Inter-TSF basic TSF data consistency

FPT_ISA.1.1/CK The TSF shall enforce the ***Key Management SFP*** when importing ***cryptographic key***, controlled under the SFP, from outside of the TOE.

FPT_ISA.1.2/CK The TSF shall use the security attributes associated with the imported ***cryptographic key***.

FPT_ISA.1.3/CK The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the ***cryptographic key*** received.

FPT_ISA.1.4/CK The TSF shall ensure that interpretation of the security attributes of the imported ***cryptographic key*** is as intended by the source of the ***cryptographic key***.

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

(1) The TSF imports the TSF data in certificates only after successful verification of the validity of the certificate including verification of digital signature of the issuer and validity time period.

(2) **None**[50]***.***

PP application note 15: The operational environment is obligated to use trust center services for secure key management, cf. OE.SecManag.

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

Hierarchical to: No other components.

Dependencies: No dependencies.

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

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

(1) the TOE reports about conflicts between the Key identity of stored cryptographic keys and cryptographic keys to be imported,

(2) the TOE does not change the security attributes Key identity, Key type, Key usage type and Key validity time period of the key being imported

when interpreting ***the imported key data object.***

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

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or

FDP_IFC.1 Subset information flow control]

[FMT_MTD.1 Management of TSF data or

FMT_MTD.3 Secure TSF data]

[FMT_MSA.1 Management of security attributes, or

FMT_MSA.4 Security attribute value inheritance]

FPT_TDC.1 Inter-TSF basic TSF data consistency

FPT_ESA.1.1/CK The TSF shall enforce the ***Key Management SFP*** when exporting ***cryptographic key***, controlled under the SFP(s), outside of the TOE.

FPT_ESA.1.2/CK The TSF shall export the ***cryptographic key*** with the ***cryptographic key's*** associated security attributes.

FPT_ESA.1.3/CK The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported ***cryptographic key***.

FPT_ESA.1.4/CK The TSF shall enforce the following rules when ***cryptographic key*** is exported from the TOE: **None.**[51]

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

[50] [assignment: additional importation control rules]

[51] [assignment: additional exportation control rules]

6.1.2 Data encryption

6.1.2.1 FCS_COP.1/ED Cryptographic operation – User data encryption and decryption

Hierarchical to: No other components.

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

FCS_COP.1.1/ED The TSF shall perform data encryption and decryption in accordance with a specified cryptographic algorithm ***symmetric data encryption according to AES-128 and*** **AES-256**[52] ***in CBC and*** **no other**[53] ***mode*** and cryptographic key size ***128 bits,*** **256 bits**[54] that meet the following: ***NIST SP800-38A [NIST-SP800-38A], ISO/IEC 18033-3 [ISO/IEC 18033-3], ISO/IEC 10116 [ISO/IEC 10116]***.

PP application note 17: Data encryption and decryption should be combined with data integrity mechanisms in Encrypt-then-MAC order, i. e. the MAC is calculated for the ciphertext and verified before decryption. The modes of operation should combine encryption with data integrity mechanisms to authenticated encryption, e. g. the Cipher Block Chaining Mode (CBC, cf. NIST SP800-38A) should be combined with CMAC (cf. FCS_COP.1/MAC) or HMAC (cf. FCS_COP.1/HMAC). For combination of symmetric encryption, decryption and data integrity mechanisms by means of CCM or GCM refer to the next section 6.1.3.

6.1.3 Hybrid encryption with MAC for user data

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

Hierarchical to: No other components.

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

FCS_COP.1.1/HEM The TSF shall perform ***hybrid data encryption and MAC calculation*** in accordance with a specified cryptographic algorithm ***asymmetric key encryption according to*** **FCS_CKM.1/ECKA-EG and FCS_CKM.1/AES_RSA** [55]***, symmetric data encryption according to AES-128,*** **AES-256**[56] ***[FIPS197] in*** **CBC**[57] **[NIST-SP800-38A]** ***mode with*** **CMAC [NIST-SP800-38B]**[58] ***calculation*** and cryptographic ***symmetric*** key sizes ***128 bits,*** **256 bits**[59] that meet the following***: the referenced standards above according to the chosen selection.***

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

[53] [selection: CRT, OFB, CFB, no other]

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

[55] [selection: FCS_CKM.1/ECKA-EG, FCS_CKM.1/AES_RSA]

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

[57] [selection: CBC,[NIST-SP800-38A], CCM,[NIST-SP800-38C], GCM][NIST-SP800-38D]]

[58] [selection: CMAC,[NIST-SP800-38B], GMAC,[NIST-SP800-38D], HMAC][RFC2104]]

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

PP application note 18: Hybrid data encryption and MAC calculation is a self-contained security services of the TOE. The generation and encryption of the seed, derivation of encryption and MAC keys as well as the AES encryption and MAC calculation are only a steps of this service. The hybrid encryption is combined with MAC as data integrity mechanisms for the cipher text, i. e. encrypt-then-MAC creation for CMAC.

Developer note: For further informnation, please refer to the guidance [Guidance_OPE].

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

Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1/HDM	The TSF shall perform ***hybrid MAC verification and data decryption*** in accordance with a specified cryptographic algorithm ***asymmetric key decryption according to*** **FCS_CKM.5/ECKA-EG and FCS_CKM.5/AES-RSA**[60]***, verification of*** CMAC [NIST-SP800-38B][61] ***and symmetric data decryption according to AES with*** AES-128, AES-256[62] ***[FIPS197] in mode*** CBC **[NIST-SP800-38A]** [63] and cryptographic ***symmetric*** key sizes ***128 bits,*** **256 bits**[64] that meet the following***: the referenced standards above according to the chosen selection.***

PP application note 19: Hybrid data decryption and MAC verification is a self-contained security services of the TOE. The decryption of the seed and derivation of the encryption key and MAC keys as well as the AES decryption and MAC verification are only a steps of this service. The used symmetric key shall meet the AES CMAC and the AES algorithm for decryption of the cipher text for MAC, e. g. verification-then-decrypt for CMAC.

Developer note: For further informnation, please refer to the guidance [Guidance_OPE].

6.1.4 Data integrity mechanisms

Cryptographic data integrity mechanisms comprise 2 types of mechanisms – symmetric message authentication code mechanisms and asymmetric digital signature mechanisms. A message authentication code mechanism comprises the generation of a MAC for original message, the verification of a given pair of message and MAC and symmetric key management. The MAC may be applied to plaintext without encryption but if combined with encryption it should be applied to ciphertexts in Encrypt-then-MAC order.

6.1.4.1 FCS_COP.1/MAC Cryptographic operation – MAC using AES

Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or

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

[61] [selection: CMAC,[NIST-SP800-38B], GCM,[NIST-SP800-38D], HMAC][RFC2104]]

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

[63] [selection: CBC,[NIST-SP800-38A], CCM,[NIST-SP800-38C], GMAC][NIST-SP800-38D]]

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

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

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/MAC The TSF shall perform ***MAC generation and verification*** in accordance with a specified cryptographic algorithm ***AES-128*** and AES-256[65] ***[FIPS197] CMAC [NIST-SP800-38B] and*** no other[66] and cryptographic key sizes ***128 bits***, 256 bits[67] that meet the following: ***the referenced standards above according to the chosen selection.***

PP application note 20: The MAC may be applied to plaintext and cipher text. The AES-128 CMAC is mandatory. The selection of AES-256 and the key sizes shall correspond to each other.

6.1.4.2 FCS_COP.1/HMAC Cryptographic operation – HMAC

Hierarchical to: No other components.

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

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

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/HMAC The TSF shall perform ***HMAC generation and verification*** in accordance with a specified cryptographic algorithm ***HMAC-SHA256 and*** no other[68] and cryptographic key sizes 256 bit[69] that meet the following: ***RFC2104 [RFC2104], ISO/IEC 9797-2 [ISO/IEC 9797-2].***

PP application note 21: The cryptographic key is a random bit string generated by FCS_RNG.1 or a referenced internal secret. The cryptographic key sizes assigned in FCS_COP.1/HMAC must be at least 128 bits.

Developer note: Please note that the HMAC function is based on the HMAC of the Java Card platform [ST_Javacard] and neither protects input, output or the key.

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

Hierarchical to: No other components.

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

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

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/CDS-ECDSA The TSF shall perform ***signature-creation*** in accordance with a specified cryptographic algorithm ***ECDSA with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384, Curve P-521***[70] and cryptographic key sizes 256, 384, 512 and 521 bit[71] that meet the following:

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

[66] [selection: GMAC,[NIST-SP800-38D], no other]

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

[68] [selection: HMAC-SHA-1, HMACSHA384, no other]

[69] [assignment: cryptographic key sizes]

[70] [selection: elliptic curves in the table 2]

[71] [selection: key size in the table 2]

RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4][72]**.**

PP application note 22: The selection of elliptic curve and cryptographic key sizes shall correspond to each other, e. g. elliptic curve brainpoolP256r1 and key size 256 bits.

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

Hierarchical to: No other components.

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

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

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/VDS-ECDSA The TSF shall perform ***signature-verification*** in accordance with a specified cryptographic algorithm ***ECDSA with*** **brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384, Curve P-521**[73] and cryptographic key sizes **256, 384, 512 and 521 bit**[74] that meet the following: **RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4]**[75]**.**

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

Hierarchical to: No other components.

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

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

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/CDS-RSA The TSF shall perform ***signature-creation*** in accordance with a specified cryptographic algorithm ***RSA and EMSA-PSS*** and cryptographic key sizes **2000-4096 bit**[76] that meet the following: ***ISO/IEC 14888-2 [ISO/IEC 14888-2], PKCS #1, v2.2 [PKCS#1]***.

PP application note 23: The cryptographic key sizes assigned in FCS_CKM.1/RSA must be at least 2000 bits. Cryptographic key sizes of at least 3000 bits are recommended.

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

Hierarchical to: No other components.

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

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

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

[72] [selection: standards in the table 2]

[73] [selection: elliptic curves in the table 2]

[74] [selection: key size in the table 2]

[75] [selection: standards in the table 2]

[76] [assignment: cryptographic key sizes]

FCS_COP.1.1/VDS-RSA The TSF shall perform ***signature-verification*** in accordance with a specified cryptographic algorithm ***RSA and EMSA-PSS*** and cryptographic key sizes 2000-4096 bit[77] that meet the following: ***ISO/IEC 14888-2 [ISO/IEC 14888-2], PKCS #1, v2.2 [PKCS#1].***

PP application note 24: The cryptographic key sizes assigned in FCS_CKM.1/RSA must be at least 2000 bits. Cryptographic key sizes of at least 3000 bits are recommended.

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

Hierarchical to: FDP_DAU.1 Basic Data Authentication

Dependencies: FIA_UID.1 Timing of identification

FDP_DAU.2.1/Sig The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of ***user data imported according to FDP_ITC.2/UD by means of*** FCS_COP.1/CDS-RSA, FCS_COP.1/CDS-ECDSA[78] ***and keys holding the security attributes Key identity assigned to the guarantor and Key usage type "Signature service".***

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

PP application note 25: The TSF according to FDP_DAU.2/Sig is intended for a signature service for user data. The user data source shall select the security attributes Key entity of the guarantor and Key usage type "Signature service" of the cryptographic key for the signature service in the security attributes provided with the user data. The user data source subject shall meet the Key access control attributes for the signature-creation operation. The verification of the evidence requires a certificate showing the identity of the key entity as user generated the evidence and the key usage type as digital signature.

6.1.5 Authentication and attestation of the TOE, trusted channel

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

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_API.1.1/PACE The TSF shall provide ***a PACE in ICC role*** to prove the identity of the ***TOE*** to an external entity ***and establishing a trusted channel according to FTP_ITC.1 case 1 or 2***.

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

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_API.1.1/CA The TSF shall provide ***a Chip Authentication Version 2 according to [TR03110] section 3.4*** to prove the identity of the ***TOE*** to an external entity ***and establishing a trusted channel according to FTP_ITC.1 case 3***.

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

Hierarchical to: FDP_DAU.1 Basic Data Authentication

[77] [assignment: cryptographic key sizes]

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

Dependencies: FIA_UID.1 Timing of identification

FDP_DAU.2.1/Att The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of ***attestation data by means of*** **FCS_COP.1/CDS-ECDSA**[79] ***and keys holding the security attributes Key identity assigned to the TOE sample and Key usage type "Attestation".***

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

PP application note 26: The attestation data shall represent the TOE sample as genuine sample of the certified product. The attestation data may include the identifier of the certified product, the serial number of the device or a group of product samples as certified product, the hash value of the TSF implementation and some TSF data as result of self-test, or other data. It may be generated internally or may include internally generated and externally provided data. The assigned cryptographic mechanisms shall be appropriate for attestation meeting OSP.SecCryM, e. g. digital signature, a group signature or a direct anonymous attestation mechanism as used for Trusted Platform Modules [TPMLib, Part 1] or FIDO U2F Authenticators [FIDO-ECDAA].

6.1.5.4 FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1 The TSF shall provide a communication channel between TSF and another trusted IT product that is **logically separated from other communication channels**[80]and provides assured identification of its end points **Authentication of TOE and remote entity according to the case in Table 10**[81] and protection of the channel data from modification or disclosure **according to the case in Table 10**[82] ***as required by*** **cryptographic operation according to the case in Table 10**[83].

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

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

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

[79] [selection: FCS_COP.1/CDS-RSA, FCS_COP.1/CDS-ECDSA, ECDAA according to [selection: [TPMLib1][FIDO-ECDAA]], [assignment: other cryptographic authen-tication mechanism]]

[80] [selection: logically separated from other communication channels, using physical separated ports]

[81] [selection: Authentication of TOE and remote entity according to the case in Table 10]

[82] [assignment: according to the case in Table 10]

[83] [selection: cryptographic operation according to the case in Table 10]

3	FIA_API.1/CA, FIA_UAU.5.1 (4) or (5), and (6)	FCS_CKM.1/TCAP	modification	FCS_COP.1/TCM
			disclosure	FCS_COP.1/TCE

Table 10: Operation in SFR for trusted channel

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

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/PACE The TSF shall generate cryptographic keys for ***MAC with for FCS_COP.1/TCM and if selected encryption keys for FCS_COP.1/TCE*** in accordance with a specified cryptographic key ***agreement*** algorithm ***PACE with*** **brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384**[84] ***and Generic Mapping in ICC role*** and specified cryptographic key sizes **128, 256 bits**[85] that meet the following***: ICAO Doc9303, Part 11, section 4.4 [ICAO Doc9303]***.

PP application note 27: PACE is used to authenticate the TOE and the application component, or TOE and human user using a terminal. It establishes a trusted channel with MAC integrity protection and if selected encryption.

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

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/TCAP The TSF shall generate cryptographic keys ***for encryption according to FCS_COP.1/TCE and MAC according to FCS_COP.1/TCM*** in accordance with a specified cryptographic key ***agreement*** algorithms ***Terminal Authentication version 2 and Chip Authentication Version 2*** and specified cryptographic key ***sizes*** **128 bits, 256 bits**[86] that meet the following: ***BSI TR-03110 [TR-03110], section 3.3 and 3.4***.

PP application note 28: The terminal authentication protocol version 2 is used for authentication of the Application component according to FIA_UAU.5 and is a prerequisite for Chip Authentication Version 2.

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

Hierarchical to: No other components.

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

[84] [selection: elliptic curves in table 2]

[85] [selection: 128 bits, 192 bits, 256 bits]

[86] [selection: 128 bits, 192 bits, 256 bits]

FCS_COP.1.1/TCE The TSF shall perform ***encryption and decryption*** in accordance with a specified cryptographic algorithm ***AES in*** **CBC [NIST-SP800-38A]**[87] ***mode*** and cryptographic key sizes **128 bits, 256 bits**[88] that meet the following: ***[FIPS197].***

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

Hierarchical to: No other components.

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

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

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/TCM The TSF shall perform ***MAC calculation and MAC verification*** in accordance with a specified cryptographic algorithm ***AES*** **CMAC [NIST-SP800-38B]**[89] and cryptographic key sizes **128 bits, 256 bits**[90] that meet the following***: [FIPS197].***

6.1.6 User identification and authentication

6.1.6.1 FIA_ATD.1 User attribute definition – Identity based authentication

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users:

(1) Identity,

(2) Authentication reference data,

(3) Role.

6.1.6.2 FMT_MTD.1/RAD Management of TSF data – Authentication reference data

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

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

(1) create the ***initial Authentication reference data of all authorized users*** to **User Administrator**[91]***,***

(2) delete the Authentication reference data of an authorized user to **User Administrator**[92]***,***

(3) modify the Authentication reference data to the corresponding authorized user.

[87] [selection: CBC,[NIST-SP800-38A], CCM,[NIST-SP800-38C], GCM][NIST-SP800-38D]]

[88] [selection: 128 bits, 192 bits, 256 bits]

[89] [selection: CMAC,[NIST-SP800-38B], GMAC][NIST-SP800-38D]]

[90] [selection: 128 bits, 192 bits, 256 bits]

[91] [selection: Administrator, User Administrator]

[92] [selection: Administrator, User Administrator]

(4) create the permanently stored session key of trusted channel as Authentication reference data to **User Administrator**[93]

(5) define the time in range **1 – (2^32-1) seconds**[94] ***after which the user security attribute Role is reset according to FMT_SAE.1 to*** **User Administrator**[95]**,**

(6) define the value **Unauthenticated user**[96] ***to which the security attribute Role shall be reset according to FMT_SAE.1 to*** **User Administrator**[97]***.***

PP application note 29: <Refined> The User Administrator is responsible for user management. The User Administrator install and revoke a user as known authorized user of the TSF as defined in clause (1). The User Administrator may define additional authentication reference data as described in clause (3), i. e. the trusted channel combines initial authentication of communication endpoints (cf. FIA_UAU.5.1 clause (3) and (4)) with agreement of session keys used for authentication of exchanged messages (cf. FIA_UAU.5.1 clause (5)). The session keys may be permanently stored for the trusted communication with the known authorized entity. The user manages its own authentication reference data to prevent impersonation based of known authentication data (e.g. as addressed by FMT_MTD.3). The bullets (2) to (6) are refinements in order to avoid an iteration of component and therefore printed in bold in the original protection profile.

6.1.6.3 FMT_MTD.3 Secure TSF data

Hierarchical to: No other components.

Dependencies: FMT_MTD.1 Management of TSF data

FMT_MTD.3.1 The TSF shall ensure that only secure values are accepted for ***passwords by enforcing change of initial passwords after first successful authentication of the user to different operational password***.

6.1.6.4 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 **a User Administrator configurable positive integer within 1 - 15**[98] unsuccessful authentication attempts occur related to

(1) PACE based authentication,

(2) Password based authentication,

(3) Cryptographic Entity Authentication.[99]

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been **met**[100], the TSF shall **delay the next authentication attempt or block the authentication, configurable by the administrator**.[101]

[93] [selection: Administrator, User Administrator]

[94] [assignment: time frame]

[95] [selection: Administrator, User Administrator]

[96] [selection: Unidentified user, Unauthenticated user]

[97] [selection: Administrator, User Administrator],

[98] [selection: [assignment: positive integer number], an [selection: Administrator, User Administrator] configurable positive integer within [assignment: range of acceptable values]]

[99] [assignment: list of authentication events]

[100] [selection: met, surpassed]

[101] [assignment: list of actions]

6.1.6.5 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.

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***.

FIA_USB.1.3 The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users:

(1) after successful identification of the user the attribute Role of the subject shall be changed from Unidentified user to Unauthenticated user;

(2) after successful authentication of the user for a selected role the attribute Role of the subject shall be changed from Unauthenticated User to that role;

(3) after successful re-authentication of the user for a selected role the attribute Role of the subject shall be changed to that role.

6.1.6.6 FMT_SAE.1 Time-limited authorisation

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles

FPT_STM.1 Reliable time stamps

FMT_SAE.1.1 The TSF shall restrict the capability to specify an expiration time for ***Role*** to User Administrator[102]***.***

FMT_SAE.1.2 For each of these security attributes, the TSF shall be able to ***reset the Role to the value assigned according to FMT_MTD.1/RAD, clause (6)*** after the expiration time for the indicated security attribute has passed.

PP application note 30: <Applied> The TSF shall implement means to handle expiration time for the roles whithin a session (i.e. between power-up and power-down of the TOE) which may not necessarily meet the requirements for a reliable time stamp as required by FPT_STM.1. If this security target, the time stamp according to FPT_STM.1 is used to meet FMT_SAE.1.

6.1.6.7 FIA_UID.1 Timing of identification

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UID.1.1 The TSF shall allow

(1) self test according to FPT_TST.1,

(2) identification of the TOE to the user,

(3) **Selected key operations, if explicitly configured for the respective key.**[103]

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

[102] [selection: Administrator, User Administrator]

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

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

6.1.6.8 FIA_UAU.1 Timing of authentication

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.1.1 The TSF shall allow

(1) self test according to FPT_TST.1,

(2) authentication of the TOE to the user,

(3) identification of the user to the TOE and selection of **a set of role** [104]***for authentication,***

(4) **Selected key operations, if explicitly configured for the respective key** [105]

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.

PP application note 31: Clause (2) and (3) in FIA_UAU.1.1 allows mutual identification for mutual authentication, e.g. by exchange of certificates.

6.1.6.9 FIA_UAU.5 Multiple authentication mechanisms

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UAU.5.1 The TSF shall provide

(1) password authentication,

(2) PACE with Generic Mapping with TOE in ICC and user in PCD context with establishment of trusted channel according to FTP_ITC.1,

(3) certificate based Terminal Authentication Version 2 according to section 3.3 in [TR-03110] with the TOE in ICC and user in PCD context,

(4) Terminal Authentication Version 2 with the TOE in ICC context and user in PCD context modified by omitting the verification of the certificate chain,

(5) Chip Authentication Version 2 with establishment of trusted channel according to FTP_ITC.1,

(6) message authentication by MAC verification of received messages

to support user authentication.

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

(1) password authentication shall be used for authentication of human users if enabled according to FMT_MOF.1.1, clause (1),

(2) PACE shall be used for authentication of human users using terminals with establishment of trusted channel according to FTP_ITC.1,

(3) PACE may be used for authentication of IT entities with establishment of trusted channel according to FTP_ITC.1,

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

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

(4) certificate based Terminal Authentication Version 2 may be used for authentication of users which certificate imported as TSF data,

(5) simplified version of Terminal Authentication Version 2 may be used for authentication of identified users associated with known user's public key,

(6) message authentication by MAC verification of received messages shall be used after initial authentication of remote entity according to clauses (2) or (3) for trusted channel according to FTP_ITC.1,

(7) None[106].

6.1.6.10 FIA_UAU.6 Re-authenticating

Hierarchical to: No other components.

Dependencies: No dependencies.

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

(1) changing to a role not selected for the current valid authentication session,

(2) power on or reset,

(3) every message received from entities after establishing trusted channel according to FIA_UAU.5.1, clause (2), (3) or (6),

(4) None[107].

6.1.7 Access control

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

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]

[FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path]

FPT_TDC.1 Inter-TSF basic TSF data consistency

FDP_ITC.2.1/UD The TSF shall enforce the ***Cryptographic Operation SFP*** when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.2.2/UD The TSF shall use the security attributes associated with the imported user data.

FDP_ITC.2.3/UD The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.

FDP_ITC.2.4/UD The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.

FDP_ITC.2.5/UD The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE:

(1) user data imported for encryption according to FCS_COP.1/ED shall be imported with Key identity of the key and the identification of the requested cryptographic operation,

(2) user data imported for encryption according to FCS_COP.1/HEM shall be imported with Key identity of the public key encryption key or key agreement method,

[106] [assignment: additional rules]

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

(3) user data imported for decryption according to FCS_COP.1/HDM shall be imported with Key identity of the asymmetric decryption key, encrypted seed and data integrity checksum,

(4) user data imported for digital signature creation shall be imported with the Key identity of the private signature key,

(5) user data imported for digital signature verification shall be imported with digital signature and Key identity of the public signature key.

PP application note 32: Keys to be used for the cryptographic operation of the imported user data are identified by security attribute Key identity.

6.1.7.2 FDP_ETC.2 Export of user data with security attributes

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_ETC.2.1	The TSF shall enforce the ***Cryptographic Operation SFP*** when exporting user data, controlled under the SFP(s), outside of the TOE.
FDP_ETC.2.2	The TSF shall export the user data with the user data's associated security attributes.
FDP_ETC.2.3	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.
FDP_ETC.2.4	The TSF shall enforce the following rules when user data is exported from the TOE: ***(1) user data exported as ciphertext according to FCS_COP.1/HEM shall be exported with reference to key decryption key, encrypted data encryption key and data integrity checksum,*** ***(2) user data exported as plaintext according to FCS_COP.1/HDM shall be exported only if the MAC verification confirmed the integrity of the ciphertext,*** ***(3) user data exported as signed data according to FCS_COP.1/CDS-ECDSA or FCS_COP.1/CDS-RSA shall be exported with digital signature and Key identity of the used signature-creation key.***

PP application note 33: The TOE imports data to be signed by CSP shall be imported with Key identity of the signature key and exports the signature. In case of internally generated data exported as signed data shall be exported with Key identity of the used key in order to enable identification of the corresponding signature verification key. Note, the TOE may implement more than one signature-creation key for signing internally generated data.

6.1.7.3 FDP_ETC.1 Export of user data without security attributes

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

6.1.7.4 FDP_ACC.1/Oper Subset access control – Cryptographic operation

Hierarchical to: No other components

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/Oper The TSF shall enforce the Cryptographic Operation SFP on

(1) subjects: Crypto-Officer[108], Key Owner, none[109];

(2) objects: operational cryptographic keys, user data;

(3) operations: cryptographic operation

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

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialisation

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

(1) subjects: subjects with security attribute Role Crypto-Officer[110], Key Owner, none[111];

(2) objects:

(a) cryptographic keys with security attributes: Identity of the key, Key entity, Key type, Key usage type, Key access control attributes, Key validity time period;

(b) user data.

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

(1) Subject in Crypto-Officer[112] role is allowed to perform cryptographic operation on cryptographic keys in accordance with their security attributes.

(2) Subject Key Owner is allowed to perform cryptographic operation on user data with cryptographic keys in accordance with the security attribute Key entity, Key type, Key usage type, Key access control attributes and Key validity time period;

(3) None[113].

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

(1) subjects with security attribute Role are allowed to perform cryptographic operation on user data and cryptographic keys with security attributes as shown in the rows of Table 11.

(2) None[114].

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

[109] [assignment: other roles]

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

[111] [assignment: other roles]

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

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

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

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

(1) No subject is allowed to use cryptographic keys by cryptographic operation other than those identified in the security attributes Key usage type and the Key access control attributes;

(2) No subject is allowed to decrypt ciphertext according to FCS_COP.1/HDM if MAC verification fails.

***(3)* <u>**None[115].**</u>**

Access control rules for cryptographic operation:

Security attribute Role of the subject	**Security attribute of the cryptographic key**	**Cryptographic operation referenced by SFR allowed for the subject on user data with the cryptographic key**
<u>**Crypto-Officer, Key Owner[116]**</u>	***Key type: symmetric*** ***Key usage type: Key wrap*** ***Key validity time period:***	***FCS_COP.1/KW***
<u>**Crypto-Officer, Key Owner[117]**</u>	***Key type: symmetric*** ***Key usage type: Key unwrap[118]*** ***Key validity time period:***	***FCS_COP.1/KU***
(any authenticated user)	***Key type: public*** ***Key usage type: ECKA-EG*** ***Key validity time period: as in certificate***	***FCS_COP.1/HEM,*** ***FCS_CKM.1/ECKA-EG***
Key Owner	***Key type: private*** ***Key usage type: ECKA-EG*** ***Key validity time period:***	***FCS_COP.1/HDM*** ***FCS_CKM.5/ECKA-EG***
(any authenticated user)	***Key type: public*** ***Key usage type: RSA_ENC*** ***Key validity time period: as in certificate***	***FCS_COP.1/HEM*** ***FCS_CKM.1/AES_RSA***
Key Owner	***Key type: private*** ***Key usage type: RSA_ENC*** ***Key validity time period: as in certificate***	***FCS_COP.1/HDM*** ***FCS_CKM.5/AES_RSA***
Key Owner	***Key type: private***	***FCS_COP.1/CDS-ECDSA***

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

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

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

[118] Please note that the key wrap key and the key unwrap key are identical, i.e. they have the same key usage type (key wrapping) and point to the same key.

	Key usage type: DS-ECDSA ***Key validity time period:***	
(any authenticated user)	***Key type: public*** ***Key usage type: DS-ECDSA*** ***Key validity time period:***	***FCS_COP.1/VDS-ECDSA***
Key Owner	***Key type: private*** ***Key usage type: DS-RSA*** ***Key validity time period:***	***FCS_COP.1/CDS-RSA***
(any authenticated user)	***Key type: public*** ***Key usage type: DS-RSA*** ***Key validity time period:***	***FCS_COP.1/VDS-RSA***

Table 11: Security attributes and access control

6.1.8 Security Management

6.1.8.1 FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

Dependencies: No dependencies.

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

(1) management of security functions behaviour (FMT_MOF.1),

(2) management of Authentication reference data (FMT_MTD.1/RAD),

(3) management of security attributes of cryptographic keys (FMT_MSA.1/KM, FMT_MSA.2, FMT_MSA.3/KM,

***(4)* None[119].**

6.1.8.2 FMT_SMR.1 Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1 The TSF shall maintain the roles:

- ***Unidentified User,***
- ***Unauthenticated User,***
- ***Key Owner,***
- ***Application component,***
- **Crypto-Officer, User Administrator, Update Agent[120]**
- **no other roles[121].**

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

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

[120] [selection: Administrator, Crypto-Officer, User Administrator, Update Agent]

[121] [selection: [assignment: other roles], no other roles]

PP application note 34: <applied>

6.1.8.3 FMT_MSA.2 Secure security attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MSA.2.1 The TSF shall ensure that only secure values are accepted for security attributes

(1) Key identity,

(2) Key type,

(3) Key usage type,

(4) **None**[122]***.***

The cryptographic keys shall have

(1) Key identity uniquely identifying the key among all keys implemented in the TOE,

(2) exactly one Key type as secret key, private key, public key,

(3) exactly one Key usage type identifying exactly one cryptographic mechanism the key can be used for.

6.1.8.4 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 the function ***password authentication according to FIA_UAU.5.1, clause (1)*** to **User Administrator**[123]***.***

(2) disable the function password authentication according to FIA_UAU.5.1, clause (1) to **User Administrator**[124]***,***

(3) determine the behaviour of the functions trusted channel according to FDP_ITC.1.2 by defining the remote trusted IT products permitted to initiate communication via the trusted channel to **User Administrator**[125]***,***

(4) determine the behaviour of the functions trusted channel according to FDP_ITC.1.3 by defining the entities for which the TSF shall enforce communication via the trusted channel to **User Administrator**[126]***.***

[122] [assignment: additional security attributes]

[123] [selection: Administrator, User Administrator]

[124] [selection: Administrator, User Administrator]

[125] [selection: Administrator, User Administrator]

[126] [selection: Administrator, User Administrator]

PP application note 35: The refinements of FMT_MOF.1.1 in bullets (2) to (4) are made in order to avoid iteration of the component. In case of client-server architecture the applications using the TOE and supporting cryptographically protected trusted channel belong to the entities for which the TSF shall enforce trusted channel according to FDP_ITC.1, cf. FMT_MOF.1.1 in bullet (4).

6.1.9 Protection of the TSF

6.1.9.1 FDP_SDC.1 Stored data confidentiality

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP_SDC.1.1 The TSF shall ensure the confidentiality of the information of the data while it is stored in the **ACL table control area**[127] ***by encryption according to FCS_COP.1/SDE***.

PP application note 36: The memory encryption does not distinguish between user data and TSF data when encrypting memory areas. The refinement extends the SFR to any data in the assigned memory area, which may contain user data, TSF data, software and firmware as TSF implementation.

Developer note: The ACL table control area stores data for the integrity protection of the access control lists.

6.1.9.2 FCS_CKM.1/SDEK Cryptographic key generation – Stored data encryption key generation

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/SDEK The TSF shall generate cryptographic ***stored data encryption*** key in accordance with a specified cryptographic key generation algorithm **as specified in FCS_CKM.1/AES**[128] ***using random bit generation according to FCS_RNG.1*** and specified cryptographic key sizes **256 bit**[129] that meet the following: **[ISO/IEC 18033-3]**[130].

6.1.9.3 FCS_COP.1/SDE Cryptographic operation – Stored data encryption

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/SDE The TSF shall perform ***stored data encryption and decryption*** in accordance with a specified cryptographic algorithm **AES in CBC mode**[131] and cryptographic key sizes **256 bit**[132] that meet the following: **[FIPS197], [NIST-SP800-38A]**[133].

[127] [assignment: memory area]

[128] [assignment: cryptographic key generation algorithm]

[129] [assignment: cryptographic key sizes]

[130] [assignment: list of standards]

[131] [assignment: cryptographic algorithm]

[132] [assignment: cryptographic key sizes]

[133] [assignment: list of standards]

PP application note 37: The generation of data encryption keys according to FCS_CKM.1/SDEK, the encryption and the decryption according to FCS_COP.1/SDE are only used for stored data in the memory areas assigned in FDP_SDC.1.1. They are not a security services of the TOE to the user. If cryptographic algorithm does not provide integrity protection for stored user data the stored data should contain redundancy for detection of data manipulation, e. g. in order to meet FPT_TST.1.2 and FPT_TST.1.3.

6.1.9.4 FRU_FLT.2 Limited fault tolerance

Hierarchical to:	FRU_FLT.1 Degraded fault tolerance
Dependencies:	FPT_FLS.1 Failure with preservation of secure state.
FRU_FLT.2.1	The TSF shall ensure the operation of all the TOE's capabilities when the following failures occur: ***exposure to operating conditions which are not detected according to the requirement Failure with preservation of secure state (FPT_FLS.1).***

Refinement: The term "failure" above means "circumstances". The TOE prevents failures for the "circumstances" defined above.

PP application Note 38: Environmental conditions include but are not limited to power supply, clock, and other external signals (e. g. reset signal) necessary for the TOE operation.

6.1.9.5 FPT_FLS.1 Failure with preservation of secure state

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_FLS.1.1	The TSF shall preserve a secure state when the following types of failures occur: ***(1) self test fails,*** ***(2) exposure to operating conditions which may not be tolerated according to the requirement Limited fault tolerance (FRU_FLT.2) and where therefore a malfunction could occur,*** ***(3) manipulation and physical probing is detected and secure state is reached as response (FPT_PHP.3).***

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

6.1.9.6 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 and after power-on*** to demonstrate the correct operation of **<u>parts of the crypto implementation</u>**[134].
FPT_TST.1.2	The TSF shall provide authorised users with the capability to verify the integrity ***of TSF data.***
FPT_TST.1.3	The TSF shall provide authorised users with the capability to verify the integrity of ***TSF implementation.***
Application note:	The capability to verify the integrity of TSF data and TSF implementation according to FPT_TST.1.2 and FPT_TST.1.3 is possible by using the self-test functionality according to FPT_TST.1.1.

[134] [assignment: parts of TSF]

Developer note: The verification of TSF data includes the integrity of the access control lists. The verification of the TSF implementation uses the LFDBH (Load File Data Block Hash) mechanism of the Javacard (Global Platform).

6.1.9.7 FPT_PHP.3 Resistance to physical attack

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_PHP.3.1 The TSF shall resist

(1) ***physical probing and manipulation*** and (2) ***perturbation and environmental stress*** to the

(1) TSF implementation and

(2) the TSF

by responding automatically such that the SFRs are always enforced.

Refinement: The TSF will implement appropriate mechanisms to continuously counter physical probing and manipulation. In case of platform architecture the resistance to physical attacks shall include the secure execution environment for and the communication with the application component running on the TOE.

PP application note 39: "Automatic response" of protection against physical probing and manipulation means (i) assuming that there might be an attack at any time and (ii) countermeasures are provided at any time. Perturbation and environmental stress to the TSF is relevant when the TOE is running. Note, exploration of information leakage from the TOE like side channels is addressed as bypassability of TSF by the security architecture (cf. ADV_ARC.1.1D and ADV_ARC.1.5C) and shall consider these physical attack scenarios.

6.1.10 Import and verification of Update Code Package

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

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

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or

FDP_IFC.1 Subset information flow control]

[FTP_ITC.1 Inter-TSF trusted channel, or

FTP_TRP.1 Trusted path]

FPT_TDC.1 Inter-TSF basic TSF data consistency

FDP_ITC.2.1/UCP The TSF shall enforce the ***Update SFP*** when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.2.2/UCP The TSF shall use the security attributes associated with the imported user data.

FDP_ITC.2.3/UCP The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.

FDP_ITC.2.4/UCP The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.

FDP_ITC.2.5/UCP The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE:

(1) storing of encrypted Update Code Package only after successful verification of authenticity according to FCS_COP.1/VDSUCP,

(2) decrypts authentic Update Code Package according to FCS_COP.1/DecUCP.

Developer note: The integrity and authenticity of the update mechanism is ensured by the Global Platform mechanism DAP (Data Authentication Pattern). The Update Code Package is uploaded as Encrypted Load File, then verified and afterwards decrypted and persistently stored only after successful verification.

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

Hierarchical to: No other components.

Dependencies: No dependencies.

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

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

(1) the Issuer must be identified and known,

(2) the Version Number must be identified

when interpreting the TSF data from another trusted IT product.

Developer note: The issuer is identified by the successful verification of the Update Code Package digital signature. The version number is checked after verification and decryption.

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

Hierarchical to: No other components.

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

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

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/VDSUCP The TSF shall perform ***verification of the digital signature of the authorized Issuer*** in accordance with a specified cryptographic algorithm **ECDSA**[135] and cryptographic key sizes **256 bit**[136] that meet the following: **RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4]** [137].

PP application note 40: The authorized Issuer is identified in the security attribute of the received Update Code Package and the public key of the authorized Issuer shall be known as TSF data before receiving the Update Code Package. Only public key of the authorized Issuer shall be used for verification of the digital signature of the Update Code Package.

Developer note: The cryptographic mechanism is defined by the Global Platform specification.

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

Hierarchical to: No other components.

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

[135] [assignment: cryptographic algorithm]

[136] [assignment: cryptographic key sizes]

[137] [assignment: list of standards]

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

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/DecUCP The TSF shall perform ***decryption of authentic encrypted Update Code Package*** in accordance with a specified cryptographic algorithm **AES in CBC mode**[138] and cryptographic key sizes **128 bit**[139] that meet the following: **[FIPS197], [NIST SP800-38A]**[140] .

Developer note: The cryptographic mechanism is defined by the Global Platform specification.

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

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/UCP The TSF shall enforce the ***Update SFP*** on

(1) subjects: Update Agent[141];

(2) objects: Update Code Package;

(3) operations: import, store.

Developer note: The update agent is authenticated according to SCP03. Import means loading of the Update Code Package, store means instantiation of the according applet.

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

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1/UCP The TSF shall enforce the Update SFP to objects based on the following:

(1) subjects: Update Agent[142];

(2) objects: Update Code Package with security attributes Issuer and Version Number.

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

(1) Update Agent[143] is allowed to import Update Code Package according to FDP_ITC.2/UCP.

(2) Update Agent[144] is allowed to store Update Code Package if

> ***(a) authenticity is successful verified according to FCS_COP.1/VDSUCP and decrypted according to FCS_COP.1/DecUCP***

[138] [assignment: cryptographic algorithm]

[139] [assignment: cryptographic key sizes]

[140] [assignment: list of standards]

[141] [selection: Administrator, Update Agent]

[142] [selection: Administrator, Update Agent]

[143] [selection: Administrator, Update Agent]

[144] [selection: Administrator, Update Agent]

(b) the Version Number of the Update Code Package is equal or higher than the Version Number of the TSF.

FDP_ACF.1.3/UCP The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **None**[145].

FDP_ACF.1.4/UCP The TSF shall explicitly deny access of subjects to objects based on the following additional rules: **None**[146].

Developer note: The Update Agent is authenticated in two steps: SCP03 for Update Code Package import and a configurable authentication for version check and storage.

6.1.10.7 FDP_RIP.1/UCP Subset residual information protection

Hierarchical to: No other components

Dependencies: No dependencies.

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

Developer note: Deallocation of the resource means manual deletion of the

- unverified and unencrypted loaded Update Code Package or
- verified and unencrypted Update Code Package that is not verified regarding version number and thus not connected to the stored data.

6.2 Security functional requirements from the PP module Time Stamp Service

The following SFRs have been added from [PP0107] to add the time stamp service.

6.2.1 Time Stamp

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

Hierarchical to: FDP_DAU.1 Basic Data Authentication

Dependencies: FIA_UID.1 Timing of identification

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

(a) user data imported according to FDP_ITC.2/UD,

(b) exported audit records according to FMT_MTD.1/Audit clause (1) and FAU_STG.3 clause (1)

with

(1) time stamp of the evidence generation according to FPT_STM.1,

(2) and optionally the key usage counter of the signature key

by means of digital signature generated according to FCS_COP.1/CDS-ECDSA[147] ***and keys holding the dedicated values of the security attributes Key identity that***

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

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

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

indicate key ownership of the TOE sample and Key usage type "Time stamp service".

FDP_DAU.2.2/TS The TSF shall provide

(1) Key Owner

(2) User Administrator[148]

with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

PP-module application note 1:[149] The TSF according to FDP_DAU.2/TS is intended for time stamp service of the TOE for any provided user data and exported audit records. The user data source shall select the security attribute Key usage type "TimeStamp" of the signature key of the time stamp service. The signature key of exported audit records shall be defined according to FMT_MOF.1.1 clause (9). The Key usage counter allows to verify the sequence of signed data e. g. in an audit trail. The verification of the evidence requires a certificate showing the identity of the TOE sample and the key usage type of time stamp service. The format of input data and output data shall meet the BSI TR-03151 [TR-03151].

6.2.2 Access control on time stamp service

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

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/TS The TSF shall enforce the ***Cryptographic Operation SFP*** when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.2.2/TS The TSF shall use the security attributes associated with the imported user data.

FDP_ITC.2.3/TS The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.

FDP_ITC.2.4/TS The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.

FDP_ITC.2.5/TS The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE:

(1) user data imported for time stamp generation to FDP_DAU.2/TS shall be imported with security attributes Key identity of the signature key and Key usage type TimeStamp, and the identification of the requested cryptographic operation.

PP-module application note 2: Keys to be used for the cryptographic operation of the imported user data are identified by security attribute *Key identity*.

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

Hierarchical to: No other components.

[148] [assignment: list of subjects]

[149] The term „PP-module application note" links to the according application note in the PP module time stamp and audit [PP0107].

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]

FDP_ETC.2.1/TS The TSF shall enforce the ***Cryptographic Operation SFP*** when exporting user data, controlled under the SFP(s), outside of the TOE.

FDP_ETC.2.2/TS The TSF shall export the user data with the user data's associated security attributes.

FDP_ETC.2.3/TS The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.

FDP_ETC.2.4TS The TSF shall enforce the following rules when user data is exported from the TOE:

(1) user data exported as time stamped data according to FDP_DAU.2/TS shall be exported with digital signature and Key identity of the used signature-creation key.

PP-module application note 3: In case of internally generated data (e.g. audit records) the exported signed data shall be attributed with the Key identity of the used signature-creation key. Note that the TOE may implement more than one signature-creation key for signing internally generated data.

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

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control

FDP_ACF.1.1/TS The TSF shall enforce the ***Cryptographic Operation SFP*** to objects based on the following:

(1) subjects: subjects with security attribute Role Application Component, **none**[150]

(2) objects: user data .

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

(1) Application Component, **none**[151] ***is allowed to perform cryptographic operation according to FDP_DAU.2/TS on user data with cryptographic keys with Key usage type TimeStamp.***

(2) **None**[152] ***.***

FDP_ACF.1.3/TS The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **None**[153].

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

(1) No subject is allowed to use cryptographic keys by cryptographic operation other than those identified in the security attributes Key usage type and the Key access control attributes;

(2) **None**[154]***.***

[150] [assignment: other roles];

[151] [assignment: other roles]

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

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

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

6.2.3 Security Management

6.2.3.1 FMT_SMF.1/TSA Specification of Management Functions

Hierarchical to: No other components.

Dependencies: No dependencies.

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

(1) management of security functions behaviour FMT_MOF.1/TSA .

6.2.3.2 FMT_SMR.1/TSA Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1/TSA The TSF shall maintain the roles ***additional to those required by FMT_SMR.1 in the Base-PP:*** **Auditor, Timekeeper**[155] ***.***

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

PP-module application note 4: <Applied>

6.2.3.3 FMT_MOF.1/TSA 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/TSA The TSF shall restrict the ability to

(1) modify the behaviour of the function ***adjustment of the internal clock according*** to ***FPT_STM.1 clause (1) to*** **Timekeeper**[156] ***,***

(2) modify the behaviour of the function adjustment of the internal clock according to FPT_STM.1 clause (2) to **Timekeeper**[157] ***,***

(3) determine the behaviour of and modify the behaviour of the functions select the auditable events according to FAU_GEN.1 to **Auditor**[158]***,***

(4) determine the behaviour of and modify the behaviour of the functions automatic export of audit trails according to FAU_STG.3.1 clause (1) to **Auditor**[159]***,***

(5) determine the behaviour of and modify the behaviour of the functions FDP_DAU.2/TS by selection of signature key used to sign exported audit trails to **Auditor**[160] ***.***

PP-module application note 5: The SFR defines additional management of security functions behaviour for new SFR with respect to the Base-PPs. The refinements of FMT_MOF.1.1/TSA in bullets (2) to (5) are made in order to avoid iteration of the component.

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

[156] [selection: Administrator, Timekeeper]

[157] [selection: Administrator, Timekeeper]

[158] [selection: Administrator, Auditor]

[159] [selection: Administrator, Auditor]

[160] [selection: Administrator, Auditor]

6.2.4 Security audit

6.2.4.1 FAU_GEN.1 Audit data generation

Hierarchical to: No other components.

Dependencies: FPT_STM.1 Reliable time stamps

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

a) Start-up and shutdown of the audit functions;

b) All auditable events for the ***not specified*** level of audit; and

c) Discrete adjustment of the real time clock

(1) by automatic adjustment of the clock according to FPT_STM.1.1 clause (2) if selected as auditable event,

(2) by Administrator according to FPT_STM.1.1 clause (1) or (2),

(3) failure of adjustment according to FPT_STM.1.1,

d) other auditable events

(1) Start-up after power-up,

(2) Import of UCP (FDP_ITC.2/UCP),

(3) Authentication failure handling (FIA_AFL.1): the reaching of the threshold for the unsuccessful authentication attempts with claimed Identity of the user,

(4) Generation of (selected types of) signature key pairs (all FCS_CKM.1 instantiations for generation of permanent stored keys)

(5) Execution of (selected types of) cryptographic operation (all FCS_COP.1 instantiations),

(6) Cryptographic key destruction (FCS_CKM.4) of permanent stored keys,

(7) Failure with preservation of secure state (FPT_FLS.1): entering and exiting secure state,

(8) Management of security functions (FMT_MOF.1, FMT_MOF.1/TSA),

(9) None.[161]

(10) no other event[162]

(11) Management of TSF data (FMT_MTD.1/AUDIT): Export, clear and selection of events causing audit data.

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

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and

[161] [assignment: additional specifically defined auditable events]

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, **None.**[163]

PP-module application note 6: The SFR FDP_ITC.2/UCP, FIA_AFL.1, FCS_CKM.1, FCS_COP.1, FCS_CKM.4, FPT_FLS.1 and FMT_MOF.1 are defined in the Base-PP. The SFR FPT_STM.1, FMT_MOF.1/TSA and FMT_MTD.1/Audit are defined in this PP-Module.

6.2.4.2 FMT_MTD.1/Audit Management of TSF data

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

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

(1) manual export,

(2) clear after manual export,

(3) select audited events in FAU_GEN.1,

(4) define the number of audit records causing automatic export and clearing of exported audit records according to FAU_STG.3.1 clause (1),

(5) define the percentage of storage capacity of audit records if actions are assigned in FAU_STG.3.1 clause (2)

the ***audit records*** to **Auditor**[164] ***.***

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

6.2.4.3 FAU_STG.1 Protected audit trail storage

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit data generation

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

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

6.2.4.4 FAU_STG.3 Action in Case of Possible Audit Data Loss

Hierarchical to: No other components.

Dependencies: FAU_STG.1 Protected audit trail storage

FAU_STG.3.1 The TSF shall

(1) ***automatically export audit trails and clear automatically exported audit records*** if the audit trail exceeds an **Auditor**[165] ***defined number of audit records within*** **2-100**[166] ***.***

[163] [assignment: other audit relevant information]

[164] [selection: Auditor, Administrator]

[165] [selection: Administrator, Auditor]

[166] [assignment: pre-defined range]

(2) **The TOE refuses any auditable action**[167] ***if the audit trail exceeds an*** **Auditor**[168] ***settable percentage of storage capacity .***

PP-module application note 8: The ST writer shall perform the open operations in FAU_STG.3.1 element. If the number of audit records in clause (1) is set to 1 then the TSF export each audit record automatically. If the number of number of audit records in clause (1) is set higher than maximum number of audit records in the audit trail then the TSF does not export audit records automatically. The assignment of clause (2) may be “no actions” if an appropriate number of audit records is assigned in clause (1).

6.2.4.5 FPT_STM.1 Reliable time stamps

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps by means of **internal clock with accuracy 10%**[169] **with the ability of adjustment of the clock by the Timekeeper**[170]***.*** [171]

PP-module application note 9: The external trustable source (e.g. signed Network Time Protocol) provides a reliable time source for adjustment of the internal clock. The time intervals of adjustments in clause (2) may be configured by the administrator. Any adjustment or failure of adjustment of the internal clock is an auditable event according to FAU_GEN.1.1.The refinement with selection defines different cases for internal clocks and are therefore printed in bold.

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

6.2.4.6 FPT_TIT.1/Audit TSF data integrity transfer protection – Audit functionality

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or

FDP_IFC.1 Subset information flow control]

[FMT_MTD.1 Management of TSF data or

FMT_MTD.3 Secure TSF data]

FPT_TIT.1.1/Audit The TSF shall enforce ***the Update SFP,*** **Cryptographic Operation SFP** [172] to ***transmit*** TSF data ***audit records*** in a manner protected from ***modification, deletion, insertion and replay*** errors***.***

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

[168] [selection: Administrator, Auditor]

[169] [assignment: approximate deviation]

[170] [selection: Administrator, Timekeeper]

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

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

FPT_TIT.1.2/Audit The TSF shall be able to determine on receipt of TSF data ***time***, whether ***modification*** has occurred.

PP-module application note 10: The Update SFP is enforced by the export of audit records about import of UCP, cf. FAU_GEN.1.1 clause d) (2). The selection of the Key Management SFP or Cryptographic Operation SFP depends of the selection of auditable events of key management, cryptographic operations and adjustment of the internal clock (e. g. used for verification of validity time period) in FAU_GEN.1.1 clause c). The TSF transmits audit records and receives time as TSF data for security audit. The TSF protects the audit records by means of digital signature against modification and by means of time stamps and key usage counter of the signature key as part of the signature against deletion, insertion and replay as required in FPT_TIT.1.1.

6.3 Security assurance requirements

The PP requires the TOE to be evaluated to EAL4 augmented with AVA_VAN.5 and ALC_DVS.2.

6.4 Security requirements rationale

6.4.1 Dependency rationale

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

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

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

SFR	Dependencies of the SFR	SFR components
FCS_CKM.1/SDEK	FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/SDE, FCS_CKM.4
FCS_CKM.1/TCAP	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/TCE, FCS_COP.1/TCM, FCS_CKM.4
FCS_CKM.4	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]	FCS_CKM.1/ECC, FCS_CKM.1/RSA, FCS_CKM.1/ECKA-EG, FCS_CKM.1/AES_RSA, FCS_CKM.1/TCAP, FCS_CKM.1/PACE
FCS_CKM.5/AES	FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/ED FCS_CKM.4
FCS_CKM.5/AES_RSA	FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/HDM with FCS_CKM.5/AES_RSA, FCS_CKM.4
FCS_CKM.5/ECC	FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/CDS-ECDS, FCS_COP.1/VDS-ECDS, FCS_CKM.4
FCS_CKM.5/ECDHE	FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/HEM with FCS_CKM.5/ECDHE, FCS_CKM.4
FCS_CKM.5/ECKA-EG	FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1/HDM with FCS_CKM.5/ECKA-EG, FCS_CKM.4
FCS_COP.1/CDS-ECDSA	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1/ECC, FCS_CKM.4
FCS_COP.1/CDS-RSA	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1/RSA, FCS_CKM.4
FCS_COP.1/DecUCP	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	Import of UCP decryption key as TSF data with confidentiality protection FPT_TCT.1/CK, FCS_COP.1/KU, FCS_CKM.4
FCS_COP.1/ED	[FDP_ITC.1 Import of user data without security attributes, or	FCS_CKM.1/AES, FCS_CKM.4

SFR	Dependencies of the SFR	SFR components
	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/Hash	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	Hash function do not use keys
FCS_COP.1/HDM	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.5/ECKA-EG, FCS_CKM.5/AES_RSA (note deterministic FCS_CKM.5 play the role of randomized FCS_CKM.1) FCS_CKM.4
FCS_COP.1/HEM	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1/ECKA-EG, FCS_CKM.1/AES_RSA, FCS_CKM.5/ECDHE, FCS_CKM.5/AES_RSA, FCS_CKM.4
FCS_COP.1/HMAC	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_RNG.1 generates random strings as HMAC keys FCS_CKM.4
FCS_COP.1/KU	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1/AES FCS_CKM.4
FCS_COP.1/KW	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes,, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1/AES FCS_CKM.4
FCS_COP.1/MAC	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction MT_MSA.2 Secure security attributes	FCS_CKM.1/AES, FCS_CKM.4
FCS_COP.1/SDE	[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/SDEK, FCS_CKM.4

SFR	Dependencies of the SFR	SFR components
	FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	
FCS_COP.1/TCE	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1/TCAP, FCS_CKM.1/PACE, FCS_CKM.4
FCS_COP.1/TCM	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1/TCAP, FCS_CKM.1/PACE, FCS_CKM.4,
FCS_COP.1/VDS-ECDSA	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes	FPT_ISA.1/Cert (note keys are TSF data), FCS_CKM.4
FCS_COP.1/VDS-RSA	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FPT_ISA.1/Cert (note keys are TSF data), FCS_CKM.4
FCS_COP.1/VDSUCP	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	Import of signature verification key of UCP Issuer as TSF data FPT_ISA.1/Cert, FPT_TIT.1/Cert, FCS_CKM.4
FCS_RNG.1	No dependencies	
FDP_ACC.1/KM	FDP_ACF.1 Security attribute based access control	Dependency on FDP_ACF.1 is not fulfilled. Access control to key management functions are specified by FMT_MTD.1/KM because cryptographic keys are TSF data.
FDP_ACC.1/Oper	FDP_ACF.1 Security attribute based access control	FDP_ACF.1/Oper
FDP_ACC.1/UCP	FDP_ACF.1 Security attribute based access control	FDP_ACF.1/UCP
FDP_ACF.1/Oper	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.1/Oper, FMT_MSA.3/KM
FDP_ACF.1/UCP	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.1/UCP, FMT_MSA.3 is not included, because the security attributes of UCP are

SFR	Dependencies of the SFR	SFR components
		imported according to FDP_ITC.2/UCP without default values
FDP_DAU.2/Att	FIA_UID.1 Timing of identification	FIA_UID.1
FDP_DAU.2/Sig	FIA_UID.1 Timing of identification	FIA_UID.1
FDP_ETC.1	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	FDP_ACC.1/Oper
FDP_ETC.2	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	FDP_ACC.1/Oper
FDP_ITC.2/UCP	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1/UCP trusted communication is provided by FCS_COP.1/VDSUCP and FCS_COP.1/DecUCP, FPT_TDC.1/UCP
FDP_ITC.2/UD	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1/Oper trusted communication is provided by FCS_COP.1/HDM and FCS_COP.1/VDS-*, FPT_TDC.1/CK because import of user data is intended for cryptographic operation with key
FDP_RIP.1/UCP	No dependencies	
FDP_SDC.1	No dependencies	
FIA_AFL.1	FIA_UAU.1 Timing of authentication	FIA_UAU.1
FIA_API.1/CA	No dependencies	
FIA_API.1/PACE	No dependencies	
FIA_ATD.1	No dependencies	
FIA_UAU.1	FIA_UID.1 Timing of identification	FIA_UID.1
FIA_UAU.5	No dependencies	
FIA_UAU.6	No dependencies	
FIA_UID.1	No dependencies	
FIA_USB.1	FIA_ATD.1 User attribute definition	FIA_ATD.1
FMT_MOF.1	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMF.1, FMT_SMR.1
FMT_MSA.1/KM	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles	FDP_ACC.1/KM, FDP_ACC.1/Oper, FMT_SMF.1, FMT_SMR.1

SFR	Dependencies of the SFR	SFR components
	FMT_SMF.1 Specification of Management Functions	
FMT_MSA.2	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FDP_ACC.1/KM, FDP_ACC.1/Oper, FMT_MSA.1/KM, FMT_SMR.1
FMT_MSA.3/KM	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1/KM, FMT_SMR.1
FMT_MTD.1/KM	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMF.1, FMT_SMR.1
FMT_MTD.1/RAD	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMF.1, FMT_SMR.1
FMT_MTD.1/RK	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMF.1, FMT_SMR.1
FMT_MTD.3	FMT_MTD.1 Management of TSF data	FMT_MTD.1/RAD
FMT_SAE.1	FMT_SMR.1 Security roles, FPT_STM.1 Reliable time stamps	FMT_SMR.1, dependency on FPT_STM.1 is not fulfilled, cf. to the application note to FMT_STM.1
FMT_SMF.1	No dependencies	
FMT_SMR.1	FIA_UID.1 Timing of identification	FIA_UID.1
FPT_ESA.1/CK	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control][FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1/KM FMT_MTD.1/KM FMT_MSA.1/KM FPT_TDC.1/CK
FPT_FLS.1	No dependencies	
FPT_ISA.1/Cert	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control][FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute	FDP_ACC.1/KM, FMT_MTD.1/RK, FMT_MSA.1/KM, FPT_TDC.1/Cert

SFR	Dependencies of the SFR	SFR components
	value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency	
FPT_ISA.1/CK	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control][FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data] [FMT_MSA.1 Management of security attributes, or FMT_MSA.4 Security attribute value inheritance] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1/KM, FMT_MTD.1/RK, FMT_MTD.1/KM, FMT_MSA.1/KM, FPT_TDC.1/Cert
FPT_PHP.3	No dependencies	
FPT_TCT.1/CK	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control][FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]	FDP_ACC.1/KM, FMT_MTD.1/RK, FMT_MTD.1/KM
FPT_TDC.1/Cert	No dependencies	
FPT_TDC.1/CK	No dependencies	
FPT_TDC.1/UCP	No dependencies	
FPT_TIT.1/Cert	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control][FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]	FDP_ACC.1/KM, FMT_MTD.1/RK
FPT_TIT.1/CK	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control][FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]	FMT_MTD.1/KM
FPT_TST.1	No dependencies	
FRU_FLT.2	FPT_FLS.1 Failure with preservation of secure state	FPT_FLS.1
FTP_ITC.1	No dependencies	
FAU_GEN.1	FPT_STM.1 Reliable time stamps	FPT_STM.1
FAU_STG.1	FAU_GEN.1 Audit data generation	FAU_GEN.1
FAU_STG.3	FAU_STG.1 Protected audit trail storage	FAU_STG.1

SFR	Dependencies of the SFR	SFR components
FDP_ACF.1/TS	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.1/Oper in Base-PP FMT_MSA.3 in Base-PP [PP0104]
FDP_DAU.2/TS	FIA_UID.1 Timing of identification	FIA_UID.1 in Base-PP
FDP_ETC.2/TS	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	FDP_ACC.1/Oper in Base-PP
FDP_ITC.2/TS	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] FPT_TDC.1 Inter-TSF basic TSF data consistency	FDP_ACC.1/Oper, trusted communication is provided by FCS_COP.1/HDM and FCS_COP.1/VDS-*, FPT_TDC.1/CK because import of user data is intended for cryptographic operation with key with appropriate security attribute “TimeStamp”, all these SFR in Base-PP
FMT_MOF.1/TSA	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMR.1/TSA and FMT_SMR.1 in Base-PP, FMT_SMF.1/TSA
FMT_MTD.1/Audit	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FMT_SMR.1/TSA and FMT_SMR.1 in Base-PP, FMT_SMF.1/TSA
FMT_SMF.1/TSA	No dependencies	
FMT_SMR.1/TSA	FIA_UID.1 Timing of identification	FIA_UID.1 in Base-PP
FPT_STM.1	No dependencies	
FPT_TIT.1/Audit	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FMT_MTD.1 Management of TSF data or FMT_MTD.3 Secure TSF data]	FDP_ACC.1/UCP in Base-PP and FDP_ACC.1/KM and FDP_ACC.1/Oper if selected, FMT_MTD.1/Audit

Table 12: Dependency rationale

6.4.2 Security functional requirements rationale

Table 7 traces each SFR back to the security objectives for the TOE. Note that Table 7 includes also the SFRs and security objectives from [PP0107].

	O.I&A	O.AuthentTOE	O.Enc	O.DataAuth	O.RBGS	O.Tchann	O.AccCtrl	O.SecMan	O.PhysProt	O.TST	O.SecpUpCP	O.Audit	O.TimeService
FCS_CKM.1/AES			x	x				x					
FCS_CKM.1/AES_RSA			x	x				x					
FCS_CKM.1/ECC		x	x	x				x					
FCS_CKM.1/ECKA-EG			x	x				x					
FCS_CKM.1/PACE		x				x		x					
FCS_CKM.1/RSA		x	x	x				x					
FCS_CKM.1/SDEK									x				
FCS_CKM.1/TCAP		x				x		x					
FCS_CKM.4			x	x				x					
FCS_CKM.5/AES			x	x				x					
FCS_CKM.5/AES_RSA			x	x				x					
FCS_CKM.5/ECC			x	x				x					
FCS_CKM.5/ECDHE			x	x				x					
FCS_CKM.5/ECKA-EG			x	x				x					
FCS_COP.1/CDS-ECDSA		x		x									
FCS_COP.1/CDS-RSA		x		x									
FCS_COP.1/DecUCP											x		
FCS_COP.1/ED			x					x					
FCS_COP.1/Hash				x				x					
FCS_COP.1/HDM			x	x									
FCS_COP.1/HEM			x	x									
FCS_COP.1/HMAC		x		x									
FCS_COP.1/KU								x					
FCS_COP.1/KW								x					
FCS_COP.1/MAC				x									
FCS_COP.1/SDE									x				
FCS_COP.1/TCE						x							
FCS_COP.1/TCM						x							
FCS_COP.1/VDS-ECDSA				x									
FCS_COP.1/VDS-RSA				x									
FCS_COP.1/VDSUCP											x		

	O.I&A	O.AuthentTOE	O.Enc	O.DataAuth	O.RBGS	O.Tchann	O.AccCtrl	O.SecMan	O.PhysProt	O.TST	O.SecpUpCP	O.Audit	O.TimeService
FCS_RNG.1					x			x					
FDP_ACC.1/KM							x	x					
FDP_ACC.1/Oper							x						
FDP_ACC.1/UCP											x		
FDP_ACF.1/Oper							x						
FDP_ACF.1/UCP											x		
FDP_DAU.2/Att		x											
FDP_DAU.2/Sig				x									
FDP_ETC.1				x									
FDP_ETC.2			x	x									
FDP_ITC.2/UCP											x		
FDP_ITC.2/UD			x	x									
FDP_RIP.1/UCP											x		
FDP_SDC.1									x				
FIA_AFL.1	x												
FIA_API.1/CA	x	x				x							
FIA_API.1/PACE	x	x				x							
FIA_ATD.1	x						x	x					
FIA_UAU.1	x												
FIA_UAU.5	x					x							
FIA_UAU.6	x												
FIA_UID.1	x												
FIA_USB.1	x												
FMT_MOF.1	x					x							
FMT_MSA.1/KM			x	x		x	x	x					
FMT_MSA.2							x	x					
FMT_MSA.3/KM							x	x			x		
FMT_MTD.1/KM								x					
FMT_MTD.1/RAD	x												
FMT_MTD.1/RK	x		x	x				x					
FMT_MTD.3	x												

	O.I&A	O.AuthentTOE	O.Enc	O.DataAuth	O.RBGS	O.Tchann	O.AccCtrl	O.SecMan	O.PhysProt	O.TST	O.SecpUpCP	O.Audit	O.TimeService
FMT_SAE.1	x												
FMT_SMF.1								x					
FMT_SMR.1	x							x					
FPT_ESA.1/CK								x					
FPT_FLS.1									x	x			
FPT_ISA.1/Cert	x			x				x			x		
FPT_ISA.1/CK								x					
FPT_PHP.3									x				
FPT_TCT.1/CK								x			x		
FPT_TDC.1/CK			x	x				x					
FPT_TDC.1/Cert	x		x	x				x					
FPT_TDC.1/UCP											x		
FPT_TIT.1/Cert	x			x				x			x		
FPT_TIT.1/CK								x					
FPT_TST.1										x			
FRU_FLT.2									x				
FTP_ITC.1						x							
FAU_GEN.1												x	
FAU_STG.1												x	
FAU_STG.3												x	
FDP_ACF.1/TS													x
FDP_DAU.2/TS												x	x
FDP_ETC.2/TS													x
FDP_ITC.2/TS													x
FMT_MOF.1/TSA													x
FMT_MTD.1/Audit												x	
FMT_SMF.1/TSA												x	x
FMT_SMR.1/TSA												x	x
FPT_STM.1												x	x
FPT_TIT.1/Audit												x	

Table 13: Security functional requirement rationale

The following part of the chapter demonstrate that the SFRs meet all security objectives for the TOE. Note that the following text was taken from [PP0104] and [PP0107], respectively.

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

- The SFR FIA_ATD.1 lists the security attributes Identity, Authentication reference data and Role belonging to individual users and the SFR FMT_SMR.1 defines the security roles maintained by TSF.
- The SFR FIA_USB.1 requires the TSF to associate the user security attributes Identity and Role with subjects acting on the behalf of that user.
- The SFR FIA_UID.1 defines the TSF-mediated actions allowed on behalf of Unidentified User.
- The SFR FIA_UAU.1 defines the TSF-mediated actions allowed on behalf of Unauthenticated User.
- The SFR FIA_UAU.5 requires the TSF lists the authentication mechanisms and the rules for their application.
- The SFR FIA_API.1/CA and FIA_API.1/PACE require the TSF to authenticate external entities using Chip Authentication and PACE to communication endpoints of trusted channels.
- The SFR FIA_UAU.6 requires the TSF to request re-authentication of users under the listed conditions.
- The SFR FMT_MOF.1 requires the TSF to enable and disable of human user authentication.
- The SFR FMT_MTD.1/RAD and The SFR FMT_MTD.1/RK defines the management function of and the access limitation to authentication mechanisms and their TSF data including the root public keys.
- The SFR FMT_MTD.3 enforce secure values for password mechanisms.
- The SFR FMT_SAE.1 requires the TSF to limit the validity of user authentication and reset the security attribute Role to a values defined by an administrator according to FMT_MTD.1/RAD.
- The SFR FIA_AFL.1 requires the TSF to detect and react on failed authentication attempts.
- The SFR FPT_ISA.1/Cert and FPT_TIT.1/Cert require the TSF to import certificates integrity protected and with their security attributes including those for entity authentication.
- The SFR FPT_TDC.1/Cert requires the TSF to interpret the certificates correctly.

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

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

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

- The SFR FCS_CKM.1/ECC and FCS_CKM.1/RSA require (long term) key generation for the encryption and decryption security service of the TSF.
- The SFR FCS_CKM.1/AES, FCS_CKM.1/AES_RSA, FCS_CKM.5/ECDHE, and FCS_CKM.1/ECKA-EG, require key generation and FCS_CKM.5/AES, FCS_CKM.5/AES_RSA, FCS_CKM.5/ECKA-EG and FCS_CKM.5/ECC require key derivation for encryption and decryption security service of the TSF. Note the keys must be generated or agreed with the appropriate key type for encryption respectively for decryption or in case of symmetric cryptographic mechanisms for both according to FMT_MSA.1/KM.
- The FCS_COP.1/ED requires encryption and decryption as cryptographic operations for the encryption and decryption security service of the TSF.
- The FCS_COP.1/HDM requires hybrid decryption and the SFR FCS_COP.1/HEM requires hybrid encryption and decryption as cryptographic operations for the encryption and decryption security service of the TSF.
- The SFR FDP_ETC.2 require the TSF to export encrypted user data with reference to the key and data integrity checksums for decryption and FDP_ITC.2/UD require import of encrypted user data with reference to decryption key and data integrity checksums for decryption.
- The SFR FCS_CKM.4 requires the TSF to implement secure key destruction.
- The SFR FMT_MTD.1/RK requires the TSF management of root keys for key hierarchy known to the TSF if used for encryption.
- The SFR FPT_TDC.1/Cert requires the TSF to interpret consistently the security attributes of certificates (including those used for encryption and decryption).
- The SFR FPT_TDC.1/CK requires the TSF to interpret consistently the security attributes of keys (including those used for encryption and decryption).

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

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

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

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

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

- The SFR FTP_ITC.1 requires different types of trusted channel depending on the capability of the other endpoint. The cases are defined in Table 10. The remote entity and the TOE may use mutual authentication and key agreement by means of PACE according to FCS_CKM.1/PACE, shall provide integrity protection according to FCS_COP.1/TCM and may support confidentiality of the communication data according to FCS_COP.1/TCE. The cases 3 requires support of trusted channel with mutual authentication by FIA_API.1/CA, FIA_UAU.5, key agreement TCAP according to FCS_CKM.1/TCAP, encryption and MAC data authentication.
- The TOE authenticate themselves according to FIA_API.1/PACE in case of PACE. It authenticates themselves according to FIA_API.1/CA in case of TCAP as Proximity Integrated Circuit Card (PICC).
- The SFR FMT_MOF.1 limits the configuration of the trusted channel according to FTP_ITC.1.3 to an administrator.
- The SFR FMT_MSA.1/KM describe the requirements for management of key security attributes for these mechanisms.

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

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

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

- The SFR FIA_ATD.1 defines the security attributes of individual users including Role which is used to enforce the Key Management SFP.
- The SFR FDP_ACC.1/KM defines subjects, objects and operations of the Key Management SFP.
- The SFR FMT_SMF.1 lists the security management functions provided by the TSF.
- The SFR FMT_SMR.1 lists the security role supported by the TOE especially the administrator and – if supported - Crypto-Officer responsible for key management.

- The SFR FCS_CKM.1/AES, FCS_CKM.1/ECC, FCS_CKM.1/ECKA-EG. FCS_CKM.1/PACE, FCS_CKM.1/RSA, FCS_CKM.1/AES_RSA, FCS_CKM.1/TCAP require the TSF to implement key generation function according to the assigned standards.
- The SFR FCS_CKM.5/ECDHE require the TSF to implement key agreement function according to the assigned standards.
- The SFR FCS_CKM.5/AES and FCS_CKM.5/ECKA-EG require the TSF to implement key derivation function according to the assigned standards.
- The SFR FCS_CKM.1/AES_RSA and FCS_CKM.5/AES_RSA require the TSF to implement AES session key generation function with RSA key encryption respective RSA key decryption and AES key derivation according to the assigned standards.
- The SFR FCS_RNG.1 requires the TSF to implement a random number generator for key generation, key agreement functions and cryptographic operations.
- The SFR FCS_COP.1/ED requires the TSF to provide encryption and decryption according to AES which may be used for key management.
- The SFR FCS_COP.1/Hash requires the TSF to implement cryptographic primitive hash function for key derivation, cf. FCS_CKM.5.
- The SFR FPT_ISA.1/CK requires import and FPT_ESA.1/CK the export of cryptographic keys with security attributes and protection of confidentiality according to SFR FPT_TCT.1/CK and integrity protection according to FPT_TIT.1/CK.
- The SFR FPT_ISA.1/Cert requires import of certificates with security attributes and integrity protection according to FPT_TIT.1/Cert.
- The SFR FPT_TDC.1/Cert requires consistent interpretation of certificate’s content. The SFR FPT_TDC.1/CK requires consistent interpretation of security attributes imported with the key.
- The SFR FCS_COP.1/KW and FCS_COP.1/KU require the TSF key wrapping and unwrapping for key management.
- The SFR FCS_CKM.4 requires the TSF to implement secure key destruction.
- The SFR FMT_MSA.1/KM and FMT_MSA3/KM limit the setting of default values and specification of alternative initial values for security attributes of cryptographic keys to administrators. The SFR FMT_MSA.1/KM prevents modification or deletion of security attributes of keys.
- FMT_MSA.2 enforce secure values for security attributes.
- The SFR FMT_MTD.1/KM and FMT_MTD.1/RK restricts the management of cryptographic keys espacially the import of root public keys to specifically authorized users.

TOE O.TST “Self-test” is directly met by the SFR FPT_TST.1 and FPT_FLS.1. The TSF shall preserve a secure state if self test fails.

The security objective for the TOE O.PhysProt “Physical protection” is met by the directly met by the SFR FPT_PHP.3. The memory encryption required by FDP_SDC.1, FCS_CKM.1/SDEK and FCS_COP.1/SDE provides additional protection against compromise of information in the stored data. The SFR FPT_FLS.1 requires the TSF to preserve a secure state if exposure to operating conditions occurs which may not be tolerated according to the requirement Limited fault tolerance (FRU_FLT.2) or manipulation and physical probing is detected and secure state is reached as response.

The security objective for the TOE O.SecUpCP “Secure download and authorized use of Update Code Package” is met by the following SFR:

- The SFR FDP_ACC.1/UCP and FDP_ACF.1/UCP requires the TSF to provide access control to enforce SFP Update. Note the verification of the authenticity of UCP and decryption of authentic UCP are performed under control of the TSF.
- The SFR FCS_COP.1/VDSUCP requires the verification of digital signature of the Issuer and FCS_COP.1/DecUCP requires decryption of authentic of UCP.
- The SFR FDP_ITC.2/UCP requires the TSF to import UCP as user data with security attributes if the authenticity of UCP is successful verified.
- The SFR FPT_TDC.1/UCP requires the TSF to import consistently the security attributes of the UCP.
- The SFR FMT_MSA.3 requires to provide restrictive initial security attributes to enforce the SFP Update.
- The SFR FDP_RIP.1/UCP requires the TSF to remove the received UCP after unsuccessful verification of its authenticity.
- The UCP signature verification key may be updated according to FPT_ISA.1/Cert with integrity protection according to FPT_TIT.1/Cert.
- The UCP decryption key may be updated with confidentiality protection according to FPT_TCT.1/CK with FCS_COP.1/KU.

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

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

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

- The SFR FAU_GEN.1 requires the TSF to generate the audit records of auditable events.
- The SFR FAU_STG.1 and FAU_STG.3 requires the TSF to protect and to prevent loss of audit records.
- The SFR FMT_MTD.1/Audit restricts the ability to export and to delete exported audit records to an administrator. It prevents undetected deletion of audit records by generation of an audit record about deletion. The export, clear and selection of events causing audit data as management TSF data is an auditable event, cf. FAU_GEN.1, clause (11).
- The SFR FPT_TIT.1/Audit requires the TSF to protect audit records when transmitted and time when imported.
- The SFR FMT_SMF.1/TSA defines the managements functions and FMT_SMR.1/TSA the roles for the audit TSF additional to those defined in the Base-PP.
- The SFR FMT_MOF.1/TSA requires the TSF to provide the capability to define the auditable events in clause (3) and the behaviour of automatic export of audit records in clause (4).

- The SFR FDP_DAU.2/TS requires the TSF to provide the capability to export audit trails signed and time stamped.
- The SFR FPT_TIT.1/Audit defines the TSF data integrity transfer protection for the audit functionality.
- The SFR FPT_STM.1 requires the TSF to provide time stamps being part of the audit records.

6.4.3 Security assurance requirements rationale

The EAL4 was chosen to permit a developer to gain maximum assurance from positive security engineering based on good commercial development practices which, though rigorous, do not require substantial specialist knowledge, skills, and other resources. EAL4 is the highest level at which it is likely to be economically feasible to retrofit to an existing product line. EAL4 is applicable in those circumstances where developers or users require a moderate to high level of independently assured security in conventional commodity TOEs and are prepared to incur sensitive security specific engineering costs.

The augmentation of the component AVA_VAN.5 provides a higher assurance of the security by vulnerability analysis to assess the resistance to penetration attacks performed by an attacker possessing a high attack potential.

Development security is concerned with physical, procedural, personnel and other technical measures that may be used in the development environment to protect the TOE. In the particular case of a cryptographic module the TOE implements security mechanisms in hardware which details about the implementation, (e. g., from design, test and development tools) may make such attacks easier. Therefore, in the case of a cryptographic module, maintaining the confidentiality of the design and protected manufacturing is very important and the strength of the corresponding protection measures shall be balanced with respect to the assumed moderate attack potential. Therefore ALC_DVS.2 was augmented.

7 TOE summary specification (ASE_TSS)

7.1 TOE Security Functionality

7.1.1 TSF_Access: Access Control

This security functionality manages the access to objects (files, directories, data and secrets) stored in the TOE. Access is granted (or denied) in accordance to access rights that depend on appropriate identification and authentication mechanisms.

TSF_Access covers the following SFRs:

- FDP_ACC.1/KM requires that the TSF shall enforce the Key Management SFP on (1) subjects: Crypto-Officer, Key Owner; (2) objects: operational cryptographic keys; (3) operations: key generation, key derivation, key import, key export, key destruction. TSF_Access realizes the appropriate control of the access rights.
- FMT_MSA.1/KM requires that the TSF shall enforce the Key Management SFP and Cryptographic Operation SFP. This is realized by TSF_Access.
- FMT_MSA.3/KM requires that the TSF shall enforce the Key Management SFP, Cryptographic Operation SFP and Update SFP to provide restrictive default values for security attributes that are used to enforce the SFP, and that the TSF shall allow the Crypto-Officer to specify alternative initial values to override the default values when a cryptographic key object or information is created. TSF_Access realizes the appropriate control of the access rights.
- FMT_MTD.1.KM requires that the TSF shall restrict the ability to (1) create according to FCS_CKM.1 the cryptographic keys to the Crypto-Officer; (2) import according to FPT_TCT.1/CK, FPT_TIT.1/CK and FPT_ISA.1/CK the cryptographic keys to Crypto-Officer; (3) export according to FPT_TCT.1/CK, FPT_TIT.1/CK and FPT_ESA.1/CK the cryptographic keys to Crypto-Officer if security attribute of the key allows export; (4) delete according to FCS_CKM.4 the cryptographic keys to Crypto-Officer. TSF_Access implements the according access control.
- FMT_MTD.1/RK requires that the TSF shall restrict the ability to (1) create , modify, clear and delete the root key pair to Crypto-Officer, and (2) import and delete a known as authentic public key of a certification authority in a PKI to Crypto-Officer. TSF_Access realizes the appropriate control of the access rights.
- FPT_TIT.1/Cert requires that the TSF shall enforce the Key Management SFP to receive certificate in a manner protected from modification and insertion errors, and that the TSF shall be able to determine on receipt of certificate, whether modification and insertion has occurred. TSF_Access realizes the appropriate control of the access rights.
- FPT_ISA.1/Cert requires that the TSF shall enforce the Key management SFP when importing certificates , controlled under the SFP, from outside of the TOE, that the TSF shall use the security attributes associated with the imported certificate, ensure that the protocol used provides for the unambiguous association between the security attributes and the certificates received and that interpretation of the security attributes of the imported certificates is as intended by the source of the certificates, and that the TSF shall enforce a defined set of rules when importing certificates controlled under the SFP from outside the TOE. TSF_Access realizes the appropriate control of the access rights.

- FIA_ATD.1 requires that the TSF shall maintain the following list of security attributes belonging to individual users: (1) Identity, (2) Authentication reference data, (3) Role. This is realized by TSF_Auth together wit TSF_Access.
- FMT_MTD.1/RAD requires that the TSF shall restrict the ability to (1) create the initial Authentication reference data of all authorized users to User Administrator, (2) delete the Authentication reference data of an authorized user to User Administrator, (3) modify the Authentication reference data to the corresponding authorized user, (4) create the permanently stored session key of trusted channel as Authentication reference data to User Administrator, (5) define the time in range 1 – (2^32-1) seconds after which the user security attribute Role is reset according to FMT_SAE.1 to User Administrator, and (6) define the value Unauthenticated user to which the security attribute Role shall be reset according to FMT_SAE.1 to User Administrator. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.
- FMT_SAE.1 requires that the TSF shall restrict the capability to specify an expiration time for Role to User Administrator, and that for each of these security attributes, the TSF shall be able to reset the Role to the value assigned according to FMT_MTD.1/RAD, clause (6) after the expiration time for the indicated security attribute has passed. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.
- FIA_UID.1 requires that the TSF shall allow (1) self test according to FPT_TST.1, (2) identification of the TOE to the user, (3) None on behalf of the user to be performed before the user is identified, and that the TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of the Unauthenticated User. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.
- FIA_UAU.1 requires that the TSF shall allow (1) self test according to FPT_TST.1, (2) authentication of the TOE to the user, (3) identification of the user to the TOE and selection of a set of role for authentication, (4) none on behalf of the user to be performed before the user is authenticated, and that the TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.
- FIA_UAU.6 requires that the TSF shall re-authenticate the user under the conditions (1) changing to a role not selected for the current valid authentication session, (2) power on or reset, (3) every message received from entities after establishing trusted channel according to FIA_UAU.5.1 clause (2), (3) or (6), (4) None. This is part of TSF_SecureMessaging, based on TSF_Access and TSF_Auth.
- FDP_ACC.1/Oper requires that the TSF shall enforce the Cryptographic Operation SFP on (1) subjects: Crypto-Officer , Key Owner, none; (2) objects: operational cryptographic keys, user data; (3) operations: cryptographic operation. This is realized by TSF_Admin together with TSF_Access and TSF_Crypto.
- FDP_ACF.1/Oper requires that the TSF shall enforce the Cryptographic Operation SFP to objects based on the following: (1) subjects: subjects with security attribute Role Crypto-Officer , Key Owner, none; (2) objects: (a) cryptographic keys with security attributes: Identity of the key, Key entity, Key type, Key usage type, Key access control attributes, Key validity time period; (b) user data. It also requires that the TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: (1) Subject in Crypto-Officer role is allowed to perform cryptographic operation on cryptographic keys in accordance with their security attributes, (2) Subject Key Owner is allowed to perform cryptographic operation on user data with cryptographic keys in accordance with the security attribute Key entity, Key type, Key usage type,

Key access control attributes and Key validity time period; (3) None. Furthermore, it requires that the TSF shall explicitly authorise access of subjects to objects based on the following additional rules: (1) subjects with security attribute Role are allowed to perform cryptographic operation on user data and cryptographic keys with security attributes as shown in the rows of Table 11, (2) None, and that the TSF shall explicitly deny access of subjects to objects based on the following additional rules: (1) No subject is allowed to use cryptographic keys by cryptographic operation other than those identified in the security attributes Key usage type and the Key access control attributes; (2) No subject is allowed to decrypt ciphertext according to FCS_COP.1/HDM if MAC verification fails; (3) None. This is realized by TSF_Admin together with TSF_Access and TSF_Crypto.

- FDP_ACF.1/TS requires that the TSF shall enforce the Cryptographic Operation SFP to objects based on the following: (1) subjects: subjects with security attribute Role Application Component, (2) objects: user data. It requires that the TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: (1) Application Component, none is allowed to perform cryptographic operation according to FDP_DAU.2/TS on user data with cryptographic keys with Key usage type TimeStamp. (2) None. It further requires that the TSF shall explicitly authorise access of subjects to objects based on the following additional rules: None, and that the TSF shall explicitly deny access of subjects to objects based on the rules: (1) No subject is allowed to use cryptographic keys by cryptographic operation other than those identified in the security attributes Key usage type and the Key access control attributes; (2) None. This is realized by TSF_Admin together with TSF_Access.
- FMT_SMR.1/TSA requires that the TSF shall maintain the roles additional to those required by FMT_SMR.1 in the Base-PP: Auditor,Timekeeper; and the TSF shall be able to associate users with roles. This is realized by TSF_Admin together with TSF_Access.
- FMT_MOF.1/TSA requires that the TSF shall restrict the ability to (1) modify the behaviour of the function adjustment of the internal clock according to FPT_STM.1 clause (1) to Timekeeper, (2) modify the behaviour of the function adjustment of the internal clock according to FPT_STM.1 clause (2) to Timekeeper, (3) determine the behaviour of and modify the behaviour of the functions select the auditable events according to FAU_GEN.1 to Auditor, (4) determine the behaviour of and modify the behaviour of the functions automatic export of audit trails according to FAU_STG.3.1 clause (1) to Auditor, (5) determine the behaviour of and modify the behaviour of the functions FDP_DAU.2/TS by selection of signature key used to sign exported audit trails to Auditor. This is realized by TSF_Admin together with TSF_Access.
- FMT_MTD.1/Audit requires that the TSF shall restrict the ability to (1) manual export, (2) clear after manual export, (3) select audited events in FAU_GEN.1, (4) define the number of audit records causing automatic export and clearing of exported audit records according to FAU_STG.3.1 clause (1), (5) define the percentage of storage capacity of audit records if actions are assigned in FAU_STG.3.1 clause (2) the audit records to Auditor. This is realized by TSF_Admin together with TSF_Access.
- FPT_STM.1 requires that the TSF shall be able to provide reliable time stamps by means of internal clock with accuracy 10% with the ability of adjustment of the clock by the Timekeeper. This is realized by TSF_Admin together with TSF_Access.

7.1.2 TSF_Admin: Administration

This Security Functionality manages the security functional policies as well as the timer and audit storage.

TSF_Admin covers the following SFRs:

- FMT_MSA.3/KM requires that the TSF shall enforce the Key Management SFP, Cryptographic Operation SFP and Update SFP to provide restrictive default values for security attributes that are used to enforce the SFP, and that the TSF shall allow the Crypto-Officer to specify alternative initial values to override the default values when a cryptographic key object or information is created. This is partially realized by TSF_Admin.
- FPT_TCT.1/CK requires that the TSF shall enforce the Key Management SFP by providing the ability to transmit and receive cryptographic keys in a manner protected from unauthorised disclosure according to FCS_COP.1/KW and FCS_COP.1/KU. This is partially realized by TSF_Admin.
- FPT_TIT.1/CK requires that the TSF shall enforce the Key Management SFP to transmit and receive cryptographic keys in a manner protected from modification and insertion errors according to FCS_COP.1/KW, and that the TSF shall be able to determine on receipt of cryptographic keys, whether modification and insertion has occurred according to FCS_COP.1/KU. This is partially realized by TSF_Admin.
- FPT_ISA.1/CK requires that the TSF shall enforce the Key Management SFP when importing cryptographic key, controlled under the SFP, from outside of the TOE, that the TSF shall use the security attributes associated with the imported cryptographic key, that the TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the cryptographic key received, that the TSF shall ensure that interpretation of the security attributes of the imported cryptographic key is as intended by the source of the cryptographic key, and that the TSF shall enforce the following rule when importing cryptographic key controlled under the SFP from outside the TOE: The TSF imports the TSF data in certificates only after successful verification of the validity of the certificate including verification of digital signature of the issuer and validity time period. This is partially realized by TSF_Admin.
- FPT_TDC.1/CK requires that the TSF shall provide the capability to consistently interpret security attributes of the imported cryptographic keys when shared between the TSF and another trusted IT product, and that the TSF shall use the following rules: (1) the TOE reports about conflicts between the Key identity of stored cryptographic keys and cryptographic keys to be imported, (2) the TOE does not change the security attributes Key identity, Key type, Key usage type and Key validity time period of the key being imported when interpreting the imported key data object. This is partially realized by TSF_Admin.
- FPT_ESA.1/CK requires that the TSF shall enforce the Key Management SFP when exporting cryptographic key, controlled under the SFP(s), outside of the TOE, that the TSF shall export the cryptographic key with the cryptographic key's associated security attributes, that the TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported cryptographic key, and that the TSF shall enforce no other rules when a cryptographic key is exported from the TOE. This is partially realized by TSF_Admin.
- FMT_MTD.1/RAD requires that the TSF shall restrict the ability to (1) create the initial Authentication reference data of all authorized users to User Administrator, (2) delete the Authentication reference data of an authorized user to User Administrator, (3) modify the Authentication reference data to the corresponding authorized user, (4) create the permanently stored session key of trusted channel as Authentication reference data to User Administrator, (5) define the time in range 1 – (2^32-1) seconds after which the user security attribute Role is reset according to FMT_SAE.1 to User Administrator, and (6) define the value Unauthenticated user to which the security attribute

Role shall be reset according to FMT_SAE.1 to User Administrator. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.

- FMT_MTD.3 requires that the TSF shall ensure that only secure values are accepted for passwords by enforcing change of initial passwords after first successful authentication of the user to different operational password. This is realized by TSF_Admin together with TSF_Auth.
- FIA_AFL.1 requires that the TSF shall detect when a User Administrator configurable positive integer within 1 - 15 unsuccessful authentication attempts occur related to (1) PACE based authentication, (2) Password based authentication, (3) Cryptographic Entity Authentication; and when the defined number of unsuccessful authentication attempts has been met, the TSF shall delay the next authentication attempt or block the authentication, configurable by the administrator. This is realized by TSF_Admin together with TSF_Auth.
- FIA_USB.1 requires that the TSF shall associate the following user security attributes with subjects acting on the behalf of that user: (1) Identity, (2) Role; it requires that 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; it requires that the TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: (1) after successful identification of the user the attribute Role of the subject shall be changed from Unidentified user to Unauthenticated user; (2) after successful authentication of the user for a selected role the attribute Role of the subject shall be changed from Unauthenticated User to that role; (3) after successful re-authentication of the user for a selected role the attribute Role of the subject shall be changed to that role. This is realized by TSF_Admin together with TSF_Auth.
- FMT_SAE.1 requires that the TSF shall restrict the capability to specify an expiration time for Role to User Administrator, and that for each of these security attributes, the TSF shall be able to reset the Role to the value assigned according to FMT_MTD.1/RAD, clause (6) after the expiration time for the indicated security attribute has passed. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.
- FIA_UID.1 requires that the TSF shall allow (1) self test according to FPT_TST.1, (2) identification of the TOE to the user, (3) None on behalf of the user to be performed before the user is identified, and that the TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of the Unauthenticated User. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.
- FIA_UAU.1 requires that the TSF shall allow (1) self test according to FPT_TST.1, (2) authentication of the TOE to the user, (3) identification of the user to the TOE and selection of a set of role for authentication, (4) none on behalf of the user to be performed before the user is authenticated, and that the TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.
- FDP_ITC.2/UD requires that the TSF shall enforce the Cryptographic Operation SFP when importing user data, controlled under the SFP, from outside of the TOE, that the TSF shall use the security attributes associated with the imported user data, that the TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received, that 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, and that the TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: (1) user data imported

for encryption according to FCS_COP.1/ED shall be imported with Key identity of the key and the identification of the requested cryptographic operation, (2) user data imported for encryption according to FCS_COP.1/HEM shall be imported with Key identity of the public key encryption key or key agreement method, (3) user data imported for decryption according to FCS_COP.1/HDM shall be imported with Key identity of the asymmetric decryption key, encrypted seed and data integrity checksum, (4) user data imported for digital signature creation shall be imported with the Key identity of the private signature key, (5) user data imported for digital signature verification shall be imported with digital signature and Key identity of the public signature key. This is realized by TSF_Admin.

- FDP_ETC.2 requires that the TSF shall enforce the Cryptographic Operation SFP when exporting user data, controlled under the SFP(s), outside of the TOE, that the TSF shall export the user data with the user data's associated security attributes, that the TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data, that the TSF shall enforce the following rules when user data is exported from the TOE: (1) user data exported as ciphertext according to FCS_COP.1/HEM shall be exported with reference to key decryption key, encrypted data encryption key and data integrity checksum, (2) user data exported as plaintext according to FCS_COP.1/HDM shall be exported only if the MAC verification confirmed the integrity of the ciphertext, (3) user data exported as signed data according to FCS_COP.1/CDS-ECDSA or FCS_COP.1/CDS-RSA shall be exported with digital signature and Key identity of the used signature-creation key. This is realized by TSF_Admin.
- FDP_ETC.1 requires that the TSF shall enforce the Cryptographic Operation SFP when exporting user data as plaintext according to FCS_COP.1/HDM, controlled under the SFP(s), outside of the TOE, an that the TSF shall export the successfully MAC verified and decrypted ciphertext as plaintext according to FCS_COP.1/HDM without the user data's associated security attributes. This is realized by TSF_Admin based on functionality from TSF_Crypto.
- FDP_ACC.1/Oper requires that the TSF shall enforce the Cryptographic Operation SFP on (1) subjects: Crypto-Officer , Key Owner, none; (2) objects: operational cryptographic keys, user data; (3) operations: cryptographic operation. This is realized by TSF_Admin together with TSF_Access and TSF_Crypto.
- FDP_ACF.1/Oper requires that the TSF shall enforce the Cryptographic Operation SFP to objects based on the-following: (1) subjects: subjects with security attribute Role Crypto-Officer , Key Owner, none; (2) objects: (a) cryptographic keys with security attributes: Identity of the key, Key entity, Key type, Key usage type, Key access control attributes, Key validity time period; (b) user data. It also requires that the TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: (1) Subject in Crypto-Officer role is allowed to perform cryptographic operation on cryptographic keys in accordance with their security attributes, (2) Subject Key Owner is allowed to perform cryptographic operation on user data with cryptographic keys in accordance with the security attribute Key entity, Key type, Key usage type, Key access control attributes and Key validity time period; (3) None. Furthermore, it requires that the TSF shall explicitly authorise access of subjects to objects based on the following additional rules: (1) subjects with security attribute Role are allowed to perform cryptographic operation on user data and cryptographic keys with security attributes as shown in the rows of Table 11, (2) None, and that the TSF shall explicitly deny access of subjects to objects based on the following additional rules: (1) No subject is allowed to use cryptographic keys by cryptographic operation other than those identified in the security attributes Key usage type and the Key access control attributes; (2)

No subject is allowed to decrypt ciphertext according to FCS_COP.1/HDM if MAC verification fails; (3) None. This is realized by TSF_Admin together with TSF_Access and TSF_Crypto.

- FMT_SMF.1 requires that the TSF shall be capable of performing the following management functions: (1) management of security functions behaviour (FMT_MOF.1), (2) management of Authentication reference data (FMT_MTD.1/RAD), (3) management of security attributes of cryptographic keys (FMT_MSA.1/KM, FMT_MSA.2, FMT_MSA.3/KM, (4) None. This is realized by TSF_Admin together with TSF_Auth and TSF_Crypto.
- FMT_SMR.1 requires that the TSF shall maintain the roles: Unidentified User, Unauthenticated User, Key Owner, Application component, Crypto-Officer, User Administrator, Update Agent, and no other roles, and that the TSF shall be able to associate users with roles. This is realized by TSF_Admin.
- FMT_MSA.2 requires that the TSF shall ensure that only secure values are accepted for security attributes (1) Key identity, (2) Key type, (3) Key usage type, (4) None; and that the cryptographic keys shall have (1) Key identity uniquely identifying the key among all keys implemented in the TOE, (2) exactly one Key type as secret key, private key, public key, (3) exactly one Key usage type identifying exactly one cryptographic mechanism the key can be used for. TSF_Admin together with TSF_Crypto.
- FMT_MOF.1 requires that the TSF shall restrict the ability to (1) enable the function password authentication according to FIA_UAU.5.1, clause (1) to User Administrator, (2) disable the function password authentication according to FIA_UAU.5.1, clause (1) to User Administrator, (3) determine the behaviour of the functions trusted channel according to FDP_ITC.1.2 by defining the remote trusted IT products permitted to initiate communication via the trusted channel to User Administrator, (4) determine the behaviour of the functions trusted channel according to FDP_ITC.1.3 by defining the entities for which the TSF shall enforce communication via the trusted channel to User Administrator. This is realized by TSF_Admin together with TSF_Auth and TSF_SecureMessaging.
- FDP_ITC.2/UCP requires that the TSF shall enforce the Update SFP when importing user data, controlled under the SFP, from outside of the TOE; that the TSF shall use the security attributes associated with the imported user data; that the TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received; that 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; and that the TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: (1) storing of encrypted Update Code Package only after successful verification of authenticity according to FCS_COP.1/VDSUCP, (2) decrypts authentic Update Code Package according to FCS_COP.1/DecUCP. This is realized by TSF_Admin and mechanisms provided by TSF_OS.
- FPT_TDC.1/UCP requires that the TSF shall provide the capability to consistently interpret security attributes Issuer and Version Number when shared between the TSF and another trusted IT product, and that the TSF shall use the following rules: (1) the Issuer must be identified and known, (2) the Version Number must be identified when interpreting the TSF data from another trusted IT product. This is realized by TSF_Admin and mechanisms provided by TSF_OS.
- FCS_COP.1/VDSUCP requires that the TSF shall perform verification of the digital signature of the authorized Issuer in accordance with ECDSA and key size 256 bit that meet RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4]. This is realized by TSF_Admin and mechanisms provided by TSF_OS.

- FCS_COP.1/DecUCP requires that the TSF shall perform decryption of authentic encrypted Update Code Package in accordance with AES in CBC mode and key size 128 bit that meet [FIPS197], [NIST SP800-38A]. This is realized by TSF_Admin and mechanisms provided by TSF_OS.
- FDP_ACC.1/UCP requires that the TSF shall enforce the Update SFP on (1) subjects: Update Agent; (2) objects: Update Code Package; (3) operations: import, store. This is realized by TSF_Admin.
- FDP_ACF.1/UCP requires that the TSF shall enforce the Update SFP to objects based on the following: (1) subjects: Update Agent; (2) objects: Update Code Package with security attributes Issuer and Version Number. It requires that the TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: (1) Update Agent is allowed to import Update Code Package according to FDP_ITC.2/UCP. (2) Update Agent is allowed to store Update Code Package if (a) authenticity is successful verified according to FCS_COP.1/VDSUCP and decrypted according to FCS_COP.1/DecUCP; (b) the Version Number of the Update Code Package is equal or higher than the Version Number of the TSF. It also requires that the TSF shall explicitly authorise access of subjects to objects based on the following additional rules: None and that the TSF shall explicitly deny access of subjects to objects based on the following additional rules: None. This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).
- FDP_RIP.1/UCP requires that the TSF shall ensure that any previous information content of a resource is made unavailable upon the deallocation of the resource after unsuccessful verification of the digital signature of the Issuer according to FCS_COP.1/VDSUCP the following objects: received Update Code Package. This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).
- FDP_DAU.2/TS requires that the TSF shall provide a capability to generate evidence that can be used as a guarantee of the existence at certain point in time, sequence and validity of (a) user data imported according to FDP_ITC.2/UD, (b) exported audit records according to FMT_MTD.1/Audit clause (1) and FAU_STG.3 clause (1) with (1) time stamp of the evidence generation according to FPT_STM.1, (2) and optionally the key usage counter of the signature key by means of digital signature generated according to FCS_COP.1/CDS-ECDSA and keys holding the dedicated values of the security attributes Key identity that indicate key ownership of the TOE sample and Key usage type "Time stamp service". This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).
- FDP_ITC.2/TS requires that the TSF shall enforce the Cryptographic Operation SFP when importing user data, controlled under the SFP, from outside of the TOE, that the TSF shall use the security attributes associated with the imported user data, that the TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received, that 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, and that the TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: (1) user data imported for time stamp generation to FDP_DAU.2/TS shall be imported with security attributes Key identity of the signature key and Key usage type TimeStamp, and the identification of the requested cryptographic operation. This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).
- FDP_ETC.2/TS requires that the TSF shall enforce the Cryptographic Operation SFP when exporting user data, controlled under the SFP(s), outside of the TOE, that the TSF shall export the user data with the user data's associated security attributes, that the TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data, and that the TSF shall enforce the following rules when user data is exported from the TOE: (1) user

data exported as time stamped data according to FDP_DAU.2/TS shall be exported with digital signature and Key identity of the used signature-creation key. This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).

- FDP_ACF.1/TS requires that the TSF shall enforce the Cryptographic Operation SFP to objects based on the following: (1) subjects: subjects with security attribute Role Application Component, (2) objects: user data. It requires that the TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: (1) Application Component, none is allowed to perform cryptographic operation according to FDP_DAU.2/TS on user data with cryptographic keys with Key usage type TimeStamp. (2) None. It further requires that the TSF shall explicitly authorise access of subjects to objects based on the following additional rules: None, and that the TSF shall explicitly deny access of subjects to objects based on the rules: (1) No subject is allowed to use cryptographic keys by cryptographic operation other than those identified in the security attributes Key usage type and the Key access control attributes; (2) None. This is realized by TSF_Admin together with TSF_Access.
- FMT_SMF.1/TSA requires that the TSF shall be capable of performing the following management functions: (1) management of security functions behaviour FMT_MOF.1/TSA. This is realized by TSF_Admin.
- FMT_SMR.1/TSA requires that the TSF shall maintain the roles additional to those required by FMT_SMR.1 in the Base-PP: Auditor,Timekeeper; and the TSF shall be able to associate users with roles. This is realized by TSF_Admin together with TSF_Access.
- FMT_MOF.1/TSA requires that the TSF shall restrict the ability to (1) modify the behaviour of the function adjustment of the internal clock according to FPT_STM.1 clause (1) to Timekeeper, (2) modify the behaviour of the function adjustment of the internal clock according to FPT_STM.1 clause (2) to Timekeeper, (3) determine the behaviour of and modify the behaviour of the functions select the auditable events according to FAU_GEN.1 to Auditor, (4) determine the behaviour of and modify the behaviour of the functions automatic export of audit trails according to FAU_STG.3.1 clause (1) to Auditor, (5) determine the behaviour of and modify the behaviour of the functions FDP_DAU.2/TS by selection of signature key used to sign exported audit trails to Auditor. This is realized by TSF_Admin together with TSF_Access.
- FAU_GEN.1 requires that the TSF shall be able to generate an audit record of the following auditable events:
    - a) Start-up and shutdown of the audit functions;
    - b) All auditable events for the not specified level of audit; and
    - c) Discrete adjustment of the real time clock
        - (1) by automatic adjustment of the clock according to FPT_STM.1.1 clause (2) if selected as auditable event,
        - (2) by Administrator according to FPT_STM.1.1 clause (1) or (2),
        - (3) failure of adjustment according to FPT_STM.1.1,
    - d) other auditable events
        - (1) Start-up after power-up,
        - (2) Import of UCP (FDP_ITC.2/UCP),
        - (3) Authentication failure handling (FIA_AFL.1): the reaching of the threshold for the unsuccessful authentication attempts with claimed Identity of the user,
        - (4) Generation of (selected types of) signature key pairs (all FCS_CKM.1 instantiations for generation of permanent stored keys)

  - (5) Execution of (selected types of) cryptographic operation (all FCS_COP.1 instantiations),
  - (6) Cryptographic key destruction (FCS_CKM.4) of permanent stored keys,
  - (7) Failure with preservation of secure state (FPT_FLS.1): entering and exiting secure state,
  - (8) Management of security functions (FMT_MOF.1, FMT_MOF.1/TSA),
  - (9) None.
  - (10) no other event
  - (11) Management of TSF data (FMT_MTD.1/AUDIT): Export, clear and selection of events causing audit data.

  It also requires that 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, 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, None. This is realized by TSF_Admin.

- FMT_MTD.1/Audit requires that the TSF shall restrict the ability to (1) manual export, (2) clear after manual export, (3) select audited events in FAU_GEN.1, (4) define the number of audit records causing automatic export and clearing of exported audit records according to FAU_STG.3.1 clause (1), (5) define the percentage of storage capacity of audit records if actions are assigned in FAU_STG.3.1 clause (2) the audit records to Auditor. This is realized by TSF_Admin together with TSF_Access.
- FAU_STG.1requires that the TSF shall protect the stored audit records in the audit trail from unauthorised deletion, and that the TSF shall be able to prevent unauthorised modifications to the stored audit records in the audit trail. This is realized by TSF_Admin.
- FAU_STG.3 requires that the TSF shall (1) automatically export audit trails and clear automatically exported audit records if the audit trail exceeds an Auditor defined number of audit records within 2-100, and that (2) the TOE refuses any auditable action if the audit trail exceeds an Auditor settable percentage of storage capacity. This is realized by TSF_Admin.
- FPT_STM.1 requires that the TSF shall be able to provide reliable time stamps by means of internal clock with accuracy 10% with the ability of adjustment of the clock by the Timekeeper. This is realized by TSF_Admin together with TSF_Access.
- FPT_TIT.1/Audit requires that the TSF shall enforce the Update SFP, Cryptographic Operation SFP to transmit TSF data audit records in a manner protected from modification, deletion, insertion and replay errors, and that the TSF shall be able to determine on receipt of TSF data time, whether modification has occurred. This is realized by TSF_Admin.

7.1.3 TSF_Secret: Secret key management

This Security Functionality ensures secure management of secrets such as cryptographic keys. This covers secure key storage, access to keys as well as secure key deletion.

TSF_Secret covers the following SFRs:

- FDP_SDC.1 requires that the TSF shall ensure the confidentiality of the information of the data while it is stored in the user chosen memory area by encryption according to FCS_COP.1/SDE. This is realized by TSF_Secret based on TSF_Crypto (itself based on TSF_OS).
- FCS_CKM.1/SDEK requires that the TSF shall generate cryptographic stored data encryption key in accordance with a specified cryptographic key generation algorithm as specified in FCS_CKM.1/AES

using random bit generation according to FCS_RNG.1 and key sizes 128, 256 bit that meet [ISO/IEC 18033-3]. This is realized by TSF_Secret based on TSF_Crypto (itself based on TSF_OS).

- FCS_COP.1.1/SDE requires that the TSF shall perform stored data encryption and decryption in accordance with AES in CBC mode and key sizes 128, 256 bit that meet: [FIPS197], [NIST-SP800-38A]. This is realized by TSF_Secret based on TSF_Crypto (itself based on TSF_OS).

7.1.4 TSF_Crypto: Cryptographic operations

This Security Functionality performs high level cryptographic operations. The implementation is based on the Security Functionalities provided by TSF_OS.

TSF_Crypto covers the following SFRs:

- FCS_COP.1/Hash requires that the TSF shall perform hash generation in accordance with a specified cryptographic algorithm SHA-256, SHA-384, SHA-512. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FPT_TIT.1/Cert requires that the TSF shall enforce the Key Management SFP to receive certificate in a manner protected from modification and insertion errors, and that the TSF shall be able to determine on receipt of certificate, whether modification and insertion has occurred. This is partially realized by TSF_Crypto.
- FPT_ISA.1/Cert requires that the TSF shall enforce the Key management SFP when importing certificates , controlled under the SFP, from outside of the TOE, that the TSF shall use the security attributes associated with the imported certificate, ensure that the protocol used provides for the unambiguous association between the security attributes and the certificates received and that interpretation of the security attributes of the imported certificates is as intended by the source of the certificates, and that the TSF shall enforce a defined set of rules when importing certificates controlled under the SFP from outside the TOE. This cryptographic functionality is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FPT_TDC.1/Cert requires that the TSF shall provide the capability to consistently interpret security attributes of cryptographic keys in the certificate and identity of the certificate issuer when shared between the TSF and another trusted IT product, and that the TSF shall use the following rules: (1) the TOE reports about conflicts between the Key identity of stored cryptographic keys and cryptographic keys to be imported, (2) the TOE does not change the security attributes Key identity, Key entity, Key type, Key usage type and Key validity time period of public key being imported from the certificate, (3) the identity of the certificate issuer shall meet the identity of the signer of the certificate when interpreting the certificate from a trust center. This is realized by TSF_Crypto.
- FCS_RNG.1 requires that the TSF shall provide a deterministic random number generator that implements a set of properties to fulfill the definition of [AIS20]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_CKM.1/AES requires that the TSF shall generate cryptographic AES key in accordance with a specified cryptographic key generation algorithm AES and key size 128 bits, 256 bits that meet [ISO18033-3]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_CKM.5/AES requires that the TSF shall derive AES keys from a byte array of variable length in accordance with a specified cryptographic key derivation algorithms using bit strings derived from input parameters with KDF and specified cryptographic key sizes 128 bits, 256 bits that meet [NIST-

SP800-56C]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).

- FCS_CKM.1/ECC requires that the TSF shall generate elliptic curve key pairs in accordance with a specified cryptographic key generation algorithm and specified cryptographic key sizes 256, 384, 512 and 521 bit, respectively that meet [RFC5639], TR-03111, section 4.1.3 [TR03111], or FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 and D.1.2.5 [FIPS186-4], respectively. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_CKM.5/ECC requires that the TSF shall derive cryptographic elliptic curve key pair from a byte array of variable length in accordance with a specified cryptographic key derivation algorithm using bit string derived from input parameters with X9.63 Key Derivation Function according to [TR03111], page 27 and specified cryptographic key sizes 256, 384, 512 bit that meet the following: RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR03111], or FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS186-4], respectively , and [TR03111]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_CKM.1/RSA requires that the TSF shall generate cryptographic RSA key pair in accordance with a specified cryptographic key generation algorithm and specified cryptographic key sizes from 2000 bit to 4096 bit in one bit steps that meet the following: PKCS #1 v2.2 [PKCS1]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_CKM.5/ECDHE requires that the TSF shall derive cryptographic ephemeral keys for data encryption and MAC with AES-128, AES-256 from an agreed shared secret in accordance with a specified cryptographic key derivation algorithm (Elliptic Curve Diffie-Hellman ephemeral key agreement) with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384 and 256-bit random ECP group, 384-bit random ECP group, brainpoolP256r1, brainpoolP384r1, brainpoolP512r1 with a key derivation from the shared secret X9.63 Key Derivation Function and specified cryptographic key sizes 128 bits, 256 bits that meet the following: TR-03111 [TR-03111]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_CKM.1/ECKA-EG requires that the TSF shall generate an ephemeral cryptographic elliptic curve key pair for ECKGA-EG [TR-03111], sender role in accordance with a specified cryptographic key generation algorithm with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384 and specified cryptographic key sizes 256 bit, 384 bit, 512 bit that meet RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_CKM.5/ECKA-EG requires that the TSF shall derive cryptographic data encryption key and MAC keys for AES 128, AES-256 from a private and a public ECC key in accordance with a specified cryptographic key derivation algorithm ECKGA-EG [TR-03111] with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384 and X9.63 Key Derivation Function and specified cryptographic symmetric key sizes 128 bits, 256 bits that meet TR-03111 [TR-03111], chapter 4.3.2.2. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_CKM.1/AES_RSA requires that the TSF shall generate and encrypt seed, derive cryptographic keys from seed for data encryption and MAC with AES-128, AES-256 in accordance with a specified cryptographic key generation algorithm: X9.63 Key Derivation Function [ANSI-X9.63] and RSA EME-OAEP [PKCS#1] and specified cryptographic symmetric key sizes 128 bits, 256 bits that meet ISO/IEC 18033-3 [ISO/IEC 18033-3], PKCS #1 v2.2 [PKCS#1]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).

- FCS_CKM.5/AES_RSA requires that the TSF shall derive cryptographic data encryption key and MAC key for AES-128, AES-256 from decrypted RSA encrypted seed in accordance with RSA EME-OAEP [PKCS#1] and X9.63 [ANSI-X9.63] Key Derivation Function and specified cryptographic symmetric key sizes 128 bits, 256 bits that meet ISO/IEC 14888-2 [ISO/IEC 14888-2]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_CKM.4 requires that the TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method: overwriting the keys. This is partially realized by TSF_Crypto (for keys not directly initiated as cryptographic keys in TSF_OS).
- FCS_COP.1/KW requires that the TSF shall perform key wrap in accordance with AES-Keywrap KWP and cryptographic key sizes of the key encryption key 128 bit that meet NIST SP800-38F [NIST-SP800-38F]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_COP.1/KU requires that the TSF shall perform key unwrap in accordance with AES-Keywrap KWP and cryptographic key sizes of the key encryption key of 128 bit that meet NIST SP800-38F [NIST-SP800-38F]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FPT_TCT.1/CK requires that the TSF shall enforce the Key Management SFP by providing the ability to transmit and receive cryptographic keys in a manner protected from unauthorised dis-closure according to FCS_COP.1/KW and FCS_COP.1/KU. This is partially realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FPT_TIT.1/CK requires that the TSF shall enforce the Key Management SFP to transmit and receive cryptographic keys in a manner protected from modification and insertion errors according to FCS_COP.1/KW, and that the TSF shall be able to determine on receipt of cryptographic keys, whether modification and insertion has occurred according to FCS_COP.1/KU. This is partially realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FPT_ISA.1/CK requires that the TSF shall enforce the Key Management SFP when importing cryptographic key, controlled under the SFP, from outside of the TOE, that the TSF shall use the security attributes associated with the imported cryptographic key, that the TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the cryptographic key received, that the TSF shall ensure that interpretation of the security attributes of the imported cryptographic key is as intended by the source of the cryptographic key, and that the TSF shall enforce the following rule when importing cryptographic key controlled under the SFP from outside the TOE: The TSF imports the TSF data in certificates only after successful verification of the validity of the certificate including verification of digital signature of the issuer and validity time period. This is partially realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FPT_TDC.1/CK requires that the TSF shall provide the capability to consistently interpret security attributes of the imported cryptographic keys when shared between the TSF and another trusted IT product, and that the TSF shall use the following rules: (1) the TOE reports about conflicts between the Key identity of stored cryptographic keys and cryptographic keys to be imported, (2) the TOE does not change the security attributes Key identity, Key type, Key usage type and Key validity time period of the key being imported when interpreting the imported key data object. This is partially realized by TSF_Crypto.

- FPT_ESA.1/CK requires that the TSF shall enforce the Key Management SFP when exporting cryptographic key, controlled under the SFP(s), outside of the TOE, that the TSF shall export the cryptographic key with the cryptographic key's associated security attributes, that the TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported cryptographic key, and that the TSF shall enforce no other rules when a cryptographic key is exported from the TOE. This is partially realized by TSF_Crypto.
- FCS_COP.1/ED requires that the TSF shall perform data encryption and decryption in accordance with symmetric data encryption according to AES-128 and AES-256 in CBC and no other mode and cryptographic key size 128 bits, 256 bits that meet NIST SP800-38A [NIST-SP800-38A], ISO/IEC 18033-3 [ISO/IEC 18033-3], ISO/IEC 10116 [ISO/IEC 10116]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_COP.1/HEM requires that the TSF shall perform hybrid data encryption and MAC calculation in accordance with a specified cryptographic algorithm asymmetric key encryption according to FCS_CKM.1/ECKA-EG , symmetric data encryption according to AES-128, AES-256 [FIPS197] in CBC [NIST-SP800-38A] mode with CMAC [NIST-SP800-38B] calculation and cryptographic symmetric key sizes 128 bits, 256 bits that meet the referenced standards above. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_COP.1/HDM requires that the TSF shall perform hybrid MAC verification and data decryption in accordance with a specified cryptographic algorithm asymmetric key decryption according to FCS_CKM.5/ECKA-EG , verification of CMAC [NIST-SP800-38B] and symmetric data decryption according to AES with AES-128, AES-256 [FIPS197] in mode CBC [NIST-SP800-38A] and cryptographic symmetric key sizes 128 bits, 256 bits that meet the referenced standards above. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_COP.1/MAC requires that the TSF shall perform MAC generation and verification in accordance with a specified cryptographic algorithm AES-128 and AES-256 [FIPS197] CMAC [NIST-SP800-38B] and cryptographic key sizes 128 bits, 256 bits that meet the referenced standards above. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_COP.1/HMAC requires that the TSF shall perform HMAC generation and verification in accordance with a specified cryptographic algorithm HMAC-SHA256 and cryptographic key sizes 256 bit that meet RFC2104 [RFC2104], ISO/IEC 9797-2 [ISO/IEC 9797-2]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_COP.1/CDS-ECDSA requires that the TSF shall perform signature-creation in accordance with a specified cryptographic algorithm ECDSA with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384, Curve P-521 and cryptographic key sizes 256, 384, 512 and 521 bit that meet RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_COP.1/VDS-ECDSA requires that the TSF shall perform signature-verification in accordance with a specified cryptographic algorithm ECDSA with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384, Curve P-521 and cryptographic key sizes 256, 384, 512 and 521 bit that meet RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FCS_COP.1/CDS-RSA requires that the TSF shall perform signature-creation in accordance with a specified cryptographic algorithm RSA and EMSA-PSS and cryptographic key sizes 2000-4096 bit

that meet ISO/IEC 14888-2 [ISO/IEC 14888-2], PKCS #1, v2.2 [PKCS#1]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).

- FCS_COP.1/VDS-RSA requires that the TSF shall perform signature-verification in accordance with a specified cryptographic algorithm RSA and EMSA-PSS and cryptographic key sizes 2000-4096 bit that meet the following: ISO/IEC 14888-2 [ISO/IEC 14888-2], PKCS #1, v2.2 [PKCS#1]. This is realized by TSF_Crypto based on cryptographic functionality of the platform (TSF_OS).
- FDP_DAU.2/Sig requires that the TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of user data imported according to FDP_ITC.2/UD by means of FCS_COP.1/CDS-RSA, FCS_COP.1/CDS-ECDSA and keys holding the security attributes Key identity assigned to the guarantor and Key usage type “Signature service”, and that the TSF shall provide external entities with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FIA_API.1/PACE requires that the TSF shall provide PACE in ICC role to prove the identity of the TOE to an external entity and establishing a trusted channel according to FTP_ITC.1 case 1 or 2. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FIA_API.1/CA requires that the TSF shall provide Chip Authentication Version 2 according to [TR03110] section 3.4 to prove the identity of the TOE to an external entity and establishing a trusted channel according to FTP_ITC.1 case 3. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS (and TSF_Auth).
- FDP_DAU.2/Att requires that the TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of attestation data by means of FCS_COP.1/CDS-ECDSA and keys holding the security attributes Key identity assigned to the TOE sample and Key usage type “Attestation”, and that the TSF shall provide external entities with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS (and TSF_Auth).
- FTP_ITC.1 requires that the TSF shall provide a communication channel between TSF and another trusted IT product that is logically separated from other communication channels and provides assured identification of its end points; authentication of TOE and remote entity according to the case in Table 10 and protection of the channel data from modification or disclosure according to the case in Table 10 as required by cryptographic operation according to the case in Table 10; in addition, the TSF shall permit the remote trusted IT product determined according to FMT_MOF.1.1 clause (3) to initiate communication via the trusted channel and the TSF shall initiate communication via the trusted channel for communication with entities defined according to FMT_MOF.1 clause (4). This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.1/PACE requires that the TSF shall generate cryptographic keys for MAC for FCS_COP.1/TCM and encryption keys for FCS_COP.1/TCE in accordance with a specified cryptographic key agreement algorithm PACE with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384 and Generic Mapping in ICC role and specified cryptographic key sizes 128, 256 bits that meet ICAO Doc9303, Part 11, section 4.4 [ICAO Doc9303]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.1/TCAP requires that the TSF shall generate cryptographic keys for encryption according to FCS_COP.1/TCE and MAC according to FCS_COP.1/TCM in accordance with Terminal Authentication version 2 and Chip Authentication Version 2 and specified cryptographic key sizes 128 bits, 256

bits that meet BSI TR-03110 [TR-03110], section 3.3 and 3.4. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS. It is part of TSF_Auth and of TSF_Secure Messaging.

- FCS_COP.1/TCE requires that the TSF shall perform encryption and decryption in accordance with AES in CBC [NIST-SP800-38A] mode and cryptographic key sizes 128 bits, 256 bits that meet [FIPS197]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS. It is part of TSF_Secure Messaging.
- FCS_COP.1/TCM requires that the TSF shall perform MAC calculation and MAC verification in accordance with AES CMAC [NIST-SP800-38B] and cryptographic key sizes 128 bits, 256 bits that meet [FIPS197]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS. It is part of TSF_Secure Messaging.
- FIA_UAU.5 requires that the TSF shall provide (1) password authentication, (2) PACE with Generic Mapping with TOE in ICC and user in PCD context with establishment of trusted channel according to FTP_ITC.1, (3) certificate based Terminal Authentication Version 2 according to section 3.3 in [TR-03110] with the TOE in ICC and user in PCD context, (4) Terminal Authentication Version 2 with the TOE in ICC context and user in PCD context modified by omitting the verification of the certificate chain, (5) Chip Authentication Version 2 with establishment of trusted channel according to FTP_ITC.1, (6) message authentication by MAC verification of received message to support user authentication. It also requires that the TSF shall authenticate any user's claimed identity according to the rules (1) password authentication shall be used for authentication of human users if enabled according to FMT_MOF.1.1, clause (1), (2) PACE shall be used for authentication of human users using terminals with establishment of trusted channel according to FTP_ITC.1, (3) PACE may be used for authentication of IT entities with establishment of trusted channel according to FTP_ITC.1, (4) certificate based Terminal Authentication Version 2 may be used for authentication of users which certificate imported as TSF data, (5) simplified version of Terminal Authentication Version 2 may be used for authentication of identified users associated with known user's public key, (6) message authentication by MAC verification of received messages shall be used after initial authentication of remote entity according to clauses (2) or (3) for trusted channel according to FTP_ITC.1, (7) None. This is realized by TSF_Auth based on functionality from TSF_Crypto, which itself is based on cryptographic functionality of TSF_OS.
- FDP_ETC.1 requires that the TSF shall enforce the Cryptographic Operation SFP when exporting user data as plaintext according to FCS_COP.1/HDM, controlled under the SFP(s), outside of the TOE, an that the TSF shall export the successfully MAC verified and decrypted ciphertext as plaintext according to FCS_COP.1/HDM without the user data's associated security attributes. This is realized by TSF_Admin based on functionality from TSF_Crypto.
- FDP_ACC.1/Oper requires that the TSF shall enforce the Cryptographic Operation SFP on (1) subjects: Crypto-Officer , Key Owner, none; (2) objects: operational cryptographic keys, user data; (3) operations: cryptographic operation. This is realized by TSF_Admin together with TSF_Access and TSF_Crypto.
- FDP_ACF.1/Oper requires that the TSF shall enforce the Cryptographic Operation SFP to objects based on the following: (1) subjects: subjects with security attribute Role Crypto-Officer , Key Owner, none; (2) objects: (a) cryptographic keys with security attributes: Identity of the key, Key entity, Key type, Key usage type, Key access control attributes, Key validity time period; (b) user data. It also requires that the TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: (1) Subject in Crypto-Officer role is allowed to perform cryptographic operation on cryptographic keys in accordance with their security

attributes, (2) Subject Key Owner is allowed to perform cryptographic operation on user data with cryptographic keys in accordance with the security attribute Key entity, Key type, Key usage type, Key access control attributes and Key validity time period; (3) None. Furthermore, it requires that the TSF shall explicitly authorise access of subjects to objects based on the following additional rules: (1) subjects with security attribute Role are allowed to perform cryptographic operation on user data and cryptographic keys with security attributes as shown in the rows of Table 11, (2) None, and that the TSF shall explicitly deny access of subjects to objects based on the following additional rules: (1) No subject is allowed to use cryptographic keys by cryptographic operation other than those identified in the security attributes Key usage type and the Key access control attributes; (2) No subject is allowed to decrypt ciphertext according to FCS_COP.1/HDM if MAC verification fails; (3) None. This is realized by TSF_Admin together with TSF_Access and TSF_Crypto.

- FMT_SMF.1 requires that the TSF shall be capable of performing the following management functions: (1) management of security functions behaviour (FMT_MOF.1), (2) management of Authentication reference data (FMT_MTD.1/RAD), (3) management of security attributes of cryptographic keys (FMT_MSA.1/KM, FMT_MSA.2, FMT_MSA.3/KM, (4) None. This is realized by TSF_Admin together with TSF_Auth and TSF_Crypto.
- FMT_MSA.2 requires that the TSF shall ensure that only secure values are accepted for security attributes (1) Key identity, (2) Key type, (3) Key usage type, (4) None; and that the cryptographic keys shall have (1) Key identity uniquely identifying the key among all keys implemented in the TOE, (2) exactly one Key type as secret key, private key, public key, (3) exactly one Key usage type identifying exactly one cryptographic mechanism the key can be used for. TSF_Admin together with TSF_Crypto.
- FDP_SDC.1 requires that the TSF shall ensure the confidentiality of the information of the data while it is stored in the user chosen memory area by encryption according to FCS_COP.1/SDE. This is realized by TSF_Secret based on TSF_Crypto (itself based on TSF_OS).
- FCS_CKM.1/SDEK requires that the TSF shall generate cryptographic stored data encryption key in accordance with a specified cryptographic key generation algorithm as specified in FCS_CKM.1/AES using random bit generation according to FCS_RNG.1 and key sizes 128, 256 bit that meet [ISO/IEC 18033-3]. This is realized by TSF_Secret based on TSF_Crypto (itself based on TSF_OS).
- FCS_COP.1.1/SDE requires that the TSF shall perform stored data encryption and decryption in accordance with AES in CBC mode and key sizes 128, 256 bit that meet: [FIPS197], [NIST-SP800-38A]. This is realized by TSF_Secret based on TSF_Crypto (itself based on TSF_OS).
- FDP_ACF.1/UCP requires that the TSF shall enforce the Update SFP to objects based on the following: (1) subjects: Update Agent; (2) objects: Update Code Package with security attributes Issuer and Version Number. It requires that the TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: (1) Update Agent is allowed to import Update Code Package according to FDP_ITC.2/UCP. (2) Update Agent is allowed to store Update Code Package if (a) authenticity is successful verified according to FCS_COP.1/VDSUCP and decrypted according to FCS_COP.1/DecUCP; (b) the Version Number of the Update Code Package is equal or higher than the Version Number of the TSF. It also requires that the TSF shall explicitly authorise access of subjects to objects based on the following additional rules: None and that the TSF shall explicitly deny access of subjects to objects based on the following additional rules: None. This is realized by TSF_Admin together wit TSF_Crypto (based on TSF_OS).
- FDP_RIP.1/UCP requires that the TSF shall ensure that any previous information content of a resource is made unavailable upon the deallocation of the resource after unsuccessful verification of

the digital signature of the Issuer according to FCS_COP.1/VDSUCP the following objects: received Update Code Package. This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).

- FDP_DAU.2/TS requires that the TSF shall provide a capability to generate evidence that can be used as a guarantee of the existence at certain point in time, sequence and validity of (a) user data imported according to FDP_ITC.2/UD, (b) exported audit records according to FMT_MTD.1/Audit clause (1) and FAU_STG.3 clause (1) with (1) time stamp of the evidence generation according to FPT_STM.1, (2) and optionally the key usage counter of the signature key by means of digital signature generated according to FCS_COP.1/CDS-ECDSA and keys holding the dedicated values of the security attributes Key identity that indicate key ownership of the TOE sample and Key usage type "Time stamp service". This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).
- FDP_ITC.2/TS requires that the TSF shall enforce the Cryptographic Operation SFP when importing user data, controlled under the SFP, from outside of the TOE, that the TSF shall use the security attributes associated with the imported user data, that the TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received, that 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, and that the TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: (1) user data imported for time stamp generation to FDP_DAU.2/TS shall be imported with security attributes Key identity of the signature key and Key usage type TimeStamp, and the identification of the requested cryptographic operation. This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).
- FDP_ETC.2/TS requires that the TSF shall enforce the Cryptographic Operation SFP when exporting user data, controlled under the SFP(s), outside of the TOE, that the TSF shall export the user data with the user data's associated security attributes, that the TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data, and that the TSF shall enforce the following rules when user data is exported from the TOE: (1) user data exported as time stamped data according to FDP_DAU.2/TS shall be exported with digital signature and Key identity of the used signature-creation key. This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).

7.1.5 TSF_ SecureMessaging: Secure Messaging

This Security Functionality realizes a secure communication channel after successful authentication. Please note that SFRs of the FCS_COP group are realized within TSF_Crypto, even if they are used by TSF_SecureMessaging.

TSF_SecureMessaging covers the following SFRs:

- FTP_ITC.1 requires that the TSF shall provide a communication channel between TSF and another trusted IT product that is logically separated from other communication channels and provides assured identification of its end points Authentication of TOE and remote entity according to the case in Table 10 and protection of the channel data from modification or disclosure according to the case in Table 10 as required by cryptographic operation according to the case in Table 10; in addition, the TSF shall permit the remote trusted IT product determined according to FMT_MOF.1.1 clause (3) to initiate communication via the trusted channel and the TSF shall initiate communication via the trusted channel for communication with entities defined according to FMT_MOF.1 clause (4). This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS, but part of TSF_SecureMessaging.

- FCS_CKM.1/TCAP requires that the TSF shall generate cryptographic keys for encryption according to FCS_COP.1/TCE and MAC according to FCS_COP.1/TCM in accordance with Terminal Authentication version 2 and Chip Authentication Version 2 and specified cryptographic key sizes 128 bits, 256 bits that meet BSI TR-03110 [TR-03110], section 3.3 and 3.4. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS. It is part of TSF_Auth and of TSF_Secure Messaging.
- FCS_COP.1/TCE requires that the TSF shall perform encryption and decryption in accordance with AES in CBC [NIST-SP800-38A] mode and cryptographic key sizes 128 bits, 256 bits that meet [FIPS197]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS. It is part of TSF_Secure Messaging.
- FCS_COP.1/TCM requires that the TSF shall perform MAC calculation and MAC verification in accordance with AES CMAC [NIST-SP800-38B] and cryptographic key sizes 128 bits, 256 bits that meet [FIPS197]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS. It is part of TSF_Secure Messaging.
- FIA_UAU.6 requires that the TSF shall re-authenticate the user under the conditions (1) changing to a role not selected for the current valid authentication session, (2) power on or reset, (3) every message received from entities after establishing trusted channel according to FIA_UAU.5.1 clause (2), (3) or (6), (4) None. This is part of TSF_SecureMessaging, based on TSF_Access and TSF_Auth.
- FMT_MOF.1 requires that the TSF shall restrict the ability to (1) enable the function password authentication according to FIA_UAU.5.1, clause (1) to User Administrator, (2) disable the function password authentication according to FIA_UAU.5.1, clause (1) to User Administrator, (3) determine the behaviour of the functions trusted channel according to FDP_ITC.1.2 by defining the remote trusted IT products permitted to initiate communication via the trusted channel to User Administrator, (4) determine the behaviour of the functions trusted channel according to FDP_ITC.1.3 by defining the entities for which the TSF shall enforce communication via the trusted channel to User Administrator. This is realized by TSF_Admin together with TSF_Auth and TSF_SecureMessaging.

7.1.6 TSF_Auth: Authentication protocols

This security functionality realizes different authentication mechanisms. TSF_Auth covers the following SFRs:

- FPT_TIT.1/Cert requires that the TSF shall enforce the Key Management SFP to receive certificate in a manner protected from modification and insertion errors, and that the TSF shall be able to determine on receipt of certificate, whether modification and insertion has occurred. This is partially realized by TSF_Auth.
- FIA_API.1/CA requires that the TSF shall provide Chip Authentication Version 2 according to [TR03110] section 3.4 to prove the identity of the TOE to an external entity and establishing a trusted channel according to FTP_ITC.1 case 3. This is realized by TSF_Auth, while the cryptographic functionality is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FDP_DAU.2/Att requires that the TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of attestation data by means of FCS_COP.1/CDS-ECDSA and keys holding the security attributes Key identity assigned to the TOE sample and Key usage type "Attestation", and that the TSF shall provide external entities with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence. This is realized by TSF_Auth, while the cryptographic functionality is realized by TSF_Crypto based on cryptographic functionality of TSF_OS .

- FCS_CKM.1/TCAP requires that the TSF shall generate cryptographic keys for encryption according to FCS_COP.1/TCE and MAC according to FCS_COP.1/TCM in accordance with Terminal Authentication version 2 and Chip Authentication Version 2 and specified cryptographic key sizes 128 bits, 256 bits that meet BSI TR-03110 [TR-03110], section 3.3 and 3.4. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS. It is part of TSF_Auth and of TSF_Secure Messaging.
- FIA_ATD.1 requires that the TSF shall maintain the following list of security attributes belonging to individual users: (1) Identity, (2) Authentication reference data, (3) Role. This is realized by TSF_Auth together wit TSF_Access.
- FMT_MTD.1/RAD requires that the TSF shall restrict the ability to (1) create the initial Authentication reference data of all authorized users to User Administrator, (2) delete the Authentication reference data of an authorized user to User Administrator, (3) modify the Authentication reference data to the corresponding authorized user, (4) create the permanently stored session key of trusted channel as Authentication reference data to User Administrator, (5) define the time in range 1 – (2^32-1) seconds after which the user security attribute Role is reset according to FMT_SAE.1 to User Administrator, and (6) define the value Unauthenticated user to which the security attribute Role shall be reset according to FMT_SAE.1 to User Administrator. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.
- FMT_MTD.3 requires that the TSF shall ensure that only secure values are accepted for passwords by enforcing change of initial passwords after first successful authentication of the user to different operational password. This is realized by TSF_Admin together with TSF_Auth.
- FIA_AFL.1 requires that the TSF shall detect when a User Administrator configurable positive integer within 1 - 15 unsuccessful authentication attempts occur related to (1) PACE based authentication, (2) Password based authentication, (3) Cryptographic Entity Authentication; and when the defined number of unsuccessful authentication attempts has been met, the TSF shall delay the next authentication attempt or block the authentication, configurable by the administrator. This is realized by TSF_Admin together with TSF_Auth.
- FIA_USB.1 requires that the TSF shall associate the following user security attributes with subjects acting on the behalf of that user: (1) Identity, (2) Role; it requires that 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; it requires that the TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: (1) after successful identification of the user the attribute Role of the subject shall be changed from Unidentified user to Unauthenticated user; (2) after successful authentication of the user for a selected role the attribute Role of the subject shall be changed from Unauthenticated User to that role; (3) after successful re-authentication of the user for a selected role the attribute Role of the subject shall be changed to that role. This is realized by TSF_Admin together with TSF_Auth.
- FMT_SAE.1 requires that the TSF shall restrict the capability to specify an expiration time for Role to User Administrator, and that for each of these security attributes, the TSF shall be able to reset the Role to the value assigned according to FMT_MTD.1/RAD, clause (6) after the expiration time for the indicated security attribute has passed. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.
- FIA_UID.1 requires that the TSF shall allow (1) self test according to FPT_TST.1, (2) identification of the TOE to the user, (3) None on behalf of the user to be performed before the user is identified, and that the TSF shall require each user to be successfully identified before allowing any other TSF-

mediated actions on behalf of the Unauthenticated User. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.

- FIA_UAU.1 requires that the TSF shall allow (1) self test according to FPT_TST.1, (2) authentication of the TOE to the user, (3) identification of the user to the TOE and selection of a set of role for authentication, (4) none on behalf of the user to be performed before the user is authenticated, and that the TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user. This is realized by TSF_Admin together with TSF_Access and TSF_Auth.
- FIA_UAU.5 requires that the TSF shall provide (1) password authentication, (2) PACE with Generic Mapping with TOE in ICC and user in PCD context with establishment of trusted channel according to FTP_ITC.1, (3) certificate based Terminal Authentication Version 2 according to section 3.3 in [TR-03110] with the TOE in ICC and user in PCD context, (4) Terminal Authentication Version 2 with the TOE in ICC context and user in PCD context modified by omitting the verification of the certificate chain, (5) Chip Authentication Version 2 with establishment of trusted channel according to FTP_ITC.1, (6) message authentication by MAC verification of received message to support user authentication. It also requires that the TSF shall authenticate any user's claimed identity according to the rules (1) password authentication shall be used for authentication of human users if enabled according to FMT_MOF.1.1, clause (1), (2) PACE shall be used for authentication of human users using terminals with establishment of trusted channel according to FTP_ITC.1, (3) PACE may be used for authentication of IT entities with establishment of trusted channel according to FTP_ITC.1, (4) certificate based Terminal Authentication Version 2 may be used for authentication of users which certificate imported as TSF data, (5) simplified version of Terminal Authentication Version 2 may be used for authentication of identified users associated with known user's public key, (6) message authentication by MAC verification of received messages shall be used after initial authentication of remote entity according to clauses (2) or (3) for trusted channel according to FTP_ITC.1, (7) None. This is realized by TSF_Auth based on functionality from TSF_Crypto, which itself is based on cryptographic functionality of TSF_OS.
- FIA_UAU.6 requires that the TSF shall re-authenticate the user under the conditions (1) changing to a role not selected for the current valid authentication session, (2) power on or reset, (3) every message received from entities after establishing trusted channel according to FIA_UAU.5.1 clause (2), (3) or (6), (4) None. This is part of TSF_SecureMessaging, based on TSF_Access and TSF_Auth.
- FMT_SMF.1 requires that the TSF shall be capable of performing the following management functions: (1) management of security functions behaviour (FMT_MOF.1), (2) management of Authentication reference data (FMT_MTD.1/RAD), (3) management of security attributes of cryptographic keys (FMT_MSA.1/KM, FMT_MSA.2, FMT_MSA.3/KM, (4) None. This is realized by TSF_Admin together with TSF_Auth and TSF_Crypto.
- FMT_MOF.1 requires that the TSF shall restrict the ability to (1) enable the function password authentication according to FIA_UAU.5.1, clause (1) to User Administrator, (2) disable the function password authentication according to FIA_UAU.5.1, clause (1) to User Administrator, (3) determine the behaviour of the functions trusted channel according to FDP_ITC.1.2 by defining the remote trusted IT products permitted to initiate communication via the trusted channel to User Administrator, (4) determine the behaviour of the functions trusted channel according to FDP_ITC.1.3 by defining the entities for which the TSF shall enforce communication via the trusted channel to User Administrator. This is realized by TSF_Admin together with TSF_Auth and TSF_SecureMessaging.

7.1.7 TSF_Integrity: Integrity protection

This Security Functionality protects the integrity of internal data. This function makes use of the underlying Java Card OS.

TSF_Integrity covers the following SFRs:

- FPT_TST.1 requires that the TSF shall run a suite of self tests during initial start-up, at the request of the authorised user and after power-on to demonstrate the correct operation of the Java Card platform, that the TSF shall provide authorised users with the capability to verify the integrity of TSF data, and that the TSF shall provide authorised users with the capability to verify the integrity of TSF implementation. This is part TSF_Integrity and based on TSF_OS.
- FPT_PHP.3 requires that the TSF shall resist (1) physical probing and manipulation and (2) perturbation and environmental stress to the (1) TSF implementation and (2) the TSF by responding automatically such that the SFRs are always enforced. A refinement adds that the TSF will implement appropriate mechanisms to continuously counter physical probing and manipulation. In case of platform architecture the resistance to physical attacks shall include the secure execution environment for and the communication with the application component running on the TOE. This is part TSF_Integrity and based on TSF_OS.

7.1.8 TSF_OS: Javacard OS Security Functionalities

The Javacard operation system (part of the TOE) features a set of certified security functionalities. The realization is partly based on the security functionalities of the certified cryptographic library and the certified IC platform:

- FCS_COP.1/Hash requires that the TSF shall perform hash generation in accordance with a specified cryptographic algorithm SHA-256, SHA-384, SHA-512. This is based on cryptographic functionality of TSF_OS.
- FPT_TIT.1/Cert requires that the TSF shall enforce the Key Management SFP to receive certificate in a manner protected from modification and insertion errors, and that the TSF shall be able to determine on receipt of certificate, whether modification and insertion has occurred. TSF_OS realizes the main part of the necessary cryptographic mechanisms.
- FPT_ISA.1/Cert requires that the TSF shall enforce the Key management SFP when importing certificates , controlled under the SFP, from outside of the TOE, that the TSF shall use the security attributes associated with the imported certificate, ensure that the protocol used provides for the unambiguous association between the security attributes and the certificates received and that interpretation of the security attributes of the imported certificates is as intended by the source of the certificates, and that the TSF shall enforce a defined set of rules when importing certificates controlled under the SFP from outside the TOE. This cryptographic functionality is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_RNG.1 requires that the TSF shall provide a deterministic random number generator that implements a set of properties to fulfill the DRG.3 definition of [AIS20]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.1/AES requires that the TSF shall generate cryptographic AES key in accordance with a specified cryptographic key generation algorithm AES and key size 128 bits, 256 bits that meet [ISO18033-3]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.5/AES requires that the TSF shall derive AES keys from a byte array of variable length in accordance with a specified cryptographic key derivation algorithms using bit strings derived from

input parameters with KDF and specified cryptographic key sizes 128 bits, 256 bits that meet [NIST-SP800-56C]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.

- FCS_CKM.1/ECC requires that the TSF shall generate elliptic curve key pairs in accordance with a specified cryptographic key generation algorithm and specified cryptographic key sizes 256, 384, 512 and 521 bit, respectively that meet [RFC5639], TR-03111, section 4.1.3 [TR03111], or FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 and D.1.2.5 [FIPS186-4], respectively. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.5/ECC requires that the TSF shall derive cryptographic elliptic curve key pair from a byte array of variable length in accordance with a specified cryptographic key derivation algorithm using bit string derived from input parameters with X9.63 Key Derivation Function according to [TR03111], page 27 and specified cryptographic key sizes 256, 384, 512 bit that meet the following: RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR03111], or FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS186-4], respectively , and [TR03111]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.1/RSA requires that the TSF shall generate cryptographic RSA key pair in accordance with a specified cryptographic key generation algorithm and specified cryptographic key sizes from 2000 bit to 4096 bit in one bit steps that meet the following: PKCS #1 v2.2 [PKCS1]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.5/ECDHE requires that the TSF shall derive cryptographic ephemeral keys for data encryption and MAC with AES-128, AES-256 from an agreed shared secret in accordance with a specified cryptographic key derivation algorithm (Elliptic Curve Diffie-Hellman ephemeral key agreement) with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384 and 256-bit random ECP group, 384-bit random ECP group, brainpoolP256r1, brainpoolP384r1, brainpoolP512r1 with a key derivation from the shared secret X9.63 Key Derivation Function and specified cryptographic key sizes 128 bits, 256 bits that meet the following: TR-03111 [TR-03111]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.1/ECKA-EG requires that the TSF shall generate an ephemeral cryptographic elliptic curve key pair for ECKGA-EG [TR-03111], sender role in accordance with a specified cryptographic key generation algorithm with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384 and specified cryptographic key sizes 256 bit, 384 bit, 512 bit that meet RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.5/ECKA-EG requires that the TSF shall derive cryptographic data encryption key and MAC keys for AES 128, AES-256 from a private and a public ECC key in accordance with a specified cryptographic key derivation algorithm ECKGA-EG [TR-03111] with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384 and X9.63 Key Derivation Function and specified cryptographic symmetric key sizes 128 bits, 256 bits that meet TR-03111 [TR-03111], chapter 4.3.2.2. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.1/AES_RSA requires that the TSF shall generate and encrypt seed, derive cryptographic keys from seed for data encryption and MAC with AES-128, AES-256 in accordance with a specified cryptographic key generation algorithm: X9.63 Key Derivation Function [ANSI-X9.63] and RSA EME-OAEP [PKCS#1] and specified cryptographic symmetric key sizes 128 bits, 256 bits that meet ISO/IEC 18033-3 [ISO/IEC 18033-3], PKCS #1 v2.2 [PKCS#1]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.

- FCS_CKM.5/AES_RSA requires that the TSF shall derive cryptographic data encryption key and MAC key for AES-128, AES-256 from decrypted RSA encrypted seed in accordance with RSA EME-OAEP [PKCS#1] and X9.63 [ANSI-X9.63] Key Derivation Function and specified cryptographic symmetric key sizes 128 bits, 256 bits that meet ISO/IEC 14888-2 [ISO/IEC 14888-2]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.4 requires that the TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method: overwriting the keys. This is partially realized by TSF_OS (for keys directly initiated as cryptographic keys in the Java Card OS).
- FCS_COP.1/KW requires that the TSF shall perform key wrap in accordance with AES-Keywrap KWP and cryptographic key sizes of the key encryption key 128 bit that meet NIST SP800-38F [NIST-SP800-38F]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_COP.1/KU requires that the TSF shall perform key unwrap in accordance with AES-Keywrap KWP and cryptographic key sizes of the key encryption key of 128 bit that meet NIST SP800-38F [NIST-SP800-38F]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FPT_TCT.1/CK requires that the TSF shall enforce the Key Management SFP by providing the ability to transmit and receive cryptographic keys in a manner protected from unauthorised disclosure according to FCS_COP.1/KW and FCS_COP.1/KU. This is partially realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FPT_TIT.1/CK requires that the TSF shall enforce the Key Management SFP to transmit and receive cryptographic keys in a manner protected from modification and insertion errors according to FCS_COP.1/KW, and that the TSF shall be able to determine on receipt of cryptographic keys, whether modification and insertion has occurred according to FCS_COP.1/KU. This is partially realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FPT_ISA.1/CK requires that the TSF shall enforce the Key Management SFP when importing cryptographic key, controlled under the SFP, from outside of the TOE, that the TSF shall use the security attributes associated with the imported cryptographic key, that the TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the cryptographic key received, that the TSF shall ensure that interpretation of the security attributes of the imported cryptographic key is as intended by the source of the cryptographic key, and that the TSF shall enforce the following rule when importing cryptographic key controlled under the SFP from outside the TOE: The TSF imports the TSF data in certificates only after successful verification of the validity of the certificate including verification of digital signature of the issuer and validity time period. This is partially realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_COP.1/ED requires that the TSF shall perform data encryption and decryption in accordance with symmetric data encryption according to AES-128 and AES-256 in CBC and no other mode and cryptographic key size 128 bits, 256 bits that meet NIST SP800-38A [NIST-SP800-38A], ISO/IEC 18033-3 [ISO/IEC 18033-3], ISO/IEC 10116 [ISO/IEC 10116]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_COP.1/HEM requires that the TSF shall perform hybrid data encryption and MAC calculation in accordance with a specified cryptographic algorithm asymmetric key encryption according to FCS_CKM.1/ECKA-EG , symmetric data encryption according to AES-128, AES-256 [FIPS197] in CBC [NIST-SP800-38A] mode with CMAC [NIST-SP800-38B] calculation and cryptographic symmetric key sizes 128 bits, 256 bits that meet the referenced standards above. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.

- FCS_COP.1/HDM requires that the TSF shall perform hybrid MAC verification and data decryption in accordance with a specified cryptographic algorithm asymmetric key decryption according to FCS_CKM.5/ECKA-EG , verification of CMAC [NIST-SP800-38B] and symmetric data decryption according to AES with AES-128, AES-256 [FIPS197] in mode CBC [NIST-SP800-38A] and cryptographic symmetric key sizes 128 bits, 256 bits that meet the referenced standards above. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_COP.1/MAC requires that the TSF shall perform MAC generation and verification in accordance with a specified cryptographic algorithm AES-128 and AES-256 [FIPS197] CMAC [NIST-SP800-38B] and cryptographic key sizes 128 bits, 256 bits that meet the referenced standards above. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_COP.1/HMAC requires that the TSF shall perform HMAC generation and verification in accordance with a specified cryptographic algorithm HMAC-SHA256 and cryptographic key sizes 256 bit that meet RFC2104 [RFC2104], ISO/IEC 9797-2 [ISO/IEC 9797-2]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_COP.1/CDS-ECDSA requires that the TSF shall perform signature-creation in accordance with a specified cryptographic algorithm ECDSA with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384, Curve P-521 and cryptographic key sizes 256, 384, 512 and 521 bit that meet RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_COP.1/VDS-ECDSA requires that the TSF shall perform signature-verification in accordance with a specified cryptographic algorithm ECDSA with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384, Curve P-521 and cryptographic key sizes 256, 384, 512 and 521 bit that meet RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_COP.1/CDS-RSA requires that the TSF shall perform signature-creation in accordance with a specified cryptographic algorithm RSA and EMSA-PSS and cryptographic key sizes 2000-4096 bit that meet ISO/IEC 14888-2 [ISO/IEC 14888-2], PKCS #1, v2.2 [PKCS#1]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_COP.1/VDS-RSA requires that the TSF shall perform signature-verification in accordance with a specified cryptographic algorithm RSA and EMSA-PSS and cryptographic key sizes 2000-4096 bit that meet the following: ISO/IEC 14888-2 [ISO/IEC 14888-2], PKCS #1, v2.2 [PKCS#1]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FDP_DAU.2/Sig requires that the TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of user data imported according to FDP_ITC.2/UD by means of FCS_COP.1/CDS-RSA, FCS_COP.1/CDS-ECDSA and keys holding the security attributes Key identity assigned to the guarantor and Key usage type “Signature service”, and that the TSF shall provide external entities with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FIA_API.1/PACE requires that the TSF shall provide PACE in ICC role to prove the identity of the TOE to an external entity and establishing a trusted channel according to FTP_ITC.1 case 1 or 2. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.

- FIA_API.1/CA requires that the TSF shall provide Chip Authentication Version 2 according to [TR03110] section 3.4 to prove the identity of the TOE to an external entity and establishing a trusted channel according to FTP_ITC.1 case 3. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS (and TSF_Auth).
- FDP_DAU.2/Att requires that the TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of attestation data by means of FCS_COP.1/CDS-ECDSA and keys holding the security attributes Key identity assigned to the TOE sample and Key usage type "Attestation", and that the TSF shall provide external entities with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS (and TSF_Auth).
- FTP_ITC.1 requires that the TSF shall provide a communication channel between TSF and another trusted IT product that is logically separated from other communication channels and provides assured identification of its end points Authentication of TOE and remote entity according to the case in Table 10 and protection of the channel data from modification or disclosure according to the case in Table 10 as required by cryptographic operation according to the case in Table 10; in addition, the TSF shall permit the remote trusted IT product determined according to FMT_MOF.1.1 clause (3) to initiate communication via the trusted channel and the TSF shall initiate communication via the trusted channel for communication with entities defined according to FMT_MOF.1 clause (4). This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.1/PACE requires that the TSF shall generate cryptographic keys for MAC for FCS_COP.1/TCM and encryption keys for FCS_COP.1/TCE in accordance with a specified cryptographic key agreement algorithm PACE with brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, Curve P-256, Curve P-384 and Generic Mapping in ICC role and specified cryptographic key sizes 128, 256 bits that meet ICAO Doc9303, Part 11, section 4.4 [ICAO Doc9303]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS.
- FCS_CKM.1/TCAP requires that the TSF shall generate cryptographic keys for encryption according to FCS_COP.1/TCE and MAC according to FCS_COP.1/TCM in accordance with Terminal Authentication version 2 and Chip Authentication Version 2 and specified cryptographic key sizes 128 bits, 256 bits that meet BSI TR-03110 [TR-03110], section 3.3 and 3.4. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS. It is part of TSF_Auth and of TSF_Secure Messaging.
- FCS_COP.1/TCE requires that the TSF shall perform encryption and decryption in accordance with AES in CBC [NIST-SP800-38A] mode and cryptographic key sizes 128 bits, 256 bits that meet [FIPS197]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS. It is part of TSF_Secure Messaging.
- FCS_COP.1/TCM requires that the TSF shall perform MAC calculation and MAC verification in accordance with AES CMAC [NIST-SP800-38B] and cryptographic key sizes 128 bits, 256 bits that meet [FIPS197]. This is realized by TSF_Crypto based on cryptographic functionality of TSF_OS. It is part of TSF_Secure Messaging.
- FIA_UAU.5 requires that the TSF shall provide (1) password authentication, (2) PACE with Generic Mapping with TOE in ICC and user in PCD context with establishment of trusted channel according to FTP_ITC.1, (3) certificate based Terminal Authentication Version 2 according to section 3.3 in [TR-03110] with the TOE in ICC and user in PCD context, (4) Terminal Authentication Version 2 with the TOE in ICC context and user in PCD context modified by omitting the verification of the certificate chain, (5) Chip Authentication Version 2 with establishment of trusted channel according to FTP_ITC.1, (6) message authentication by MAC verification of received message to support user

authentication. It also requires that the TSF shall authenticate any user's claimed identity according to the rules (1) password authentication shall be used for authentication of human users if enabled according to FMT_MOF.1.1, clause (1), (2) PACE shall be used for authentication of human users using terminals with establishment of trusted channel according to FTP_ITC.1, (3) PACE may be used for authentication of IT entities with establishment of trusted channel according to FTP_ITC.1, (4) certificate based Terminal Authentication Version 2 may be used for authen-tication of users which certificate imported as TSF data, (5) simplified version of Terminal Authentication Version 2 may be used for au-thentication of identified users associated with known user's public key, (6) message authentication by MAC verification of received messages shall be used after initial authentication of remote entity according to clauses (2) or (3) for trusted channel according to FTP_ITC.1, (7) None. This is realized by TSF_Auth based on functionality from TSF_Crypto, which itself is based on cryptographic functionality of TSF_OS.

- FDP_SDC.1 requires that the TSF shall ensure the confidentiality of the information of the data while it is stored in the user chosen memory area by encryption according to FCS_COP.1/SDE. This is realized by TSF_Secret based on TSF_Crypto (itself based on TSF_OS).
- FCS_CKM.1/SDEK requires that the TSF shall generate cryptographic stored data encryption key in accordance with a specified cryptographic key generation algorithm as specified in FCS_CKM.1/AES using random bit generation according to FCS_RNG.1 and key sizes 128, 256 bit that meet [ISO/IEC 180333]. This is realized by TSF_Secret based on TSF_Crypto (itself based on TSF_OS).
- FCS_COP.1.1/SDE requires that the TSF shall perform stored data encryption and decryption in accordance with AES in CBC mode and key sizes 128, 256 bit that meet: [FIPS197], [NIST-SP800-38A]. This is realized by TSF_Secret based on TSF_Crypto (itself based on TSF_OS).
- FRU_FLT.2 requires that the TSF shall ensure the operation of all the TOE's capabilities when the following failures occur: exposure to operating conditions which are not detected according to the requirement Failure with preservation of secure state (FPT_FLS.1). This is realized by TSF_OS.
- FPT_FLS.1 requires that the TSF shall preserve a secure state when the following types of failures occur: (1) self test fails, (2) exposure to operating conditions which may not be tolerated according to the requirement Limited fault tolerance (FRU_FLT.2) and where therefore a malfunction could occur, (3) manipulation and physical probing is detected and secure state is reached as response (FPT_PHP.3). A refinement adds that when the TOE is in a secure error mode the TSF shall not perform any cryptographic operations and all data output interfaces shall be inhibited by the TSF. This is realized by TSF_OS.
- FPT_TST.1 requires that the TSF shall run a suite of self tests during initial start-up, at the request of the authorised user and after power-on to demonstrate the correct operation of the Java Card platform, that the TSF shall provide authorised users with the capability to verify the integrity of TSF data, and that the TSF shall provide authorised users with the capability to verify the integrity of TSF implementation. This is part TSF_Integrity and based on TSF_OS.
- FPT_PHP.3 requires that the TSF shall resist (1) physical probing and manipulation and (2) perturbation and environmental stress to the (1) TSF implementation and (2) the TSF by responding automatically such that the SFRs are always enforced. A refinement adds that the TSF will implement appropriate mechanisms to continuously counter physical probing and manipulation. In case of platform architecture the resistance to physical attacks shall include the secure execution environment for and the communication with the application component running on the TOE. This is part TSF_Integrity and based on TSF_OS.

- FDP_ITC.2/UCP requires that the TSF shall enforce the Update SFP when importing user data, controlled under the SFP, from outside of the TOE; that the TSF shall use the security attributes associated with the imported user data; that the TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received; that 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; and that the TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: (1) storing of encrypted Update Code Package only after successful verification of authenticity according to FCS_COP.1/VDSUCP, (2) decrypts authentic Update Code Package according to FCS_COP.1/DecUCP. This is realized by TSF_Admin and mechanisms provided by TSF_OS.
- FPT_TDC.1/UCP requires that the TSF shall provide the capability to consistently interpret security attributes Issuer and Version Number when shared between the TSF and another trusted IT product, and that the TSF shall use the following rules: (1) the Issuer must be identified and known, (2) the Version Number must be identified when interpreting the TSF data from another trusted IT product. This is realized by TSF_Admin and mechanisms provided by TSF_OS.
- FCS_COP.1/VDSUCP requires that the TSF shall perform verification of the digital signature of the authorized Issuer in accordance with ECDSA and key size 256 bit that meet RFC5639 [RFC5639], TR-03111, section 4.1.3 [TR-03111], FIPS PUB 186-4 B.4 and D.1.2.3, D.1.2.4 AND D.1.2.5 [FIPS PUB 186-4]. This is realized by TSF_Admin and mechanisms provided by TSF_OS.
- FCS_COP.1/DecUCP requires that the TSF shall perform decryption of authentic encrypted Update Code Package in accordance with AES in CBC mode and key size 128 bit that meet [FIPS197], [NIST SP800-38A]. This is realized by TSF_Admin and mechanisms provided by TSF_OS.
- FDP_ACF.1/UCP requires that the TSF shall enforce the Update SFP to objects based on the following: (1) subjects: Update Agent; (2) objects: Update Code Package with security attributes Issuer and Version Number. It requires that the TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: (1) Update Agent is allowed to import Update Code Package according to FDP_ITC.2/UCP. (2) Update Agent is allowed to store Update Code Package if (a) authenticity is successful verified according to FCS_COP.1/VDSUCP and decrypted according to FCS_COP.1/DecUCP; (b) the Version Number of the Update Code Package is equal or higher than the Version Number of the TSF. It also requires that the TSF shall explicitly authorise access of subjects to objects based on the following additional rules: None and that the TSF shall explicitly deny access of subjects to objects based on the following additional rules: None. This is realized by TSF_Admin together wit TSF_Crypto (based on TSF_OS).
- FDP_RIP.1/UCP requires that the TSF shall ensure that any previous information content of a resource is made unavailable upon the deallocation of the resource after unsuccessful verification of the digital signature of the Issuer according to FCS_COP.1/VDSUCP the following objects: received Update Code Package. This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).
- FDP_DAU.2/TS requires that the TSF shall provide a capability to generate evidence that can be used as a guarantee of the existence at certain point in time, sequence and validity of (a) user data imported according to FDP_ITC.2/UD, (b) exported audit records according to FMT_MTD.1/Audit clause (1) and FAU_STG.3 clause (1) with (1) time stamp of the evidence generation according to FPT_STM.1, (2) and optionally the key usage counter of the signature key by means of digital signature generated according to FCS_COP.1/CDS-ECDSA and keys holding the dedicated values of the security attributes Key identity that indicate key ownership of the TOE sample and Key usage type "Time stamp service". This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).

- FDP_ITC.2/TS requires that the TSF shall enforce the Cryptographic Operation SFP when importing user data, controlled under the SFP, from outside of the TOE, that the TSF shall use the security attributes associated with the imported user data, that the TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received, that 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, and that the TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: (1) user data imported for time stamp generation to FDP_DAU.2/TS shall be imported with security attributes Key identity of the signature key and Key usage type TimeStamp, and the identification of the requested cryptographic operation. This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).
- FDP_ETC.2/TS requires that the TSF shall enforce the Cryptographic Operation SFP when exporting user data, controlled under the SFP(s), outside of the TOE, that the TSF shall export the user data with the user data's associated security attributes, that the TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data, and that the TSF shall enforce the following rules when user data is exported from the TOE: (1) user data exported as time stamped data according to FDP_DAU.2/TS shall be exported with digital signature and Key identity of the used signature-creation key. This is realized by TSF_Admin together with TSF_Crypto (based on TSF_OS).

7.2 TOE summary specification rationale

This summary specification shows that the TSF and assurance measures are appropriate to fulfill the TOE security requirements.

Each TOE security functional requirement is implemented by at least one security functionality. The mapping of TOE Security Requirements and TOE Security Functionalities is given in the following table. The description of the TSF is given in section 7.1.

	TSF_Access	TSF_Admin	TSF_Secret	TSF_Crypto	TSF_SecureMessaging	TSF_Auth	TSF_Integrity	TSF_OS
FDP_ACC.1/KM	x							
FMT_MSA.1/KM	x							
FMT_MSA.3/KM	x	x						
FMT_MTD.1/KM	x							
FCS_COP.1/Hash				x				x
FMT_MTD.1/RK	x							
FPT_TIT.1/Cert	x			x		x		x
FPT_ISA.1/Cert	x			x				x

	TSF_Access	TSF_Admin	TSF_Secret	TSF_Crypto	TSF_SecureMessaging	TSF_Auth	TSF_Integrity	TSF_OS
FPT_TDC.1/Cert				x				
FCS_RNG.1				x				x
FCS_CKM.1/AES				x				x
FCS_CKM.5/AES				x				x
FCS_CKM.1/ECC				x				x
FCS_CKM.5/ECC				x				x
FCS_CKM.1/RSA				x				x
FCS_CKM.5/ECDHE				x				x
FCS_CKM.1/ECKA-EG				x				x
FCS_CKM.5/ECKA-EG				x				x
FCS_CKM.1/AES_RSA				x				x
FCS_CKM.5/AES_RSA				x				x
FCS_CKM.4				x				x
FCS_COP.1/KW				x				x
FCS_COP.1/KU				x				x
FPT_TCT.1/CK		x		x				x
FPT_TIT.1/CK		x		x				x
FPT_ISA.1/CK		x		x				x
FPT_TDC.1/CK		x		x				
FPT_ESA.1/CK		x		x				
FCS_COP.1/ED				x				x
FCS_COP.1/HEM				x				x
FCS_COP.1/HDM				x				x
FCS_COP.1/MAC				x				x
FCS_COP.1/HMAC				x				x
FCS_COP.1/CDS-ECDSA				x				x
FCS_COP.1/VDS-ECDSA				x				x
FCS_COP.1/CDS-RSA				x				x
FCS_COP.1/VDS-RSA				x				x
FDP_DAU.2/Sig				x				x
FIA_API.1/PACE				x				x
FIA_API.1/CA				x		x		x

	TSF_Access	TSF_Admin	TSF_Secret	TSF_Crypto	TSF_SecureMessaging	TSF_Auth	TSF_Integrity	TSF_OS
FDP_DAU.2/Att				x		x		x
FTP_ITC.1				x	x			x
FCS_CKM.1/PACE				x				x
FCS_CKM.1/TCAP				x	x	x		x
FCS_COP.1/TCE				x	x			x
FCS_COP.1.1/TCM				x	x			x
FIA_ATD.1	x					x		
FMT_MTD.1/RAD	x	x				x		
FMT_MTD.3		x				x		
FIA_AFL.1		x				x		
FIA_USB.1		x				x		
FMT_SAE.1	x	x				x		
FIA_UID.1	x	x				x		
FIA_UAU.1	x	x				x		
FIA_UAU.5				x		x		x
FIA_UAU.6	x				x	x		
FDP_ITC.2/UD		x						
FDP_ETC.2		x						
FDP_ETC.1		x		x				
FDP_ACC.1/Oper	x	x		x				
FDP_ACF.1/Oper	x	x		x				
FMT_SMF.1		x		x		x		
FMT_SMR.1		x						
FMT_MSA.2		x		x				
FMT_MOF.1		x			x	x		
FDP_SDC.1			x	x				x
FCS_CKM.1/SDEK			x	x				x
FCS_COP.1/SDE			x	x				x
FRU_FLT.2								x
FPT_FLS.1								x
FPT_TST.1							x	x
FPT_PHP.3							x	x

	TSF_Access	TSF_Admin	TSF_Secret	TSF_Crypto	TSF_SecureMessaging	TSF_Auth	TSF_Integrity	TSF_OS
FDP_ITC.2/UCP		x						x
FPT_TDC.1/UCP		x						x
FCS_COP.1/VDSUCP		x						x
FCS_COP.1/DecUCP		x						x
FDP_ACC.1/UCP		x						
FDP_ACF.1/UCP		x		x				x
FDP_RIP.1/UCP		x		x				x
FDP_DAU.2/TS		x		x				x
FDP_ITC.2/TS		x		x				x
FDP_ETC.2/TS		x		x				x
FDP_ACF.1/TS	x	x						
FMT_SMF.1/TSA		x						
FMT_SMR.1/TSA	x	x						
FMT_MOF.1/TSA	x	x						
FAU_GEN.1		x		x				x
FMT_MTD.1/Audit	x	x						
FAU_STG.1		x						x
FAU_STG.3		x						
FPT_STM.1	x	x						
FPT_TIT.1/Audit		x						

Table 14: Mapping of TOE Security Requirements and TOE Security Functionalities.

8 References

In the following tables, the references used in this document are summarized.

Common Criteria

[CC_1]	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 3.1, Revision 5, April 2017; CCMB-2017-04-001.
[CC_2]	Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; Version 3.1, Revision 5, April 2017; CCMB-2017-04-002.
[CC_3]	Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; Version 3.1, Revision 5, April 2017; CCMB-2017-04-003.
[CC_4]	Common Methodology for Information Technology Security Evaluation, Evaluation Methodology; Version 3.1, Revision 5, April 2017; CCMB-2017-04-004.

Protection Profiles

[PP0104]	Common Criteria Protection Profile “Cryptographic Service Provider”, BSI-CC-PP-0104-2019, Version 0.9.8, Bundesamt für Sicherheit in der Informationstechnik.
[PP0107]	Common Criteria Protection Profile Configurations Cryptographic Service Provider – Time Stamp Service and Audit (PPC-CSP-TS-Au) Protection Profile-Module CSP Time Stamp Service and Audit (PPM-TS-Au), BSI-CC-PP-0107-2019, Version 0.9.5, Bundesamt für Sicherheit in der Informationstechnik.
[PP_Javacard]	Java Card Protection Profile - Open Configuration, Version 3.0 (May 2012), Published by Oracle, Inc.

TOE and Platform References

[ST_Javacard]	Security Target Lite NXP JCOP 4.7 SE051, Rev. 2.1, 29 June 2022; Evaluation document, Public, NSCIB-CC-0095534-2MA.
[Zert_Javacard]	Certification Report NXP JCOP 4.7 SE051, Report number: NSCIB-CC-0095534-CR2, TÜV Rheinland Nederland B.V., 25 November 2021. with Assurance Continuity Maintenance Report JCOP 4.7 SE051, Report number: NSCIB-CC-0095534-2MA, TÜV Rheinland Nederland B.V., 05 July 2022.
[ST_IC]	NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library (R1/R2) - Security Target Lite - Rev. 2.5 — 4 May 2022, BSI-DSZ-CC-1136-V2.
[Zert_IC]	Certification Report BSI-DSZ-CC-1136-V2-2022 for NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library (R1/R2) from NXP Semiconductors Germany GmbH; 2022-06-02.
[Guidance_PRE]	cryptovision CSP – Java Card applet providing Cryptographic Service Provider - Preparation Guidance (AGD_PRE).
[Guidance_OPE]	cryptovision CSP – Java Card applet providing Cryptographic Service Provider - Operational Guidance (AGD_OPE).

[GP_CIC]	GlobalPlatform Card Common Implementation Configuration Version 1.0, February 2014
[AGD_PRE]	JCOP 4.7 SE051, User manual for JCOP 4.7 SE051, User Guidance and Administrator Manual, Revision 1.5, DocNo 581815, NXP Semiconductors, 14 June 2022.

Cryptography

AIS20	Anwendungshinweise und Interpretationen zum Schema (AIS); AIS 20, Version 3, 15.05.2013, Bundesamt für Sicherheit in der Informationstechnik
ANSI X9.62-2005	ANSI X9.62-2005: Public Key Cryptography for the Financial Services Industry: the Elliptic Curve Digital Signature Algorithm (ECDSA), American National Standards Institute (ANSI), 2005.
ANSI-X9.63	ANSI-X9.63, Key Agreement and Key Transport Using Elliptic Curve Cryptography, 2011
FIPS PUB 180-4	NIST, Secure Hash, Standard (SHS), 2012
FIPS PUB 186-4	NIST, Digital Signature Standard (DSS), 2013
FIPS197	Federal Information Processing Standards Publication 197 (FIPS PUB 197), Advanced Encryption Standard (AES), 2001
ICAO Doc9303	ICAO Doc9303ICAO: Machine Readable Travel Documents, ICAO Doc9303, Part 11: Security Mechanisms for MRTDSs, seventh edition, 2015
ISO/IEC 10116	ISO/IEC 10116 Information Technology - Security techniques, Modes of operation for an n-bit block cipher, 2017
ISO/IEC 14888-2	ISO/IEC 14888-2 Information technology – Security techniques, Digital signatures with appendix – Part 2: Integer factorization based mechanisms, 2008
ISO/IEC 14888-3	ISO/IEC 14888-3:2015: Information technology – Security techniques – Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms, 2016.
ISO/IEC 18033-3	ISO/IEC 18033-3 Information technology - Security techniques, Encryption algorithms - Part 3: Block ciphers, 2010
ISO/IEC 9797-2	ISO/IEC 9797-2 Information Technology - Security techniques, Message Authentication Codes (MACs), Part 2: Mechanisms using a dedicated hash-function, 2011
JILGuidance	Joint Interpretation Library, Guidance for smartcard evaluation, Version 2.0, February 2010
NIST-SP800-38A	NIST, SP800-38A Recommendation for Block Cipher Modes of Operation: Methods and Techniques
NIST-SP800-38B	NIST, SP800-38B Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, May 2005
NIST-SP800-38C	NIST, Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality, May 2004
NIST-SP800-38D	NIST, SP800-38D Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007
NIST-SP800-38F	NIST , SP800-38F Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping, 2012
NIST-SP800-56C	NIST, Recommendation for Key Derivation through Extraction-then-Expansion, Special Publication SP800-56C, November 2011

PKCS#1	PKCS #1 v2.2: RSA Cryptographic Standard, https://www.emc.com/emc-plus/rsa-labs/pkcs/files/h11300-wp-pkcs-1v2-2-rsa-cryptography-standard.pdf, 27.10.2012
RFC2104	RFC2104, HMAC: Keyed-Hashing for Message Authentication
RFC5639	RFC5639, Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation, http://www.ietf.org/rfc/rfc5639.txt, 2010
RFC5903	RFC5903, Elliptic Curve Groups modulo a Prime (ECP Groups) for IKE and IKEv2
RFC6954	RFC6954, Using the Elliptic Curve Cryptography (ECC) Brainpool Curves for the Internet Key Exchange Protocol Version 2 (IKEv2)
SOGIS IT-TDs	SOG-IS, Recognition Agreement Management Committee Policies and Procedures, SOGIS IT-Technical Domains, February 2011
TPMLib,Part 1	Trusted Platform Module Library, Part 1: Architecture, Family “2.0”, Level 00, Revision 01.38, September 29, 2016
TR-03110	BSI, Technical Guideline TR-03110 Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token – Part 2 - Protocols for electronic IDentification, Authentication and trust Services (eIDAS), Version 2.21, 2016
TR-03111	BSI, Elliptic Curve Cryptography, BSI Technical Guideline TR-03111, Version 2.1, 1.6.2018
TR-03151	BSI, Technical Guideline TR-03151 Secure Element API (SE API), Version 1.0, 5. Juni 2018

Keywords and Abbreviations

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.
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 [1], paragraph 97)
update code package	code if implemented changing the TOE implementation at the end of the TOE life time

Table 15: Glossary

Acronym	***Term***
A.xxx	Assumption
CC	Common Criteria
CSP	cryptographic service provider
ECC	Elliptic curve cryptography
HMAC	Keyed-Hash Message Authentication Code
KDF	Key derivation function
MAC	Message Authentication Code
n. a.	Not applicable
NVM	Non-volatile memory
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
SMAERS	Security Module Application for Electronic Record-keeping Systems
T.xxx	Threat
TOE	Target of Evaluation
TSF	TOE security functionality
UCP	update code package

Table 16: Abbreviations