TÜBİTAK BİLGEM UEKAE

NATIONAL RESEARCH INSTITUTE OF ELECTRONICS AND CRYPTOLOGY

eID Technologies Unit

AKiS v2.5.2N

SECURITY TARGET LITE

Revision no	01
Revision date	27.05.2019
Document code	AKiS-252N-STL-01
Prepared by	eID Technologies Unit
Approved by	AKiS Project Manager

REVISION HISTORY

Revision #	Revision Reason	Date
01	First version	27.05.2019

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ST INTRODUCTION

1.1 ST REFERENCE

Title: Security Target Lite of AKiS v2.5.2N

Document Version: 01

CC Version: 3.1 (Revision 4)

Assurance Level: EAL 4+ AVA_VAN.5 and ALC_DVS.2

1.2 TOE REFERENCE

The current Security Target refers to the product AKiS v2.5.2N.

1.3 TOE OVERVIEW

1.3.1 TOE TYPE AND TOE COVERAGE

AKiS v2.5.2N contact based smartcard is a composite product consisting of Embedded Operating System, platform crypto library (platform library) and the platform security IC (platform IC). The crypto library is evaluated as a composite product consisting of crypto library and security IC NXP Technologies, SmartMX3 P71D320P. Detailed information for the platform is given in Section 2.

The TOE consists of

- AKiS v2.5.2N Embedded Operating System,
- IC dedicated software (test and support software including relevant libraries),
- IC dedicated crypto library,
- security IC,
- guidance documentation,
- activation data.

The coverage of the composite TOE defined in this Security Target is shown in Figure 1 as a shaded region. Platform library is a composite product using some of the features of the platform. AKiS v2.5.2N covers the features from both the platform library and the platform IC.

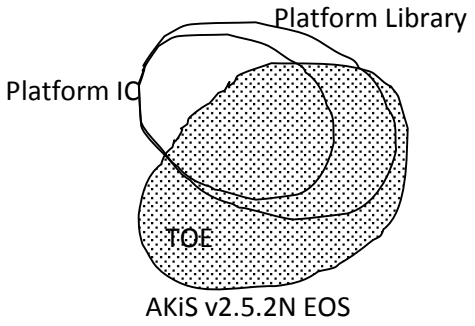


Figure 1: Coverage of the composite TOE

1.3.2 MAJOR SECURITY PROPERTIES OF THE TOE

The TOE provides the following services to the application:

- Protection against modification, probing, environmental stress and emanation attacks mainly by platform specification and Embedded Operating System support as detailed in Section 8
- Access control to services and data by using role attribute, PIN-knowledge attribute, activation agent authentication status, personalization agent authentication status, initialization agent authentication status and device authentication status.
- The following identification and authentication services:
    - activation agent identification & authentication by asymmetric cryptographic verification,
    - initialization and personalization agent identification & authentication by symmetric decryption,
    - terminal and chip identification & authentication by certificate authentication,
    - role identification & authentication by certificate authentication,
    - user identification & authentication by PIN verification.
- Security management, for services and data by supporting activation agent, initialization agent and personalization agent roles, and any other roles defined by the application.
- Secure messaging services between TOE and the terminal.
- The following cryptographic services:
    - SHA Operation,
    - AES Operation,
    - MAC, Retail-MAC and CMAC Operation,
    - TDES Operation,
    - signature generation PKCS#1 v1.5,

- signature generation PKCS#1 v2.1,
- signature generation ISO/IEC 9796-2 Scheme 1,
- signature generation ECDSA,
- signature verification ISO/IEC 9796-2 Scheme 1[1],
- asymmetric decryption PKCS#1 v1.5,
- asymmetric decryption PKCS#1 v2.1,
- asymmetric encryption/decryption RAW RSA[2],
- RSA key pair generation,
- ECC key pair generation,
- random number generation.

Note 1: The cryptographic functionality of AKiS v2.5.2N Embedded Operating System includes SHA-1, SHA-256, SHA-384, SHA-512 and RSA 1024-to-2816 bit operations. However, SHA-1 and RSA-1024 operations are not included in the scope of evaluation because of security considerations.

1.3.3 THE USAGE OF THE TOE

The TOE is designed and developed to be as a platform for smart card applications. It supports the life cycle requirements of the smart card applications and provides security services to the smart card applications.

AKiS v2.5.2N supports two different configurations to the application owner:

- chip configuration,
- SAM configuration.

Chip configuration is developed to act as user card application like eIDs. The SAM configuration is developed to act on behalf of the terminal as a secure access module.

TOE has security features as detailed in Section 1.3.2, for both configurations. But, there is a slight difference between two configurations in their secure messaging properties.

In chip configuration, two secure messaging types are performed.

1 No interface for ISO/IEC 9796-2 Scheme 1 signature generation and verification is present. The services start during secure messaging automatically.

2 No interface for RAW RSA encryption/decryption is present. The service starts during secure messaging automatically.

The first one is mutual authentication between card (chip) and the terminal by certificate exchange. In this method, both the terminal and the card possess a public key certificate and the corresponding private key. They share their trusted public keys with each other by certificate exchange procedure. Next, they agree on secure messaging keys by key agreement procedure. Finally secure messaging starts. This secure messaging starts in each mutual authentication automatically.

In the second method, a random data is generated by the terminal and sent to TOE confidentially. Next, using this random data, card and the terminal agree on the secure messaging keys by key agreement procedure. Finally, TOE starts secure messaging. Public key cryptography is used in each step of the key agreement process to ensure confidentiality. No certificate is needed in this method.

In SAM configuration, only the second method is performed.

The other difference between the two configurations is in the terminal authentication method. Chip configuration provides terminal authentication by internal and external authentication with certificate exchange. But in SAM configuration, it is provided by PIN authentication. By this way, “authenticated terminal” means PIN authenticated terminal for SAM configuration.

1.4 REQUIRED NON-TOE HW/ SW/ FIRMWARE AVAILABLE TO THE TOE

None.

1.5 TOE DESCRIPTION

1.5.1 LOGICAL VIEW

The logical view of the TOE is given in Figure 2. Logicaly, TOE consists of the communication subsystem, command subsystem, cryptographic support subsystem, security subsystem, memory and file subsystem and system subsystem.

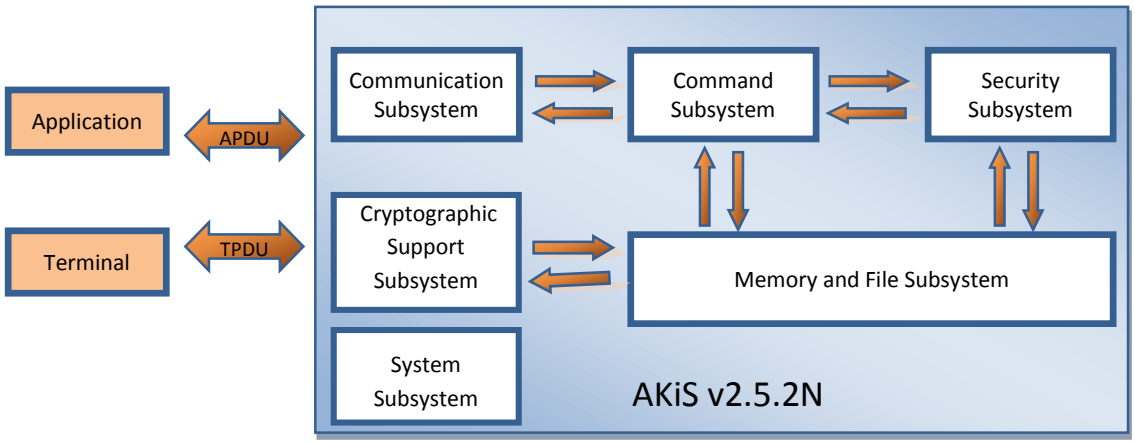


Figure 2: AKiS v2.5.2N Logical View

Communication Subsystem

Communication subsystem manages the communication between AKiS v2.5.2N and the external world. Two layered communication takes place between the outer world and AKiS v2.5.2N, for the transmission purposes T=1 protocol is implemented, for the application purposes APDU packets are used [9].

Command Subsystem

Command subsystem processes the commands received from communication subsystem. It performs the commands via help of the security subsystem, memory and file subsystem.

Cryptographic Support Subsystem

All cryptographic functions like encryption, decryption, signature generation, signature verification, random number generation, hash calculation are performed within this subsystem.

Security Subsystem

Access control conditions and lifecycle management operations are performed within this subsystem. Whenever a security control is to be done via command subsystem, it asks to the security subsystem if the action is allowed or not.

Memory and File Subsystem

Memory and file subsystem manages the non-volatile memory of the security IC. Memory and file subsystem gives services to both of the command subsystem and the security subsystem.

System Subsystem

System subsystem includes the functions related to the whole system such as security controls of the system.

1.5.2 PHYSICAL VIEW

Physical view of the TOE is given in the platform information.

1.5.3 INTERFACES

For the electrical I/O:

- ISO 1177 - Information Processing Character Structure For Start/Stop And Synchronous Character Oriented Transmission [8].

For the transmission:

- ISO 7816-3 Information Technology – Identification Cards – Integrated Circuits with Contacts Part 3: Electronic Signals and Transmission Protocols - T=1 Protocol [9].

For the application:

- ISO 7816-4 Information Technology – Identification Cards – Integrated Circuits with Contacts Part 4: Organization, security and commands for interchange [10].
- ISO 7816-8 Information Technology – Identification Cards – Integrated Circuits with Contacts Part 8: Commands for security operations [10].

- ISO 7816-9 Information Technology – Identification Cards – Integrated Circuits with Contacts Part 9: Commands for card management [12].

1.5.4 LIFE CYCLE

AKiS v2.5.2N is a composite product of Security IC and Embedded Operating System. Being a smart card application, TOE has a similar life cycle as defined IC PP [1].

There are slight differences for composite TOE. The first one, delivery of composite TOE is performed after phase 5. Also, additional sub phases are defined for composite TOE.

A brief overview is given below for common phases which are detailed in IC PP [1]. Although TOE delivery refers to “after Phase-5”, due to configuration needs after TOE delivery, Phase-6 is divided into sub phases that are described in the following section subsection.

Life Cycle Phases:

Phase-1:

- Security IC embedded software, or, Embedded Operating System, development.

Phase-2:

- IC development:
  - IC design,
  - IC dedicated software development.

Phase-3:

- IC manufacturing:
  - integration and photo mask fabrication,
  - IC production,
  - IC testing.

Phase-4:

- IC Packaging.
  - Security IC packaging (and testing).

Phase-5:

- Composite product integration.

Phase-6:

- Personalization:
  - the composite product personalization and testing stage where the user data of the composite TOE is loaded into the security IC's memory.

Phase-7:

- Operational phase:
  - the composite product usage by its issuers and consumers which may include loading and other management of applications.

Sub Phases of Phase 6 and Additional Phase Defined for Embedded Operating System

Phase-6 is separated into three following sub-phases by Embedded Operating System:

- activation sub-phase,
- initialization sub-phase,
- personalization sub-phase,

Additionally, “death phase” is added.

Activation Sub-phase:

The TOE, AKiS v2.5.2N, is activated and, at the same time, initialization key and personalization key are loaded in this phase. The TOE accepts only PERFORM SECURITY OPERATION (PSO) command, activation command and some commands that provide very limited information about the TOE in this phase. Before the activation command, activation agent is to transfer activation public key, in the same session, to the TOE via PSO: VERIFY CERTIFICATE command. Managed by activation agent, this phase is ended by activation operation in which a 2048 bit cryptogram created using activation private key is sent to the TOE via EXCHANGE CHALLENGE command. If the signature of cryptogram is verified successfully, activation is completed and the composite TOE (card) becomes ready for initialization.

Initialization Sub-phase:

This phase starts by successful authentication of initialization key. Another successful authentication is needed to complete this phase. File architecture is created by initialization agent. Application data also might be written and access rules might be defined in this phase. Commands listed in Table 1 can be used by initialization agent. Initialization agent can perform any operation by using these commands. Application specific restrictions cannot be implemented in initialization sub-phase.

Initialization operations must be performed in a secure environment.

Table 1: Commands used in the initialization and personalization phases

#	Commands
1.	KART TEST
2.	EXCHANGE CHALLENGE
3.	INITIALIZATION[3]
4.	PERSONALIZATION[4]
5.	CHANGE KEY
6.	FORMAT
7.	ERASE BINARY
8.	DIR
9.	DELETE SDO
10.	GET DATA
11.	PUT DATA
12.	GET RESPONSE
13.	GET CHALLENGE
14.	SELECT FILE
15.	CREATE FILE
16.	DELETE FILE
17.	UPDATE BINARY
18.	READ BINARY
19.	APPEND RECORD
20.	UPDATE RECORD
21.	READ RECORD
22.	GENERATE ASYMMETRIC KEY PAIR
23.	TERMINATE CARD USAGE

3 Applicable only in the initialization sub-phase
4 Applicable only in the personalization sub-phase

Personalization Sub-phase:

This phase starts by successful authentication of personalization key. Another successful authentication is needed to complete this phase. Personal information data are written and access rules are defined in this phase. Listed commands in Table 1 can be used by personalization agent. Personalization agent can perform any operation by using these commands. Application specific restrictions cannot be implemented in personalization sub-phase.

Personalization operations must be performed in a secure environment.

Death Phase:

Death phase is defined by Embedded Operating System. TOE becomes out of order and can't be used as a legitimate one. TOE enters this phase because of unsuccessful authentication attempts during activation, initialization and personalization. In addition, some critical integrity errors in operational stage cause death phase. In this phase, TOE doesn't accept any command, but the ones that provide limited information about TOE.

2 PLATFORM INFORMATION

2.1 PLATFORM IDENTIFICATION

Platform:
NXP Technologies, SmartMX3 P71D320P

Platform ST:
NXP Secure Smart Card Controller N7021 VA Security Target Lite, Rev. 1.1, 2017-05-31
Crypto Library Cobalt on N7021 VA Security Target Lite, Rev. 1.1, 5 July 2017

Platform PP Conformance:
Security IC Platform Protection Profile with Augmentation Packages Version 1.0, Registered and Certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-CC-PP-0084-2014

Platform Assurance Level:
EAL 6+ ALC_FLR.1

Platform Certification Report:
BSI-DSZ-CC-0977-2017 for NXP Secure Smart Card Controller N7021 VA including IC Dedicated Software from NXP Semiconductors Germany GmbH
BSI-DSZ-CC-1019-2017 for Crypto Library Cobalt on N7021 VA from NXP Semiconductors Germany GmbH

Common Criteria Version:
CC v3.1 Revision 4

2.2 PLATFORM DESCRIPTION

2.2.1 PRODUCT SPECIFIC FEATURES

- High-performance dual-Interface secure microprocessor
- Top-level cryptography engines with "full key length" support
  - Dedicated cryptography functional unit for symmetric DES and AES algorithms
  - 112-bit (two-key) and 168-bit (three-key) triple-DES (TDES or 3DES), in various configurations
  - AES with 128, 192 and 256-bit key length
  - Asymmetric cryptography accelerator unit, supporting RSA, ECC and related algorithms
  - RSA cryptography with arbitrary key length up to 4096 bits
  - Elliptic-curve cryptography (ECC) with key length up to 640 bits
- True Random Number Generator, compliant to AIS31
- Deterministic Random Number Generator for faster execution in cases where lower RNG entropy is sufficient
- Cyclic redundancy check (CRC) functional unit for 16 and 32-bit operation
- Large memory for operating system design flexibility
- NXP FlexMem approach
- Secure bootloader for initial loading or updates of Flash memory; suitable for use in secure manufacturing sites as well as in general environments. Various configuration options exist to manage and delegate rights for access and writing.
- Vertical Firewall technology
- Dual-interface support with wide configuration range

Security features

- 90nm CMOS technology offers strong inherent protection against invasive attacks on logic and memories
- NXP Glue Logic concept effectively de-correlates the function and location of circuitry on the device: no functional blocks are recognizable in any physical layer of the device, adding another level of protection against active and passive invasive attacks
- No use of logical hardmacro blocks; all logics in the device - including CPU, coprocessors and all other functions - are synthesized into a single glue logic area.

Functional diagram

The diagram provides a generic overview of the architecture of the SmartMX3 P71 product family. Functional blocks, pins and connections shown in this diagram are optional and represent a super-set of those elements actually implemented in a real product.

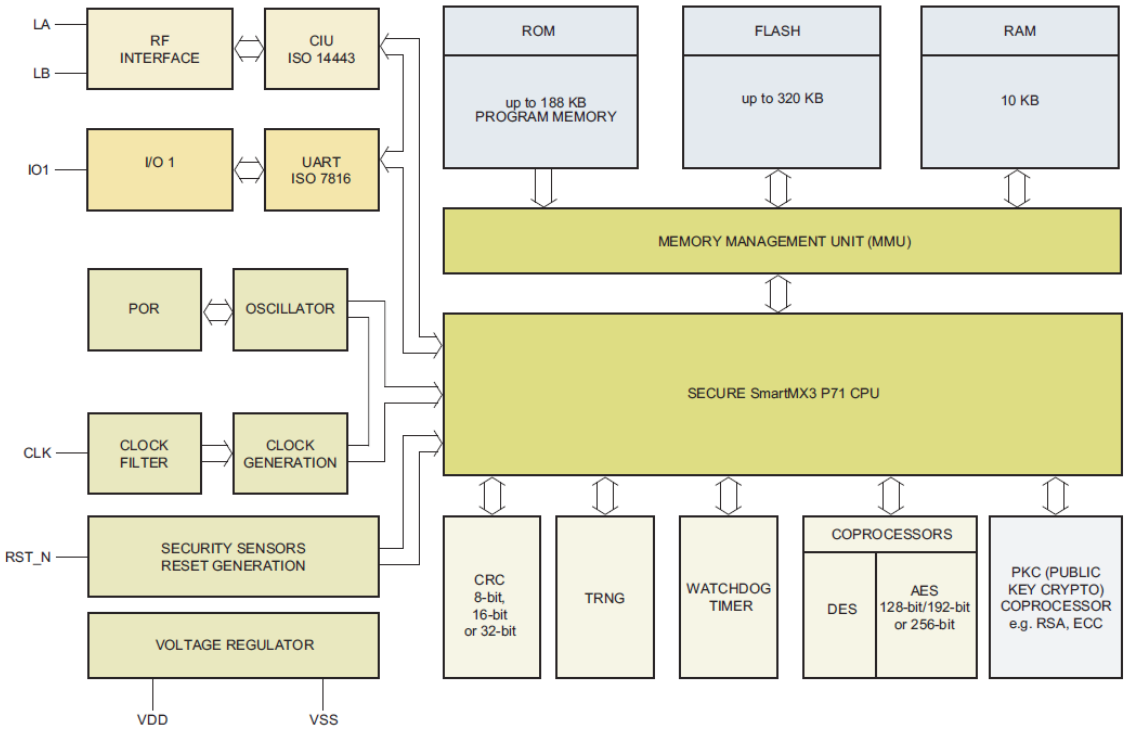


Figure 3 : Functional Diagram of the SmartMX3 P71 product family

2.2.2 CRYPTO LIBRARY

Symmetric Cipher Library

The Configurable Symmetric Cipher library component supports:

- AES encryption and decryption with key length 128, 192 and 256 bit
-
- 3-DES encryption and decryption using two single-DES keys
- 3-DES encryption and decryption using three single-DES keys
- ECB mode
- CBC mode

• **RSA Library**

The following functions are included within the RSA component (implemented according to [22]):

- RSA public key operation
- RSA private key (in CRT format) operation
- RSA calculation of public exponent from an RSA CRT private key

The supported key length for the public modulus n is between 512 and 4096 bits.

RSA Key Generation Library

RSA Key Generation (in CRT format) is included within the RSA component. The supported key length for the public modulus n is between 512 and 4096 bits.

ECC Library

The following functions are included within the ECC over GF(p) component (implemented according to [26] and [27]):

- ECDSA Signature Generation
- EC Key generation

The supported elliptic curve key length is between 128 and 640 bits.

SHA Library

The SHA library component implements hashing according to the standard [16]. It supports the algorithms SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 for messages of length equal to an integer number of bytes, up to a maximum length of 2^{61}-1 bytes in length for SHA-1, SHA-224 and SHA-256, and up to a maximum length of 2^{125}-1 bytes in length for SHA-384 and SHA-512.

RNG Library

The RNG library component implements pseudo-random number generation according to the NIST SP 800-90A specification (see [21]). The block cipher operations can be selected by the user to run in either DES (3-Key TDEA) or AES (128, 192 or 256) mode.

Hash Library

The Hash library component implements a common interface to the hashing algorithms provided by the hashing components.

3 CC CONFORMANCE CLAIM

This security target claims conformance to

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2012-09-001, Version 3.1, Revision 4, September 2012.
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2012-09-002, Version 3.1 Revision 4, September 2012.
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components; CCMB-2012-09-003, Version 3.1 Revision 4, September 2012.

As conformance claim is as follows:

- part 2 extended,
- part 3 conformant.

3.1 PP CLAIM

This ST does not claim conformance to any PP.

3.2 PACKAGE CLAIM

The current ST is conformant to the following security requirements package: assurance package EAL4 augmented with AVA_VAN.5 and ALC_DVS.2 as defined in the CC, part 3.

4 SECURITY PROBLEM DEFINITION

The TOE is the Embedded Operating System (EOS) with NXP secure crypto library on SmartMX3 (platform library) on NXP chip, P71D320P (platform IC, or the security IC). Hence application (eID application, banking application etc.) is not part of the TOE; it does not have user data and TSF data belonging to the application. But it provides containers for storing files, keys and PINs, and functionality to manage these entities to the application.

4.1 ASSETS

AKiS v2.5.2N is the composite product consisting of the Embedded Operating System, crypto library (platform library) and the security IC (platform IC). Since the security target of the platform IC and platform library claims strict conformance to the PP [1] the assets defined in section 3.1 of the protection profile apply to this Security Target.

Additional assets are defined below.

4.1.1 PRIMARY ASSETS

Primary assets represent user data in the sense of the CC. They are given in Table 2.

Table 2: Primary assets of the TOE

Asset Name	Definition	Protection Need
Files (user data stored)	All files that is provided to the application to store data	Confidentiality Integrity
User data transferred	All data transferred between TOE and external entities	Confidentiality Integrity

4.1.2 SECONDARY ASSETS

Secondary assets include TSF and TSF data of the TOE. They are given in Table 3.

Table 3: Secondary assets of the TOE

Asset Name	Definition	Protection Need
PINs	TOE should provide PIN verification mechanism to the application but it does not have natively PINs. As part of the PIN verification mechanism, PINs are stored in the containers that is provided by TOE and transferred by the TSF mechanisms. Therefore, confidentiality and integrity of the PINS are satisfied by both TOE and the application.	Confidentiality Integrity
Keys	Applications might need keys for their security functionality. TOE should provide containers to the application to store and manage them securely. Namely, confidentiality and the integrity of the keys are satisfied by TOE and the application.	Confidentiality Integrity
Access reference rules file	This is the file to be created by the application that arranges access control to the assets and to the TSF Interface. The integrity need of this file is different than the standard file (user data stored). Thus this is regarded as a different asset.	Confidentiality Integrity
Activation data	These are the data used in the activation agent authentication.	Confidentiality Integrity
Initialization and personalization data	These are the data used in authentication of initialization and personalization agents.	Confidentiality Integrity
SAM or chip PubK	SAM or Chip Public Keys (PubK) are used to verify the root CA certificates	Integrity
SAM or chip PrK	The SAM or Chip Private Keys (PrK) are used to prove the authenticity of the TOE.	Confidentiality Integrity
SAM or chip CA certificate	Root CA Certificate is the root certificate to be used to validate certificate chains.	Integrity

Asset Name	Definition	Protection Need
SAM or chip certificate	The SAM or Chip Certificates are used to prove the authenticity of SAM or Chip Public Key. They are signed by CA certificate.	Integrity
IC identification data	It is the data to uniquely identify the TOE.	Integrity
EOS code	TSF code is the EOS code that is in operation and in storage. For the proper function of TOE, integrity, confidentiality of the TSF Code must be protected. Also its correct operation must be maintained.	Integrity, confidentiality
Security services	The TOE provides security services to the application. Correct operation of the cryptographic operations is essential for the application that the TOE serves for.	Correct Operation
Files (as TSF data)	The TOE provides data containers to the application, these data containers can be used as TSF data by the application. So, TOE might include files as TSF Data in addition to other TSF data.	Confidentiality, integrity
Genuineness of TOE	Genuineness of the SC shows that it is neither copied nor cloned.	Availability

4.2 SUBJECTS AND EXTERNAL ENTITIES

This ST considers the external entities and subjects given in Table 4.

Table 4: Subjects and external entities of the TOE

Entity	Subject	Definition
Activation agent	+	Activation agent is the entity who activates the card and writes the configuration data, initialization and personalization data to the TOE.
Initialization agent	+	Initialization agent is the entity who initializes the TOE.

Entity	Subject	Definition
Personalization agent	+	Personalization agent is the entity who personalizes the TOE.
Terminal	+	The entity that card communicates with.
Application defined role	+	Any agent defined by application developer. Application developer may be thought as Initialization and personalization agent.
Card holder	-	Card holder is whom the card is issued for. It is the owner of the Chip Card.
IC developer	-	The entity that designs the IC and develops the IC Dedicated Software.
EOS developer	-	The entity that designs and develops the EOS.
Application developer	-	The entity that designs and develops the application.
IC manufacturer	-	The entity that performs the following activities: • production of the Integrated circuit, • testing the Integrated circuit, • EOS is loaded to the NVM of the IC. Flash loader mechanism is not disabled by the IC manufacturer, • writes the configuration data and IC serial number.
Card Issuer	-	The entity holding the authority to issue the cards. Card issuer employs the application developer to develop the application that fulfills its needs. After the application is developed and the TOE is received, card issuer may separate its authority to the following roles: activation agent, initialization agent and personalization agent and delegate these roles to other entities or perform them by itself.

Entity	Subject	Definition
Certificate authorities (Root CA, chip CA, terminal CA, role CA)	-	Certificate authorities are the entities which issue the certificates. Chip CA and terminal CA (valid for chip configuration) certificates are signed by the root CA.
Attacker	-	A threat agent trying to violate the system security policy. Attacker may have at most high attack potential.

4.3 THREATS

4.3.1 HARDWARE RELATED THREATS

Threats related to hardware are defined and explained in the platform protection profile [1] and also given in Table 5.

Table 5: Hardware related threats

Threat	Definition
T.Phys-Tamper	An attacker may perform physical probing of the TOE in order (i) to disclose user data or (ii) to disclose/reconstruct the TOE's Embedded Operating System or (iii) to disclose other critical information about the operation of the TOE. An attacker may physically modify the TOE in order to alter (i) its security functionality (hardware and software part, as well), (ii) the user data or the TSF-data stored on the TOE.
T.Information_Leakage	An attacker may exploit information leaking from the TOE during its usage in order to disclose confidential user data and/or TSF-data. The information leakage may be inherent in the normal operation or caused by the attacker.
T.Malfunction	An attacker may cause a malfunction of the TOE's hardware and Embedded Operating System by applying environmental stress in order to (i) deactivate or modify security features or functionality of the TOE's

Threat	Definition
	hardware or to (ii) circumvent, deactivate or modify security functions of the TOE's Embedded Operating System. This may be achieved, e.g., by operating the TOE outside the normal operating conditions, exploiting errors in the TOE's Embedded Operating System or misusing administrative functions. To exploit these vulnerabilities, an attacker needs information about the functional operation of the TOE.
T.Abuse-Func	An attacker may use functions of the TOE which may not be used after the delivery of the TOE in order (i) to manipulate or to disclose the user data stored in the TOE, (ii) to manipulate or to disclose the TSF-data stored in the TOE or (iii) to manipulate (explore, bypass, deactivate or modify) security functionality of the TOE. This threat addresses the misuse of the functions for the initialization and personalization in the operational phase after delivery to the TOE holder.

4.3.2 ADDITIONAL THREATS DUE TO COMPOSITE TOE SPECIFIC FUNCTIONALITY

In this section, threats due to composite TOE specific functionality are defined. These threats are terminal and communication related threats, and card cloning and forgery related threats. They are given in Table 6 and Table 7 respectively.

Table 6: Terminal and communication related threats

Threat	Definition
T.Eavesdropping	An attacker may monitor the communication between the TOE and the terminal/card reader to get unauthorized access to the user data and/or TSF Data.
T.Session_Hijacking	An attacker may wait until the identification and authentication process is completed and session is established between the TOE and the terminal. After the session is established, attacker may take out the TOE or the terminal from the communication channel and takes over. That way attacker bypasses the identification and authentication process and accesses to services illegitimately.

Threat	Definition
T.Man_in_The_Middle	An attacker may alter the communication between the TOE and the terminal. An attacker listens and alters the connection between the TOE and the terminal in order to access the services that he or she is unauthorized to access.
T.Skimming	The terminal which obtains smart card's interactions with the world by controlling all I/O's can observe user identification data, so this terminal must be trusted not to capture the user's identification data. Concerning a variety of fake-terminal attacks become possible, in these cases the user must be able to differentiate between "real devices" that are manufactured by a trusted party and between "fake devices" that are manufactured by the attackers. The user cannot identify that the terminal has hidden features, for example the message they sign was not altered by a malicious terminal. The security has nothing to do with the smart card/ terminal exchange; it is the back-end processing system that monitors the card.

Table 7: Card cloning and forgery related threats

Threat	Definition
T.Counterfeit	An attacker produces an unauthorized copy or reproduction of a genuine TOE to be used as part of a counterfeit operation. He or she may generate a new data set or extract completely or partially the data from a genuine TOE and copy them on another functionally appropriate chip to imitate this genuine TOE. This violates the genuineness of the TOE being used either for authentication of a Card presenter as the Card holder.
T.Unauthorised_Access	An attacker may access to data that he or she is not authorized to.
T.Unauthorised_Management	An attacker may illegitimately use the security management services of the TOE.

4.4 ORGANISATIONAL SECURITY POLICIES

Organizational security policies of the composite TOE is given in Table 8.

Table 8: Composite TOE policies

#	Policy Name	Definition
1.	P.Identification_and_Authentication	The TOE shall support • chip authentication, • terminal authentication, • PIN verification, • role holder authentication and any combination of this. In addition, TOE shall calculate the cryptographic checksum value of the Embedded Operating System HEX code in the flash memory code area and return it upon request, and each instantiation of the TOE shall include a unique identification.

2.	P.PKI	There will be Certificate Authorities (CA's) for terminal authentication, chip authentication, and role authentication and the certificates for these CA's will be signed by Root CA. Terminal certificates, chip certificates, and role certificates will be signed by the corresponding CA.
3.	P.Access_Control	Role attribute, PIN knowledge attribute, device authentication attribute of the user shall be used as a security attribute to determine the access control behavior and security management privileges during operational phase. No memory separation is required in the operational phase (the TOE is a single application EOS), the access control is rather file permission based. However, memory separation is required in between the memory areas of IC dedicated software and the EOS which is supported by the platform. Another security feature related to access control and not derived from the threats is access to Special Function Registers and hardware resources. Access control policy for the access to the SFRs and hardware resources by System Mode and the EOS code (executing in User Mode) shall be applied such that EOS gains access to resources via the NXP System Mode.
4.	P.PreOperational_Security_Management	The TOE shall support • activation agent, • initialization agent, • personalization agent functions and roles. Personalization software shall handle the user data considering their integrity as stated in the Guidance Documents.

5.	P.Operational_Security_Management	The TOE shall support any management function and role defined by the application. Software accessing the TOE in the operational phase shall check the integrity of the user data stored in the TOE in each read operation as described in the Guidance Documents.

6.	P.Cryptographic_Operations	The TOE shall support following cryptographic functions:
		• RSA key pair generation,
		• ECC key pair generation,
		• hash calculation,
		• eSign operations,
		  ◦ PKCS #1 v2.1 PSS,
		  ◦ PKCS #1 v1.5,
		  ◦ ISO/IEC 9796-2 Scheme 1,
		  ◦ ECDSA
		• asymmetric decryption,
		  ◦ PKCS #1 v2.1 OAEP,
		  ◦ PKCS #1 v1.5,
		  ◦ Raw RSA
		• asymmetric encryption,
		  ◦ Raw RSA
		• TDES operation,
		• AES operation,
		• MAC, Retail-MAC and CMAC operation,
		• Destruction of the keys used

7.	P.Process-TOE	An accurate identification must be established for the TOE. This requires that each instantiation of the TOE carries this unique identification. This also results in a unique activation cryptogram for each TOE.

4.5 ASSUMPTIONS

Assumptions for the operational environment of the composite TOE is given in Table 9.

Table 9: Composite TOE assumptions

#	Assumption Name	Definition
1.	A.Secure_Application	It is assumed that the application correctly defines the access rules of the application data.
2.	A.Key_and_Certificate_Security	It is assumed that all keys and certificates are produced, stored and used securely outside of TOE.
3.	A.PIN_Handling	It is assumed that PINS belonging to the application are handled securely by PIN owner.
4.	A.Personnel_Security	It is assumed that personnel who hold privileges over the TOE acts responsively and according to the application requirements.
5.	A.Trusted_Parties	It is assumed that the authenticated parties that the TOE communicates act responsively.
6.	A.Pre-Operational_Environment	It is assumed that the Physical environments of initialization and personalization phases are secure.

5 SECURITY OBJECTIVES

5.1 SECURITY OBJECTIVES FOR THE TOE

The TOE is the composite product consisting of the Embedded Operating System, the platform library and the platform IC. The platform IC and the Embedded Operating System have different interfaces to the external world. The platform has the physical and electrical interfaces and the Embedded Operating System has the logical interfaces. Therefore, the attacks done through the physical and electrical interfaces are mostly countered by the platform and the attacks performed through logical interfaces are countered by the Embedded Operating System.

Objectives for the embedded software (platform library and the Embedded Operating System) are defined in Table 10.

Table 10: Embedded software objectives

Objective	Definition
OT.Identification_and_Authentication	The TOE shall support (i) activation agent authentication, (ii) initialization agent authentication, (iii) personalization agent authentication, (iv) chip authentication, (v) terminal authentication[5], (vi) role certificate holder authentication and (vii) PIN verification. In addition, each instantiation of TOE shall include a unique identification number. The TOE shall support means to check FabKey data when the very first APDU command is received in the lifetime of the TOE. The TOE shall also provide means to update the EOS in its non-volatile memory for which all the security requirements of the platform are fulfilled.
OT.Access_Control	The TOE shall control the access to the user data and security services according to access control rules

5 Provided by PIN authentication for SAM Configuration.

Objective	Definition
	determined by the application. Role attribute, PIN-knowledge attribute, device authentication status and authentication should be used as security attributes during the decision of access permission.
OT.Security_Management	The TOE shall support (i) activation agent role, (ii) initialization agent role, (iii) personalization agent role, and (iv) any other roles defined by the application.
OT.Cryptographic_Operations	The TOE shall perform following: (i) asymmetric key pair generation, (ii) random number generation, (iii) hash calculation, (iv) eSign operations, (v) symmetric cryptographic operations and (vi) destruction of the keys used (all of the operations should be secure against side channel attacks)
OT.Prot_Malfunction	The TOE must ensure its correct operation. The TOE must prevent its operation outside the normal operating conditions where reliability and secure operation have not been proven or tested. This is to prevent functional errors in the TOE.
OT.Prot_Phys-Tamper	The TOE must provide protection of confidentiality and integrity of the user data, the TSF-data and the Embedded Software by means of • measuring through galvanic contacts representing a direct physical probing on the chip's surface except on pads being bonded (using standard tools for measuring voltage and current) or • measuring not using galvanic contacts, but other types of physical interaction between electrical charges (using tools used in solid-state physics research and IC failure analysis), • manipulation of the hardware and its security functionality, as well as

Objective	Definition
	• controlled manipulation of memory contents (user data, TSF-data) with a prior • reverse engineering to understand the design and its properties and functionality.
OT.Prot_Inf_Leak	The TOE must provide protection against disclosure of confidential user data and/or TSF-data stored by the IC • by measurement and analysis of the shape and amplitude of signals or the time between events found by measuring signals on the electromagnetic field, power consumption, clock, or I/O lines, • by forcing a malfunction of the TOE and/or • by a physical manipulation of the TOE.
OT.Prot_Abuse-Func	The TOE must prevent the abuse of the functions of the TOE, which may not be used in TOE operational phase, in order (i) to manipulate or to disclose the user data stored in the TOE, (ii) to manipulate or to disclose the TSF-data stored in the TOE, (iii) to manipulate (bypass, deactivate or modify) security functionality of the TOE.
OT.Chip_Auth_Proof	The TOE must enable the terminal connected to verify the authenticity of the TOE as a whole device as issued by the issuer by means of internal and external authentication.
OT.Secure_Communication	The TOE shall support secure communication with the terminal. TOE supports encryption, integrity and authenticity protection against attacks during communication between TOE and terminal.
OT.Storage_Integrity	TOE shall support storage integrity protection for User Data and TSF data.

5.2 SECURITY OBJECTIVES FOR OPERATIONAL ENVIRONMENT

Objectives for the operational environment are defined in Table 11.

Table 11: Operational environment objectives

Objective	Definition
OE.PKI	There shall be terminal authentication CA, chip authentication CA, Role CA all of which certificates are signed by Root CA. Terminal Certificates, Chip Certificates and Role Certificates must be signed by the corresponding CA.
OE.Secure_Application	Application shall correctly define the access rules of the application data. Also application should fulfill the security requirements of EOS as described in [13].
OE.Key_and_Certificate_Security	Key creation and storage outside of the TOE shall be handled securely.
OE.PIN_Handling	PIN Creation and usage by Card Holder shall be handled securely.
OE.Personnel_Security	The personnel who have privileges (EOS developer, activation agent, initialization agent and personalization agent) shall have necessary security clearances and shall act responsibly.
OE.Responsible_Parties	The parties that the TOE communicates (sends or receives data; and/or receives or gives services) should act responsively. For example, terminal should protect any data against confidentiality integrity attacks after taking TOE.
OE.Pre-Operational_Env_Sec	Physical environment of initialization and personalization phases shall be secure.
OE.User_Data_Handling	The operational environment, i.e. activation software, initialization software and personalization software and the software accessing the TOE in the operational sub-phase shall handle the user data considering the integrity means as defined in the Guidance Documents.

5.3 SECURITY OBJECTIVES RATIONALE

The justification related to the threats “Physical Tampering (T.Phys-Tamper)”, “Information Leakage (T.Information_Leakage)”, “Malfunction (T.Malfunction)”, and “Abuse of Functionality (T.Abuse-Func)” is given below.

For all threats, the corresponding objectives in Section 5.1 are stated in a way, which directly corresponds to the description of the threat in Section 4.3.1. It is clear from the description of each objective that the corresponding threat is removed. More specifically, in every case the ability to use the attack method successfully is countered by the objective.

Removal of T.Phys-Tamper and T.Malfunction is also supported by additional objectives as detailed below:

T.Phys-Tamper is mainly removed by OT.Prot_Phys-Tamper.

T.Malfunction is mainly removed by OT.Prot_Malfunction. OT.Storage_Integrity supports by detecting integrity anomalies and acting.

T.Eavesdropping is countered by OT.Secure_Communication.

T.Session_Hijacking is countered by OT.Secure_Communication.

T.Man_in_The_Middle is countered by OT.Secure_Communication.

T.Skimmming is countered by OT.Identification_and_Authentication and OT.Prot_Phys-Tamper as they provide protection against physical manipulation of authenticity verification key.

T.Counterfeit is countered by OT.Identification_and_Authentication, OT.Prot_Phys-Tamper, OT.Prot_Inf_Leak, OT.Prot_Abuse-Func, and OT.Chip_Auth_Proof. Against the identification fraud, the TOE provides identification and authentication services via OT.Identification_and_Authentication. Against the attacks to these services, the TOE protects the TSF data related with identification and authentication services. OT.Prot_Phys-Tamper, OT.Prot_Inf_Leak, and OT.Prot_Abuse-Func provide protection against disclosure of secret authentication key.

T.Unauthorised_Access is countered by OT.Access_Control. It handles the unauthorized access to the user data and services.

T.Unauthorised_Management is countered by OT.Security_Management, which provides mechanisms to manage TSF data, and also provides for the Identification and authentication requirements for the management activities.

P.Identification_and_Authentication is covered by OT.Identification_and_Authentication and OT.Cryptographic_Operations. It covers the support for the chip authentication, terminal

authentication[6], role holder authentication, and PIN verification mechanisms. CMAC property of the OT.Cryptographic_Operations is relevant for this policy.

P.PKI is covered by OE.PKI. Additionally, OT.Identification_and_Authentication covers support for the chip authentication, terminal authentication[7] and role holder authentication mechanisms. These authentication mechanisms include the verification of PKI hierarchy dictated by P.PKI.

P.Access_Control is covered by OT.Access_Control.

P.PreOperational_Security_Management is covered by OT.Security_Management and OE.User_Data_Handling.

P.Operational_Security_Management is covered by OT.Security_Management and OE.User_Data_Handling.

P.Cryptographic_Operations is covered by OT.Cryptographic_Operations.

A.Secure_Application is covered by OE.Secure_Application.

A.Key_and_Certificate_Security is covered by OE.Key_and_Certificate_Security.

A.PIN_Handling is covered by OE.PIN_Handling.

A.Personnel_Security is covered by OE.Personnel_Security.

A.Trusted_Parties is covered by OE.Responsible_Parties.

A.Pre-Operational_Environment is covered by OE.Pre-Operational_Env_Sec.

Table 12 gives the coverage of the threats, assumptions and OSPs by the objectives.

Table 12: Security objectives versus threats, OSPs, and assumptions

	OT.Storage_Integrity	OT.Prot_Phys-Tamper	OT.Prot_Inf_Leak	OT.Prot_Malfunction	OT.Prot_Abuse-Func	OT.Chip_Auth_Proof	OT.Secure_Communication	OT.Identification_and_Authentication	OT.Access_Control	OT.Security_Management	OT.Cryptographic_Operations	OE.PKI	OE.User_Data_Handling	OE.Secure_Application	OE.Key_and_Certificate_Security	OE.PIN_Handling	OE.Personnel_Security	OE.Responsible_Parties	OE.Pre-Operational_Env_Sec
T.Skimming	✓	✓					✓	✓											

6 Provided by PIN authentication for SAM Configuration.
7 Provided by PIN authentication for SAM Configuration.

T.Eavesdropping							✓											
T.Counterfeit		✓	✓		✓	✓		✓										
T.Abuse-Func					✓													
T.Information_Leakage			✓															
T.Phys-Tamper	✓	✓																
T.Malfunction	✓			✓														
T.Session_Hijacking							✓											
T.Man_in_The_Middle							✓											
T.Unauthorised_Access									✓									
T.Unauthorised_Management										✓								
P.Identification_and_Authentication								✓			✓							
P.PKI								✓				✓						
P.Access_Control									✓									
P.PreOperational_Security_Management										✓			✓					
P.Operational_Security_Management										✓			✓					
P.Cryptographic_Operations											✓							
P.Process-TOE								✓										
A.Secure_Application														✓				
A.Key_and_Certificate_Security															✓			
A.PIN_Handling																✓		
A.Personnel_Security																	✓	
A.Trusted_Parties																		✓
A.Pre-Operational_Environment																		✓

6 EXTENDED COMPONENTS

The extended components defined and described for the TOE are:

- Family FAU_SAS (Audit Data Storage),
- Family FCS_RNG (Generation of Random Numbers),
- Family FMT_LIM (Limited capabilities and availability),
- Family FIA_API (Application Proof of Identity),
- Family FPT_EMSEC (TOE Emanation).

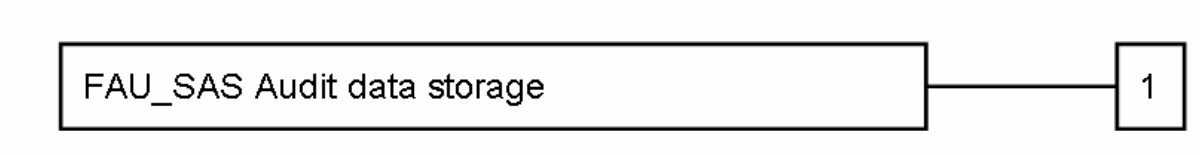
6.1 DEFINITION OF THE FAMILY FAU_SAS (AUDIT DATA STORAGE)

FAU_SAS family of the Class FAU (Security Audit) describes the functional requirements for the storage of audit data. It has a more general approach than FAU_GEN, because it does not necessarily require the data to be generated by the TOE itself and because it does not give specific details of the content of the audit records.

Family behavior

This family defines functional requirements for the storage of audit data.

Component leveling



FAU_SAS.1 Requires the TOE to provide the possibility to store audit data.

Management: FAU_SAS.1

There are no management activities foreseen.

Audit: FAU_SAS.1

There are no actions defined to be auditable.

6.1.1 FAU_SAS.1 AUDIT STORAGE

Hierarchical to: No other components.

Dependencies: No dependencies.

FAU_SAS.1.1 The TSF shall provide [assignment: *list of subjects*] with the capability to store [assignment: *list of audit information*] in the [assignment: *type of persistent memory*].

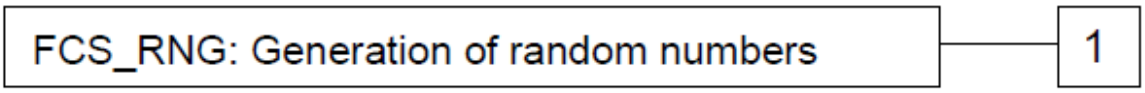
6.2 DEFINITION OF THE FAMILY FCS_RNG (GENERATION OF RANDOM NUMBERS)

This family describes the functional requirements for random number generation used for cryptographic purposes.

Family behavior

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

Component leveling:



FCS_RNG.1: Generation of random numbers requires that the random numbers meet a defined quality metric.

Management: FCS_RNG.1

There are no management activities foreseen.

Audit: FCS_RNG.1

There are no actions defined to be auditable.

6.2.1 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] 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*].

6.3 DEFINITION OF THE FAMILY FMT_LIM (LIMITED CAPABILITIES AND AVAILABILITY)

FMT_LIM of the Class FMT (Security Management) describes the functional requirements for the test features of the TOE. The new functional requirements were defined in the class FMT because this

class addresses the management of functions of the TSF. The examples of the technical mechanism used in the TOE appropriate to address the specific issues of preventing the abuse of functions by limiting the capabilities of the functions and by limiting their availability.

Family behavior

This family defines requirements that limit the capabilities and availability of functions in a combined manner. Note that FDP_ACF restricts the access to functions whereas the component Limited Capability of this family requires the functions themselves to be designed in a specific manner.

Component leveling:

FMT_LIM.1 Limited capabilities requires that the TSF is built to provide only the capabilities (perform action, gather information) necessary for its genuine purpose.

FMT_LIM.2 Limited availability requires that the TSF restrict the use of functions (refer to Limited capabilities (FMT_LIM.1)). This can be achieved, for instance, by removing or by disabling functions in a specific phase of the TOE's life-cycle.

Management: FMT_LIM.1, FMT_LIM.2

There are no management activities foreseen.

Audit: FMT_LIM.1, FMT_LIM.2

There are no actions defined to be auditable.

The TOE Functional Requirement "Limited capabilities (FMT_LIM.1)" is specified as follows.

6.3.1 FMT_LIM.1 LIMITED CAPABILITIES

Hierarchical to: No other components.

Dependencies: FMT_LIM.2 Limited availability.

FMT_LIM.1.1 The TSF shall be designed and implemented in a manner that limits its capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced [assignment: *Limited capability and availability policy*].

6.3.2 FMT_LIM.2 LIMITED AVAILABILITY

Hierarchical to: No other components.

Dependencies: FMT_LIM.1 Limited capabilities.

FMT_LIM.2.1 The TSF shall be designed in a manner that limits its availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced [assignment: *Limited capability and availability policy*].

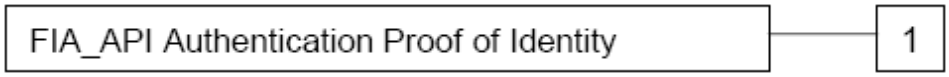
6.4 DEFINITION OF THE FAMILY FIA_API (APPLICATION PROOF OF IDENTITY)

FIA_API of the Class FIA (Identification and Authentication) describes the functional requirements for 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.1 Authentication Proof of Identity:

Management: FIA_API.1

The following actions could be considered for the management functions in FMT: Management of authentication information used to prove the claimed identity.

Audit: FIA_API.1

There are no actions defined to be auditable.

6.4.1 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: *authorized user or role*].

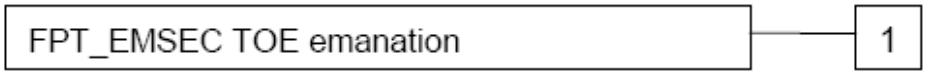
6.5 DEFINITION OF THE FAMILY FPT_EMSEC (TOE EMANATION)

FPT_EMSEC of the Class FPT (Protection of the TSF) is defined here to describe the IT security requirements of the TOE. The TOE shall prevent attacks against TOE and other secret data where the attack is based on external observable physical phenomena of the TOE. This family describes the functional requirements for the limitation of intelligible emanations which are not directly addressed by other functional requirements defined in Common Criteria Part 2.

Family behavior

This family defines requirements to mitigate intelligible emanations.

Component Leveling



FPT_EMSEC.1 TOE Emanation has two constituents:

FPT_EMSEC.1.1 Limit of emissions requires to not emit intelligible emissions enabling access to TSF data or user data.

FPT_EMSEC.1.2 Interface emanation requires to not emit interface emanation enabling access to TSF data or user data.

Management: FPT_EMSEC.1

There are no management activities foreseen.

Audit: FPT.EMSEC.1

There are no actions defined to be auditable.

6.5.1 FPT_EMSEC.1 TOE EMANATION

Hierarchical to: No other components

Dependencies: No dependencies

FPT_EMSEC.1.1 The TOE shall not emit [assignment: *types of emissions*] in excess of [assignment: *specified limits*] enabling access to [assignment: *list of types of TSF data*] and [assignment: *list of types of user data*].

FPT_EMSEC.1.2 The TSF shall ensure [assignment: *type of users*] are unable to use the following interface [assignment*: type of connection*] to gain access to [assignment: *list of types of TSF data*] and [assignment: *list of types of user data*].

7 SECURITY REQUIREMENTS

7.1 OVERVIEW

This part of the ST defines the detailed security requirements that shall be satisfied by the TOE. The statement of TOE security requirements shall define the functional and assurance security requirements that the TOE needs to satisfy in order to meet the security objectives for the TOE. The CC allows several operations to be performed on functional requirements; refinement, selection, assignment, and iteration are defined in Section 8.1 of Common Criteria Part 1 [4]. Each of these operations is used in this ST.

The **refinement** operation is used to add detail to a requirement, and thus further restricts a requirement. Refinements of security requirements are denoted in such a way that added words are in **bold text** and removed are ~~crossed out~~.

The **selection** operation is used to select one or more options provided by the CC instating a requirement. Selections having been made are denoted as underlined text.

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments are denoted by *italicized* text.

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

The **assignment in a selection** operation is used when assignment operation is selected a selection option. Assignments in selections are denoted by *italicized and underlined text*.

7.2 SECURITY FUNCTIONAL REQUIREMENTS

TOE security functional requirements of the composite product are summarized in Table 13.

Table 13: List of SFRs

CLASS FAU		
1.	FAU_SAS.1	Audit Storage
CLASS FCS		
2.	FCS_CKM.1/SM	Cryptographic Key Generation - Secure Messaging Session Keys
3.	FCS_CKM.1/SM_PER-INI	Cryptographic Key Generation - Secure Messaging Keys for Pre-Operational Phase

4.	FCS_CKM.1/RSA_KeyPair	Cryptographic Key Generation - RSA Key Pair Generation
5.	FCS_CKM.1/ECC_KeyPair	Cryptographic Key Generation - ECC Key Pair Generation
6.	FCS_CKM.2/SM	Cryptographic Key Distribution - Secure Messaging Keys
7.	FCS_CKM.2/SM_PER-INI	Cryptographic Key Distribution - Secure Messaging Keys For Pre-Operational Phases
8.	FCS_CKM.4	Cryptographic Key Destruction
9.	FCS_COP.1/SHA	Cryptographic Operation-SHA Calculation
10.	FCS_COP.1/SEC-MSG_AES	Cryptographic Operation - AES Calculation for Secure Messaging
11.	FCS_COP.1/INIT-PERSO-VER_AES	Cryptographic Operation - Initialization/Personalization Verification with AES
12.	FCS_COP.1/ENC-DEC_AES	Cryptographic Operation - Encryption/Decryption AES
13.	FCS_COP.1/ENC-DEC_TDES	Cryptographic Operation - Encryption/Decryption TDES
14.	FCS_COP.1/CMAC_AES	Cryptographic Operation - AES CMAC Computation
15.	FCS_COP.1/CMAC_TDES	Cryptographic Operation - Triple DES CMAC Computation
16.	FCS_COP.1/MAC_TDES	Cryptographic Operation - Triple DES MAC Computation
17.	FCS_COP.1/MAC_AES	Cryptographic Operation - AES MAC Computation
18.	FCS_COP.1/Retail-MAC	Cryptographic Operation - Retail MAC Computation
19.	FCS_COP.1/SIG-GEN_PKCS#1_V1.5	Cryptographic Operation - Signature Generation PKCS#1 v1.5
20.	FCS_COP.1/SIG-GEN_PKCS#1_V2.1	Cryptographic Operation - Signature Generation PKCS#1 v2.1
21.	FCS_COP.1/SIG-GEN_9796	Cryptographic Operation - Signature Generation ISO/IEC 9796-2 Scheme 1

22.	FCS_COP.1/SIG-GEN_ECDSA	Cryptographic Operation - Signature Generation ECDSA
23.	FCS_COP.1/SIG-VER_9796	Cryptographic Operation - Signature Verification ISO/IEC 9796-2 Scheme 1
24.	FCS_COP.1/DEC_PKCS#1_V1.5	Cryptographic Operation - Asymmetric Decryption PKCS#1 v.1.5
25.	FCS_COP.1/DEC_PKCS#1_V2.1	Cryptographic Operation - Asymmetric Decryption PKCS#1 v2.1 OAEP
26.	FCS_COP.1/RSA_RAW	Cryptographic Operation - Asymmetric Encryption/Decryption RAW RSA
27.	FCS_RNG.1	Generation of Random Numbers
CLASS FDP		
28.	FDP_ACC.1/Data	Subset Access Control-Data Access
29.	FDP_ACC.1/FUN	Subset Access Control - Function Access
30.	FDP_ACF.1/Data	Security Attribute Based Access Control-Data Access
31.	FDP_ACF.1/FUN	Security Attribute Based Access Control - Function Access
32.	FDP_UCT.1	Basic Data Exchange Confidentiality
33.	FDP_UIT.1	Data Exchange Integrity
34.	FDP_IFC.1	Subset Information Flow Control
35.	FDP_ITT.1	Basic Internal (User Data) Transfer Protection
36.	FDP_SDI.2	Stored data integrity monitoring and action
CLASS FIA		
37.	FIA_AFL.1/PIN	Authentication Failure Handling - PIN Verification
38.	FIA_AFL.1/ACT	Authentication Failure Handling - Activation
39.	FIA_AFL.1/INI	Authentication Failure Handling - Initialization

40.	FIA_AFL.1/PER	Authentication Failure Handling - Personalization
41.	FIA_API.1	Authentication Proof of Identity
42.	FIA_UAU.1	Timing of Authentication
43.	FIA_UAU.4	Single Use Authentication Mechanisms
44.	FIA_UAU.5	Multiple Authentication Mechanisms
45.	FIA_UAU.6	Re-Authenticating
46.	FIA_UID.1	Timing of Identification
CLASS FMT		
47.	FMT_LIM.1	FMT_LIM.1 Limited Capabilities
48.	FMT_LIM.2	FMT_LIM.2 Limited Availability
49.	FMT_SMF.1	Specification of Management Functions
50.	FMT_SMR.1	Security Roles
51.	FMT_MOF.1	Management of Security Functions Behavior
52.	FMT_MSA.1	Management of Security Attributes
53.	FMT_MTD.1/ INI_PER_AUTH_DATA	Management of TSF Data - Initialization and Personalization Authentication Data Write
54.	FMT_MTD.1/ INI_PER_AUTH_DATA_Change	Management of TSF data - Initialization and Personalization Authentication Data Change
55.	FMT_MTD.1/ Keys_and_AC_Rules_Write_and _Change	Management of TSF Data-Keys and Access Control Rules Write And Change
56.	FMT_MTD.1/PuK_Keys_Use	Management of TSF data-Public Key Usage
57.	FMT_MTD.1/PrK_Use	Management of TSF data Private Key Usage

58.	FMT_MTD.1/PIN_Management	Management of TSF data - PIN Management
CLASS FPT		
59.	FPT_EMSEC.1	TOE Emanation
60.	FPT_FLS.1	Failure with Preservation of Secure State
61.	FPT_ITT.1	Basic Internal TSF Data Transfer Protection
62.	FPT_PHP.3	Resistance to Physical Attack
63.	FPT_TST.1	TOE Testing
CLASS FRU		
64.	FRU_FLT.2	Limited Fault Tolerance

7.2.1 CLASS FAU: SECURITY AUDIT

FAU_SAS.1 Audit storage

Hierarchical to: No other components.

Dependencies: No dependencies.

FAU_SAS.1.1 The TSF shall provide the *IC Manufacturer*[8] with the capability to store *IC Identification Data* [9] in the *not changeable configuration EEPROM area* [10].

8 [assignment: list of subjects]
9 [assignment: list of audit information]
10 [assignment: type of persistent memory].

7.2.2 CLASS FCS: CRYPTOGRAPHIC SUPPORT

The following crypto is implemented and evaluated in the TOE:

- SHA Operation
- TDES Operations
- AES Operations
- CMAC Operations
- MAC Operations
- Retail-MAC Operation
- Signature Generation PKCS#1 v1.5
- Signature Generation PKCS#1 v2.1
- Signature Generation ISO/IEC 9796-2 Scheme 1
- Signature Generation ECDSA
- Signature Verification ISO/IEC 9796-2 Scheme 1
- Asymmetric Decryption PKCS#1 v1.5
- Asymmetric Decryption PKCS#1 v2.1
- Asymmetric Encryption/Decryption RAW RSA
- RSA Key Pair Generation
- ECC Key Pair Generation
- Random Number Generation

Preface regarding Security Level related to Cryptography

The strength of the cryptographic algorithms was not rated in the course of the Product Certification. To fend off attackers with high attack potential, appropriate cryptographic algorithms with adequate key lengths must be used (references can be found in national and international documents and standards). According to these standards and documents, RSA-1024 is not recommended. In addition, since cryptographic functionalities with a security level lower than 100 bits can no longer be regarded as secure without considering the application context, TDES with 112 bit keys is not recommended either. The hardware platform, however, does include countermeasures against side channel attacks on TDES. Therefore, for these functionalities it shall be checked whether the related cryptographic operations are appropriate for the intended system.

FCS_CKM.1/SM Cryptographic Key Generation - Secure Messaging Session Keys

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] fulfilled by both FCS_CKM.2, FCS_COP.1/SEC-MSG_AES and FCS_COP.1/CMAC_AES
[FCS_CKM.4 Cryptographic Key Destruction] fulfilled by FCS_CKM.4

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *Diffie-Hellman-Protocol Key Agreement*

Method[11] and specified cryptographic key sizes *32 bytes*[12] that meet the following *NIST 800-56A*[13].

Application Note 1: Generated keys by this SFR are used by both the TOE and the terminal. These keys are distributed to the terminal by FCS_CKM.2 and used by the FCS_COP.1/CMAC_AES and FCS_COP.1/SEC-MSG_AES.

FCS_CKM.1/SM_PER-INI Cryptographic Key Generation - Secure Messaging Keys for Pre-Operational Phase

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] fulfilled by FCS.CKM.2/SM_PER-INI, FCS_COP.1/SEC-MSG_AES, FCS_COP.1/CMAC_AES

[FCS_CKM.4 Cryptographic Key Destruction] fulfilled by FCS_CKM.4

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *Pre-Operational Secure Messaging Key Generation Algorithm for AKiS v2.5.2N*[14] and specified cryptographic key sizes *32 bytes*[15] that meet the following: *none*[16].

Application Note 2: This functionality is valid for pre-operational phases.

FCS_CKM.1/RSA_KeyPair Cryptographic Key Generation - RSA Key Pair Generation

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic Key Distribution, or FCS_COP.1 Cryptographic Operation] fulfilled by FCS_COP.1/SIG-GEN_PKCS#1_V1.5, FCS_COP.1/SIG-GEN_PKCS#1_V2.1, FCS_COP.1/SIG-GEN_9796, FCS_COP.1/DEC_PKCS#1_V1.5, FCS_COP.1/DEC_PKCS#1_V2.1, FCS_COP.1/RSA_RAW

[FCS_CKM.4 Cryptographic Key Destruction] is fulfilled by FCS_CKM.4

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

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *RSA-CRT*[17] and specified cryptographic key sizes *1024-to-2816 bits*[18] that meet the following: *"Regulierungsbehörde für Telekommunikation und Post: Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung (Übersicht über geeignete Algorithmen), German "Bundesanzeiger Nr. 59", p. 4695-4696, March 30th, 2005"*[19].

FCS_CKM.1/ECC_KeyPair Cryptographic Key Generation - ECC Key Pair Generation

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic Key Distribution, or FCS_COP.1 Cryptographic Operation] is fulfilled by FCS_COP.1/SIG-GEN_ECDSA

[FCS_CKM.4 Cryptographic Key Destruction] is fulfilled by FCS_CKM.4

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *ECDSA (ECC over GF(p))*[20] and specified cryptographic key sizes *128 to 640 bits*[21] that meet the following: *ISO 15946-1 [26] and "Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen: Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung (Übersicht über geeignete Algorithmen), German 'Bundesanzeiger', BAnz AT 30.01.2015 B3" [28]* [22].

FCS_CKM.2/SM Cryptographic Key Distribution - Secure Messaging Keys

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] fulfilled by FCS_CKM.1/SM

[FCS_CKM.4 Cryptographic Key Destruction] is fulfilled by FCS_CKM.4

17 [assignment: cryptographic key generation algorithm]
18 [assignment: cryptographic key sizes]
19 [assignment: list of standards]
20 [assignment: cryptographic key generation algorithm]
21 [assignment: cryptographic key sizes]
22 [assignment: list of standards]

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method *Device Authentication-Secure Messaging*[23] that meets the following: *TCKK Projesinde Kullanılan Kriptografik Algoritmalar Tanım Dokümanı, 4 Nisan 2012, v1.3, TÜBİTAK BİLGEM UEKAE Kriptoloji Birimi*[24].

FCS_CKM.2/SM_PER-INI Cryptographic Key Distribution - Secure Messaging Keys for Pre-Operational Phases

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] fulfilled by FCS_CKM.1/SM_PER-INI.

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method: *AKiS v2.5.2N SM_PER-INI key distribution method*[25] that meets the following: *TCKK Projesinde Kullanılan Kriptografik Algoritmalar Tanım Dokümanı, 4 Nisan 2012, v1.3, TÜBİTAK BİLGEM UEKAE Kriptoloji Birimi*[26].

FCS_CKM.4 Cryptographic Key Destruction

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] fulfilled by FCS_CKM.1/SM and FCS_CKM.1/SM_PER-INI

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *AKiS v2.5.2N Key Destruction Method*[27] that meets the following: *none*[28].

23 [assignment: cryptographic key distribution method]
24 [assignment: list of standards]
25 [assignment: cryptographic key distribution method]
26 [assignment: list of standards]
27 [assignment: cryptographic key destruction method]

FCS_COP.1/SHA Cryptographic Operation - SHA 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] is not fulfilled but justified.

[FCS_CKM.4 Cryptographic key destruction] is not fulfilled but justified.

FCS_COP.1.1 The TSF shall perform *hash value calculation*[29] in accordance with a specified cryptographic algorithm *SHA-256, SHA-384 and SHA-512*[30] and cryptographic key sizes *none*[31] that meet the following: *FIPS PUB 180-4 [16]*[32].

Application Note 3: TOE also has SHA-1 capability. However, it is out of scope for this certification.

FCS_COP.1/SEC-MSG_AES Cryptographic Operation - AES Calculation for Secure Messaging

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] is fulfilled by FCS_CKM.1/SM and FCS_CKM.1/SM_PER-INI.

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4.

FCS_COP.1.1 The TSF shall perform encryption and decryption[33] in accordance with a specified cryptographic algorithm AES CBC mode[34] and cryptographic key sizes 256 bits[35] that meet the following: *FIPS 197 [17], NIST SP 800-38A [18]*[36].

28 [assignment: list of standards]
29 [assignment: list of cryptographic operations]
30 [assignment: cryptographic algorithm]
31 [assignment: cryptographic key sizes]
32 [assignment: list of standards]
33 [assignment: list of cryptographic operations]
34 [assignment: cryptographic algorithm]
35 [assignment: cryptographic key sizes]
36 [assignment: list of standards]

FCS_COP.1/INIT-PERSO-VER_AES Cryptographic Operation - Initialization/Personalization Verification with AES

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] is not fulfilled but justified.

[FCS_CKM.4 Cryptographic key destruction] is not fulfilled but justified.

FCS_COP.1.1 The TSF shall perform initialization and personalization verification with decryption[37] in accordance with a specified cryptographic algorithm AES CBC mode[38] and cryptographic key sizes 256 bits[39] that meet the following: FIPS PUB 197 [17], NIST SP 800-38A [18] [40].

FCS_COP.1/ENC-DEC_AES Cryptographic Operation - Encryption/Decryption AES

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] is not fulfilled but justified.

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4.

FCS_COP.1.1 The TSF shall perform encryption and decryption[41] in accordance with a specified cryptographic algorithm AES ECB and CBC mode[42] and cryptographic key sizes 128, 192, and 256 bits[43] that meet the following: *FIPS 197 [17], NIST SP 800-38A [18]*[44].

FCS_COP.1/ENC-DEC_TDES Cryptographic Operation - Encryption/Decryption TDES

Hierarchical to: No other components.

37 [assignment: list of cryptographic operations]
38 [assignment: cryptographic algorithm]
39 [assignment: cryptographic key sizes]
40 [assignment: list of standards]
41 [assignment: list of cryptographic operations]
42 [assignment: cryptographic algorithm]
43 [assignment: cryptographic key sizes]
44 [assignment: list of standards]

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] is not fulfilled but justified.

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4.

FCS_COP.1.1 The TSF shall perform encryption and decryption[45] in accordance with a specified cryptographic algorithm Triple DES ECB and CBC mode[46] and cryptographic key sizes 112 bits[47] that meet the following: *FIPS PUB 46-3 [19], keying option 2, NIST SP 800-38A [18]*[48].

FCS_COP.1/CMAC_AES Cryptographic Operation - AES CMAC Computation

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] is fulfilled by FCS_CKM.1/SM and FCS_CKM.1/SM_PER-INI

[FCS_CKM.4 Cryptographic key destruction] fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *message authentication code*[49] in accordance with a specified cryptographic algorithm *AES-CMAC*[50] and cryptographic key sizes 16, 24, and *32 bytes*[51] that meet the following:

- *FIPS Publication 197, Advanced Encryption Standard (AES) [17],*
- *NIST Special Publication 800-38B [20]* [52]

Application Note 4: The TOE has an interface for CMAC operation with AES-128, AES-192 and AES-256 as well as an automatic use for secure messaging with AES-256.

45 [assignment: list of cryptographic operations]
46 [assignment: cryptographic algorithm]
47 [assignment: cryptographic key sizes]
48 [assignment: list of standards]
49 [assignment: list of cryptographic operations]
50 [assignment: cryptographic algorithm]
51 [assignment: cryptographic key sizes]
52 [assignment: list of standards]

FCS_COP.1/CMAC_TDES Cryptographic Operation - Triple DES CMAC Computation

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] is not fulfilled but justified.

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *message authentication code*[53] in accordance with a specified cryptographic algorithm *Triple DES CMAC*[54] and cryptographic key sizes *112 bits*[55] that meet the following:

- *FIPS PUB 46-3 [19], keying option 2,*
- *NIST Special Publication 800-38B [20]* [56]

FCS_COP.1/MAC_TDES Cryptographic Operation - Triple DES MAC Computation

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] is not fulfilled but justified.

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *message authentication code*[57] in accordance with a specified cryptographic algorithm *Triple DES CBC-MAC*[58] and cryptographic key sizes *112 bits*[59] that meet the following:

- *FIPS PUB 46-3 [19], keying option 2,*
- *ISO 9797-1, MAC algorithm 1 [25]* [60]

53 [assignment: list of cryptographic operations]
54 [assignment: cryptographic algorithm]
55 [assignment: cryptographic key sizes]
56 [assignment: list of standards]
57 [assignment: list of cryptographic operations]
58 [assignment: cryptographic algorithm]
59 [assignment: cryptographic key sizes]
60 [assignment: list of standards]

FCS_COP.1/MAC_AES Cryptographic Operation - AES MAC Computation

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] is not fulfilled but justified.

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *message authentication code*[61] in accordance with a specified cryptographic algorithm *AES CBC-MAC*[62] and cryptographic key sizes *16, 24, and 32 bytes*[63] that meet the following:

- *FIPS Publication 197, Advanced Encryption Standard (AES) [17],*
- *ISO 9797-1, MAC algorithm 1 [25]*[64]

FCS_COP.1/Retail-MAC Cryptographic Operation - Retail MAC Computation

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] is not fulfilled but justified.

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *message authentication code*[65] in accordance with a specified cryptographic algorithm *Retail MAC*[66] and cryptographic key sizes *112 bits*[67] that meet the following:

- *FIPS PUB 46-3 [19], keying option 2,*
- *ISO 9797-1, MAC algorithm 3, Padding Method 2 [25]*[68]

61 [assignment: list of cryptographic operations]
62 [assignment: cryptographic algorithm]
63 [assignment: cryptographic key sizes]
64 [assignment: list of standards]
65 [assignment: list of cryptographic operations]
66 [assignment: cryptographic algorithm]
67 [assignment: cryptographic key sizes]
68 [assignment: list of standards]

FCS_COP.1/SIG-GEN_PKCS#1_V1.5 Cryptographic Operation - Signature Generation PKCS#1 v1.5

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] is fulfilled by FCS_CKM.1/RSA_KeyPair

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *digital signature generation*[69] in accordance with specified cryptographic algorithm *RSASSA*[70] and cryptographic key sizes *1024-to-2816 bits*[71] that meet the following: *PKCS#1 v1.5, RFC 2313, March 1998* [72].

FCS_COP.1/SIG-GEN_PKCS#1_V2.1 Cryptographic Operation - Signature Generation PKCS#1 v2.1

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] is fulfilled by FCS_CKM.1/RSA_KeyPair

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *digital signature generation*[73] in accordance with a specified cryptographic algorithm *RSASSA-PSS*[74] and cryptographic key sizes *1024-to-2816 bits*[75] that meet the following: *PKCS#1 v2.1, RFC 3447, February 2003* [76].

69 [assignment: list of cryptographic operations]
70 [assignment: cryptographic algorithm]
71 [assignment: cryptographic key sizes]
72 [assignment: list of standards]
73 [assignment: list of cryptographic operations]
74 [assignment: cryptographic algorithm]
75 [assignment: cryptographic key sizes]
76 [assignment: list of standards]

FCS_COP.1/SIG-GEN_9796 Cryptographic Operation - Signature Generation ISO/IEC 9796-2 Scheme 1

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] is fulfilled by FCS_CKM.1/RSA_KeyPair

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *digital signature generation*[77] in accordance with a specified cryptographic algorithm *RSA and SHA-256, SHA-384 and SHA-512*[78] and cryptographic key sizes *1024-to-2816 bits*[79] that meet the following: *ISO/IEC 9796-2 Scheme 1, 2010* [80].

FCS_COP.1/SIG-GEN_ECDSA Cryptographic Operation - Signature Generation 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] is fulfilled by FCS_CKM.1/ECC_KeyPair

[FCS_CKM.4 Cryptographic key destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *digital signature generation*[81] in accordance with the specified cryptographic algorithm *ECDSA / ECC over GF(p)*[82] and cryptographic key sizes *128 to 640 bits*[83] that meet the following: *ISO 15946-2 [27]*[84].

Application Note 5: Due to BSI regulations the certification of the platform covers standard NIST and Brainpool elliptic curves. As a consequence, the certification of the TOE covers only these elliptic curves as well.

77 [assignment: list of cryptographic operations]
78 [assignment: cryptographic algorithm]
79 [assignment: cryptographic key sizes]
80 [assignment: list of standards]
81 [assignment: list of cryptographic operations]
82 [assignment: cryptographic algorithm]
83 [assignment: cryptographic key sizes]
84 [assignment: list of standards]

FCS_COP.1/SIG-VER_9796 Cryptographic Operation - Signature Verification ISO/IEC 9796-2 Scheme 1

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] is not fulfilled but justified

[FCS_CKM.4 Cryptographic Key Destruction] is not fulfilled but justified

FCS_COP.1.1 The TSF shall perform *digital signature verification*[85] in accordance with a specified cryptographic algorithm *RSA and SHA-256, SHA-384 and SHA-512*[86] and cryptographic key sizes *1024-to-2816 bits*[87] that meet the following: *ISO/IEC 9796-2 Scheme 1, December 2010*[88].

FCS_COP.1/DEC_PKCS#1_v1.5 Cryptographic Operation - Asymmetric Decryption PKCS#1 v.1.5

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] is fulfilled by FCS_CKM.1/RSA_KeyPair

[FCS_CKM.4 Cryptographic Key Destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *asymmetric decryption*[89] in accordance with specified cryptographic algorithm *RSAES*[90] and cryptographic key sizes *1024-to-2816 bits*[91] that meet the following: *PKCS #1 v1.5, RFC 2313, March 1998.*

85 [assignment: list of cryptographic operations]
86 [assignment: cryptographic algorithm]
87 [assignment: cryptographic key sizes]
88 [assignment: list of standards]
89 [assignment: list of cryptographic operations]
90 [assignment: cryptographic algorithm]
91 [assignment: cryptographic key sizes]

FCS_COP.1/DEC_PKCS#1_v2.1 Cryptographic Operation - Asymmetric Decryption PKCS#1 v2.1 OAEP

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] is fulfilled by FCS_CKM.1/RSA_KeyPair

[FCS_CKM.4 Cryptographic Key Destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *asymmetric decryption*[92] in accordance with a specified cryptographic algorithm *RSAES-OAEP*[93] and cryptographic key sizes *1024-to-2816 bits*[94] that meet the following: *PKCS #1 v2.1,RFC 3447, February 2003*[95]

FCS_COP.1/RSA_RAW Cryptographic Operation - Asymmetric Encryption/Decryption RAW 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] is fulfilled by FCS_CKM.1/RSA_KeyPair

[FCS_CKM.4 Cryptographic Key Destruction] is fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *asymmetric encryption/decryption*[96] in accordance with a specified cryptographic algorithm *Rivest-Shamir-Adleman (RSA-Raw)*[97] and cryptographic key sizes *1024-to-2816 bits*[98] that meet the following: *RSA Cryptography Standard*[99]

Application Note 6: TOE has no interface for these operations. They are performed automatically when chip and terminal authentication operations start.

92 [assignment: list of cryptographic operations]
93 [assignment: cryptographic algorithm]
94 [assignment: cryptographic key sizes]
95 [assignment: list of standards]
96 [assignment: list of cryptographic operations]
97 [assignment: cryptographic algorithm]
98 [assignment: cryptographic key sizes]
99 [assignment: list of standards]

FCS_RNG.1 Random Number Generation

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1 The TSF shall provide a mechanism to generate random numbers that meet:

DRG.4.1 The internal state of the RNG shall use PTRNG or class PTG.2 (as defined in [24]) as random source.
DRG.4.2 The RNG provides forward secrecy (as defined in [24]).
DRG.4.3 The RNG provides backward secrecy even if the current internal state is known (as defined in [24]).
DRG.4.4 The RNG provides enhanced forward secrecy on demand (as defined in [1]).
DRG.4.5 The internal state of the RNG is seeded by an PTRNG or class PTG.2 (as defined in [24]).

FCS_RNG.1.2 The TSF shall provide random numbers that meet:

DRG.4.6 The RNG generates output for which 2^{48} strings of bit length 128 are mutually different with probability at least $1 - 2^{-24}$.
DRG.4.7 Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A (as defined in [24]).

7.2.3 CLASS FDP: USER DATA PROTECTION

FDP_ACC.1/Data Subset Access Control - Data

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control fulfilled by FDP_ACF.1/Data

FDP_ACC.1.1: The TSF shall enforce the *Application Access Control SFP*[100] on

subjects:

- *initialization agent,*
- *personalization agent,*

100 [assignment: access control SFP]

- *terminal,*
- *application defined and allowed role,*

objects:

- *User data stored*

operations:

- *write, create, read, delete.*[101]

FDP_ACC.1/FUN Subset Access Control - Function

Hierarchical to: No other components.

Dependencies: [FDP_ACF.1 Security Attribute Based Access Control]is fulfilled by FDP_ACF.1/FUN

FDP_ACC.1.1: The TSF shall enforce the *Application Access Control SFP*[102] on

subjects:

- *activation agent,*
- *initialization agent,*
- *personalization agent,*
- *application defined and allowed role, and*

objects and operations as

- *defined commands for activation, initialization, personalization, operation and death sub-phases in [13]* [103].

FDP_ACF.1/Data Security Attributes Based Access Control - Data

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset Access Control] is fulfilled by FDP_ACC.1/Data

[FMT_MSA.3 Static Attribute Initialization] is not fulfilled but justified

101 [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]
102 [assignment: access control SFP]
103 [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]

FDP_ACF.1.1 The TSF shall enforce the A*pplication Access Control SFP*[104] to objects based on the following:

subjects:

- *initialization agent,*
- *personalization agent,*
- *terminal,*
- *application defined and allowed role,*

subject attribute:

- *authorization level of subjects,*

object:

- *user data Stored in TOE,*

object attribute:

- *data access control rules.*[105]

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

- *Application defined and allowed roles have read, write, change access according to rules determined by application developer.*
- *Successfully authenticated terminal*[106] *have read, write, and change access according to rules determined by application developer.*[107]

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *authenticated initialization and personalization agents are authorized to access all application data in pre-operational phase.*[108]

104 [assignment: access control SFP]
105 [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]
106 It means PIN authenticated terminal for SAM configuration.
107 [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]
108 [assignment: rules, based on security attributes that explicitly authorise access of subjects to objects]

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *Nobody shall be allowed to have write, create, read, and delete access user data in death phase*[109].

FDP_ACF.1/FUN Security Attributes Based Access Control - Function

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset Access Control] is fulfilled by FDP_ACC.1/FUN

[FMT_MSA.3 Static Attribute Initialization] is not fulfilled but justified

FDP_ACF.1.1 The TSF shall enforce the A*pplication Access Control SFP*[110] to objects based on the following:

subjects:

- *activation agent,*
- *initialization agent,*
- *personalization agent,*
- *Application defined and allowed roles,*

objects and their attributes as referred to in[111]

- *defined command function for activation sub-phase in document [13]*
- *defined command function for initialization sub-phase in document [13]*
- *defined command function for personalization sub-phase in document [13]*
- *defined command function for operation phase in document [13]*
- *defined command function for death phase in document [13]* [112].

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

109 [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]
110 [assignment: access control SFP]
111 [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]
112 [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]

- *Only activation agent access defined command function for activation sub-phase in document [13]*
- *Only initialization agent access defined command function for initialization sub-phase in document [13]*
- *Only personalization agent defined command function for personalization sub-phase in document [13]*
- *Only application defined and allowed roles access defined command function for operation phase in document [13]* [113].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *Any user is allowed to access defined command functions of Death Phase in stated in the document [13]* [114].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *none*[115].

FDP_UCT.1 Basic Data Exchange Confidentiality

Hierarchical to: No other components.

Dependencies: [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] not fulfilled but justified

[FDP_ACC.1 Subset Access Control, or FDP_IFC.1 Subset information flow control]

Is fulfilled by FDP_ACC.1

FDP_UCT.1.1 The TSF shall enforce the *Application Access Control SFP*[116] to transmit, receive[117] user data in a manner protected from unauthorized disclosure.

Application Note 7: This SFR is valid for the communication between TOE and Terminal.

113 [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]
114 [assignment: rules, based on security attributes that explicitly authorise access of subjects to objects]
115 [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]
116 [assignment: access control SFP(s) and/or information flow control SFP(s)]
117 [selection: transmit, receive]

FDP_UIT.1 Data Exchange Integrity

Hierarchical to: No other components.

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

[FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] not fulfilled but justified

FDP_UIT.1.1 The TSF shall enforce the *Application Access Control SFP*[118] to transmit, receive[119] user data in a manner protected from modification, deletion, insertion, replay[120] errors.

FDP_UIT.1.2 The TSF shall be able to determine on receipt of user data, whether modification, deletion, insertion, replay[121] has occurred.

Application Note 8: This SFR is valid for the communication between TOE and terminal.

FDP_IFC.1 Subset Information Flow Control

Hierarchical to: No other components.

Dependencies: [FDP_IFF.1 Simple security attributes not fulfilled but justified]

FDP_IFC.1.1 The TSF shall enforce the *Smartcard Data Processing Policy*[122] on all confidential data when they are processed between the different parts of the TOE[123].

Refinement: Data Processing Policy: User data and TSF data shall not be accessible from the TOE except when the Security IC embedded software decides to communicate the user data via an external interface. The protection shall be applied to confidential data only but without the distinction of attributes controlled by the security IC embedded software.

118 [assignment: access control SFP(s) and/or information flow control SFP(s)]
119 [selection: transmit, receive]
120 [selection: modification, deletion, insertion, replay]
121 [selection: modification, deletion, insertion, replay]
122 [assignment: information flow control SFP]
123 [assignment: list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP]

FDP_ITT.1 Basic Internal Transfer Protection

Hierarchical to: No other components.

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

FDP_ITT.1.1 The TSF shall enforce the *Platform Data Processing Policy*[124] to prevent the *disclosure*[125] of user data when it is transmitted between physically-separated parts of the TOE.

Refinement: The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor) are seen as physically-separated parts of the TOE.

FDP_SDI.2 Stored data integrity monitoring and action

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring

Dependencies: No dependencies.

FDP_SDI.2.1 The TSF shall monitor user data stored in containers controlled by the TSF for *data integrity and one- and/or more-bit-errors* on all objects, based on the following attributes: *corresponding EDC value for integrity critical user data*

- files
- file headers
- SKK and DBT tables
- special registers

FDP_SDI.2.1 Upon detection of a data integrity error, the TSF shall *inform the user by an error code.*

124 [assignment: access control SFP(s) and/or information flow control SFP(s)]
125 [selection: disclosure, modification, loss of use]

7.2.4 CLASS FIA: IDENTIFICATION AND AUTHENTICATION

FIA_AFL.1/PIN - Authentication Failure Handling - PIN Verification

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1 The TSF shall detect when *an administrator configurable positive integer within 1 to 255*[126] unsuccessful authentication attempts occur related to *PIN authentication event*[127].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met[128], the TSF shall *block the usage of PIN*[129].

FIA_AFL.1/ACT - Authentication Failure Handling - Activation

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1 The TSF shall detect when *64*[130] unsuccessful authentication attempts occur related to *activation role authentication*[131].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been *met*[132], the TSF shall *put the card into death phase*[133].

126 [selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]
127 [assignment: list of authentication events]
128 [selection: met, surpassed]
129 [assignment: list of actions]
130 [selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]
131 [assignment: list of authentication events]
132 [selection: met, surpassed]
133 [assignment: list of actions]

FIA_AFL.1/INI - Authentication Failure Handling - Initialization

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1 The TSF shall detect when *10*[134] unsuccessful authentication attempts occur related to *initialization agent Authentication*[135].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been *met*[136], the TSF shall *put the card into death phase*[137].

FIA_AFL.1/PER - Authentication Failure Handling - Personalization

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1 The TSF shall detect when *10*[138] unsuccessful authentication attempts occur related to *personalization agent authentication*[139].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met[140], the TSF shall *put the card into death phase*[141].

FIA_API.1 Authentication Proof of Identity

Hierarchical to: No other components

Dependencies: No dependency

FIA_API.1.1 The TSF shall provide a *chip authentication*[142] to prove the identity of the *card itself*[143].

134 [selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]
135 [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]
136 [selection: met, surpassed]
137 [assignment: list of actions].
138 [selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]
139 [assignment: list of authentication events]
140 [selection: met, surpassed]
141 [assignment: list of actions]
142 [assignment: authentication mechanism]
143 [assignment: authorized user or role]

Application Note 9: This SFR is valid for both Chip and SAM configuration.

FIA_UAU.1 Timing of Authentication

Hierarchical to: No other components.

Dependencies: [FIA_UID.1 Timing of identification] is fulfilled by FIA_UID.1.

FIA_UAU.1.1 The TSF shall allow

- *to read chip serial number: at pre-operational, operational and death phases, and*
- *to perform any application allowed actions*[144]

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

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

FIA_UAU.4 Single Use Authentication Mechanisms

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to

- *terminal authentication and*
- *role holder authentication.*[145]

Application Note 10: This SFR is valid for both terminal and role authentication for chip configuration. But, terminal authentication is PIN Authentication for SAM configuration as stated before. PIN Authentication is also valid for Chip Configuration. PIN authentication data might be reused normally. But this situation does not cause a security flaw by means of secure messaging capabilities.

144 [assignment: list of TSF mediated actions]
145 [assignment: identified authentication mechanism(s)]

FIA_UAU.5 Multiple Authentication Mechanisms

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UAU.5.1 The TSF shall provide *following authentication mechanisms to support user authentication:*

- *activation agent authentication,*
- *personalization agent authentication,*
- *initialization agent authentication,*
- *terminal authentication*[146]*,*
- *role authentication,*
- *PIN authentication.*[147]

FIA_UAU.5.2 The TSF shall authenticate any user's claimed identity according to the following policies:

- *The TOE will accept the activation agent as authenticated if he or she passes activation agent authentication.*
- *The TOE will accept the initialization agent as authenticated if he or she passes initialization agent authentication.*
- *The TOE will accept the personalization agent as authenticated if he or she passes personalization agent authentication.*
- *The TOE will accept the terminal as rightful terminal if the terminal passes authentication.*
- *The TOE will accept the application defined and allowed role if he or she passes role or PIN authentication*[148]*.*

FIA_UAU.6 Re-Authenticating

Hierarchical to: No other components.

146 It means PIN authentication for SAM configuration.
147 [assignment: list of multiple authentication mechanisms]
148 [assignment: rules describing how the multiple authentication mechanisms provide authentication]

Dependencies: No dependencies.

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

- *each reset or power-up,*
- *each command sent to the TOE during secure messaging.*[149]

FIA_UID.1 Timing of Identification

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UID.1.1 The TSF shall allow

- *to read chip serial number: at pre-operational, operational and death phases and*
- *to perform any application allowed action*[150]

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

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

7.2.5 CLASS FMT: SECURITY MANAGEMENT

FMT_LIM.1 Limited Capabilities

Hierarchical to: No other components.

Dependencies: [FMT_LIM.2 Limited Availability] is fulfilled by FMT_LIM.2

FMT_LIM.1.1 The TSF shall be designed and implemented in a manner that limits its capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced: *Deploying test features after TOE delivery do not allow*

- *user data and TSF data to be manipulated and disclosed,*
- *Embedded Operating System to be reconstructed and*

149 [assignment: list of conditions under which re-authentication is required]
150 [assignment: list of TSF-mediated actions]

- *substantial information about construction of TSF to be gathered which may enable other attacks.*[151]

FMT_LIM.2 Limited Availability

Hierarchical to: No other components.

Dependencies: FMT_LIM.1 Limited capabilities fulfilled by FMT_LIM.1.

FMT_LIM.2.1 The TSF shall be designed in a manner that limits its availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced: *Deploying test features after TOE delivery do not allow*

- *user data and TSF Data to be manipulated and disclosed,*
- *Embedded Operating System to be reconstructed,*
- *substantial information about construction of TSF to be gathered which may enable other attacks.*[152]

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:

- *activation,*
- *initialization,*
- *personalization,*
- *any management function defined by application developer.*[153]

151 [assignment: Limited capability and availability policy]
152 [assignment: Limited capability and availability policy]
153 [assignment: list of management functions to be provided by the TSF]

FMT_SMR.1 Security Roles

Hierarchical to: No other components.

Dependencies: [FIA_UID.1/ Timing of identification] is fulfilled by FIA_UID.1.

FMT_SMR.1.1 The TSF shall maintain the roles

- *activation agent,*
- *initialization agent,*
- *personalization agent,*
- *any management role defined by application developer.*[154]

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

Application Note 11: The term "role" in this SFR is used as a general term as defined in CC Part 2, and it is not about the authenticated role holder.

FMT_MOF.1 Management of Security Functions Behavior

Hierarchical to: No other components.

Dependencies: [FMT_SMR.1 Security Roles] is fulfilled by FMT_SMR.1

[FMT_SMF.1 Specification of Management Functions] is fulfilled by FMT_SMF.1

FMT_MOF.1.1 The TSF shall restrict the ability to disable and enable[155] the functions

- *External interface command for operational mode listed in [14]* [156] to *application defined roles*[157].

Application Note 12: Applicable only for operational phase. Not applicable for activation, initialization and personalization.

154 [assignment: the authorised identified roles]
155 [selection: determine the behaviour of, disable, enable, modify the behaviour of]
156 [assignment: list of functions]
157 [assignment: the authorised identified roles]

FMT_MSA.1 Management of Security Attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] is fulfilled by FDP_ACC.1/Data and FDP_ACC.1/FUN

[FMT_SMR.1 Security Roles] is fulfilled by FMT_SMR.1

[FMT_SMF.1 Specification of Management Functions] is fulfilled by FMT_SMF.1

FMT_MSA.1.1 The TSF shall enforce the *Application Access Control Policy*[158] to restrict the ability to query, modify, delete[159] the security attributes *access control rules of keys, PINs, user data*[160] to *initialization agent, personalization agents and application defined roles*[161].

FMT_MTD.1/INI_PER_AUTH_DATA Management of TSF Data - Initialization and Personalization Authentication Data Write

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles fulfilled by FMT_SMR.1

FMT_SMF.1 Specification of Management Functions fulfilled by FMT_SMF.1

FMT_MTD.1.1 The TSF shall restrict the ability to write[162] the *authentication reference data for Initialization and personalization agents*[163] to *activation agent*[164].

FMT_MTD.1/INI_PER_AUTH_DATA_Change Management of TSF Data - Initialization and Personalization Authentication Data Change

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles fulfilled by FMT_SMR.1

FMT_SMF.1 Specification of Management Functions fulfilled by FMT_SMF.1

158 [assignment: access control SFP(s), information flow control SFP(s)]
159 [selection: change default, query, modify, delete, [assignment: other operations]]
160 [assignment: list of security attributes]
161 [assignment: the authorised identified roles]
162 [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
163 [assignment: list of TSF data]
164 [assignment: the authorised identified roles]

FMT_MTD.1.1 The TSF shall restrict the ability to change[165] *the authentication reference data for Initialization and personalization agents*[166] to *Initialization and personalization agents*[167].

FMT_MTD.1/Keys_and_AC_Rules_Write_and_Change Management of TSF Data - Keys and Access Control Rules Write and Change

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles fulfilled by FMT_SMR.1

FMT_SMF.1 Specification of Management Functions fulfilled by FMT_SMF.1

FMT_MTD.1.1 The TSF shall restrict the ability to write and change[168] the *root Certificate Authority public key, chip authentication PuK and PrK and access control Rules*[169] to *initialization agent, personalization agent any application defined and allowed role*[170].

FMT_MTD.1/PuK_Keys_Use Management of TSF Data - Usage Public Key Usage

Hierarchical to: No other components.

Dependencies: [FMT_SMR.1 Security Roles] fulfilled by FMT_SMR.1

[FMT_SMF.1 Specification of Management Functions] fulfilled by FMT_SMF.1

FMT_MTD.1.1 The TSF shall restrict the ability to use[171] the *Root CA PuK and chip authentication PuK*[172] to *application defined and allowed roles*[173].

FMT_MTD.1/PrK_Use Management of TSF Data - Private Key Usage

Hierarchical to: No other components.

Dependencies: [FMT_SMR.1 Security Roles] fulfilled by FMT_SMR.1

[FMT_SMF.1 Specification of Management Functions] is fulfilled by FMT_SMF.1

165 [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
166 [assignment: list of TSF data]
167 [assignment: the authorised identified roles]
168 [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
169 [assignment: list of TSF data]
170 [assignment: the authorised identified roles]
171 [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
172 [assignment: list of TSF data]
173 [assignment: the authorised identified roles]

FMT_MTD.1.1 The TSF shall restrict the ability to use[174] the *chip authentication PrK*[175] to *application defined and allowed roles*[176].

FMT_MTD.1/PIN_Management Management of TSF Data - PIN Management

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles fulfilled by FMT_SMR.1

FMT_SMF.1 Specification of Management Functions fulfilled by FMT_SMF.1

FMT_MTD.1.1 The TSF shall restrict the ability to write, change, and unblock[177] the *PIN objects*[178] to *initialization agent, personalization agents, any application defined and allowed roles*[179].

7.2.6 CLASS FPT: PROTECTION OF THE TSF

FPT_EMSEC.1 TOE Emanation

Hierarchical to: No other components

Dependencies: No dependencies

FPT_EMSEC.1.1 The TOE shall not emit*, timing variations during command execution*[180] in excess of *non-useful information*[181] enabling access to *Initialization and Personalization Keys, PINs used by the application*[182], and *none*.[183]

FPT_EMSEC.1.2 The TSF shall ensure *any users*[184] are unable to use the following interface *contact interface and physical contacts*[185] to gain access to *none*.[186]

174 [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
175 [assignment: list of TSF data]
176 [assignment: the authorised identified roles]
177 [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
178 [assignment: list of TSF data]
179 [assignment: the authorised identified roles]
180 [assignment: types of emissions]
181 [assignment: specified limits]
182 [assignment: list of types of TSF data]
183 [assignment list of types of user data].
184 [assignment: type of users]
185 [assignment: type of connection]
186 [assignment: list of type of user data].

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: *(i) 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 and (ii) DFA attacks on AES, DES, 3DES, RSA, ECC*[187].

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

Application Note 13: This refinement should be understood with the following implementation details in mind: The TOE contains both hardware sensors (implemented in the chip card hardware) and software sensors (implemented in the Crypto Library software). The software sensors detect DFA attacks in cryptographic computations and this detection leads to a secure state (no computation results are output and an exception is thrown) in case such an attack occurs. The Embedded Operating System handles this exception by entering an infinite loop, thus further ensures a secure state by forcing reset**.**

FPT_ITT.1 Basic Internal TSF Data Transfer Protection

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_ITT.1.1 The TSF shall protect TSF data from disclosure[188] when it is transmitted between separate parts of the TOE.

Refinement: The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor) are seen as separated parts of the TOE.

187 [assignment: list of types of failures in the TSF]
188 [selection: disclosure, modification]

FPT_PHP.3 Resistance to Physical Attack

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_PHP.3.1 The TSF shall resist *physical manipulation and physical probing*[189] to the *TSF*[190] by responding automatically such that the SFRs are always enforced.

Refinement: The TSF will implement appropriate mechanisms to continuously counter physical manipulation and physical probing. Due to the nature of these attacks (especially manipulation) the TSF can by no means detect attacks on all of its elements. Therefore, permanent protection against these attacks is required ensuring that security functional requirements are enforced. Hence, "automatic response" means here (i) assuming that there might be an attack at any time and (ii) countermeasures are provided at any time.

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 normal operation[191] to demonstrate the **integrity of TSF data except Embedded Operating System code** and correct operation of the TSF[192].

FPT_TST.1.2 The TSF shall ~~provide authenticated users with the capability to~~ verify the integrity of TSF Data[193].

FPT_TST.1.3 The TSF shall ~~provide authenticated users with the capability to~~ verify the integrity of TSF[194].

189 [assignment: physical tampering scenarios]
190 [assignment: list of TSF devices/elements]
191 [selection: during initial start-up, periodically during normal operation, at the request of the authorized user, at the conditions [assignment: conditions under which self test should occur]]
192 [selection: [assignment: parts of TSF], the TSF]
193 [selection: [assignment: parts of TSF data], TSF data]
194 [selection: [assignment: parts of TSF], TSF]

7.2.7 CLASS FRU: RESOURCE UTILISATION

FRU_FLT.2 Limited Fault Tolerance

Hierarchical to: FRU_FLT.1Degraded fault tolerance

Dependencies: [FPT_FLS.1 Failure with Preservation of Secure State] is fulfilled by FPT_FLS.1

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)* [195].

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

Application Note 14: Environmental conditions include but are not limited to power supply, clock, and other external signals (e.g. reset signal) necessary for the TOE operation.

7.3 SECURITY ASSURANCE REQUIREMENTS

The assurance requirements for the evaluation of the TOE, its development and operating environment are to choose as the predefined assurance package EAL4 augmented by the following components:

- ALC_DVS.2 (Sufficiency of security measures),
- AVA_VAN.5 (Advanced methodical vulnerability analysis).

7.4 SECURITY REQUIREMENTS DEPENDENCIES

7.4.1 SECURITY FUNCTIONAL REQUIREMENTS DEPENDENCIES

The dependence of security functional requirements for Embedded OS the security functional requirements are defined in Table 14.

195 [assignment: list of type of failures]

Table 14: Dependency of composite TOE SFRs

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
1.	FAU_SAS.1	None	----
2.	FCS_CKM.1/SM	--- FCS_CKM.2 or FCS_COP.1 --- FCS_CKM.4	---FCS.CKM.2/SM, FCS_COP.1/SEC-MSG_AES, FCS_COP.1/CMAC_AES --- FCS_CKM.4
3.	FCS_CKM.1/SM_PER-INI	--- FCS_CKM.2 or FCS_COP.1 --- FCS_CKM.4	--- FCS.CKM.2/SM_PER-INI, FCS_COP.1/SEC-MSG_AES, FCS_COP.1/CMAC_AES --- FCS_CKM.4
4.	FCS_CKM.1/RSA_KeyPair	--- FCS_CKM.2 or FCS_COP.1 --- FCS_CKM.4	--- FCS_COP.1 /SIG-VER_PKCS, FCS_COP.1 /SIG-GEN_PKCS, FCS_COP.1 /SIG-VER_9796, FCS_COP.1 /SIG-GEN_9796 --- FCS_CKM.4
5.	FCS_CKM.1/ECC_KeyPair	--- FCS_CKM.2 or FCS_COP.1 --- FCS_CKM.4	--- FCS_COP.1 /SIG-GEN_ECDSA --- FCS_CKM.4
6.	FCS_CKM.2/SM	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- FCS_CKM.1/SM --- FCS_CKM.4
7.	FCS_CKM.2/SM_PER-INI	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- FCS_CKM.1/SM_PER-INI --- FCS_CKM.4
8.	FCS_CKM.4	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	--- FCS_CKM.1/SM_PER-INI --- FCS_CKM.1/SM
9.	FCS_COP.1/SHA	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- Not fulfilled but justified. See Note 2 below. --- Not fulfilled but justified. See Note 2 below.

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
10.	FCS_COP.1/SEC-MSG_AES	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	---FCS_CKM.1/SM and FCS_CKM.1/SM_PER-INI --- FCS_CKM.4
11.	FCS_COP.1/INIT-PERSO-VER_AES	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- Not fulfilled but justified. See Note 3 below. --- Not fulfilled but justified. See Note 4 below.
12.	FCS_COP.1/ENC-DEC_AES	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- Not fulfilled but justified. See Note 5 below. --- FCS_CKM.4
13.	FCS_COP.1/ENC-DEC_TDES	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- Not fulfilled but justified. See Note 5 below. --- FCS_CKM.4
14.	FCS_COP.1/CMAC_AES	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	---FCS_CKM.1/SM and FCS_CKM.1/SM_PER-INI See also Note 5 below. --- FCS_CKM.4
15.	FCS_COP.1/CMAC_TDES	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- Not fulfilled but justified. See Note 5 below. --- FCS_CKM.4
16.	FCS_COP.1/MAC_TDES	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- Not fulfilled but justified. See Note 5 below. --- FCS_CKM.4
17.	FCS_COP.1/MAC_AES	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- Not fulfilled but justified. See Note 5 below. --- FCS_CKM.4
18.	FCS_COP.1/Retail-MAC	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- Not fulfilled but justified. See Note 5 below. --- FCS_CKM.4
19.	FCS_COP.1/SIG-GEN_PKCS#1_V1.5	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	--- FCS_CKM.1/RSA_KeyPair --- FCS_CKM.4

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
		--- FCS_CKM.4	
20.	FCS_COP.1/SIG-GEN_PKCS#1_V2.1	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- FCS_CKM.1/RSA_KeyPair --- FCS_CKM.4
21.	FCS_COP.1/SIG-GEN_9796	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- FCS_CKM.1/RSA_KeyPair --- FCS_CKM.4
22.	FCS_COP.1/SIG-GEN_ECDSA	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- FCS_CKM.1/ECC_KeyPair --- FCS_CKM.4
23.	FCS_COP.1/SIG-VER_9796	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- FCS_CKM.1/RSA_KeyPair --- FCS_CKM.4
24.	FCS_COP.1 / DEC_PKCS#1_v1.5	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- FCS_CKM.1/RSA_KeyPair --- FCS_CKM.4
25.	FCS_COP.1 / DEC_PKCS#1_v2.1	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- FCS_CKM.1/RSA_KeyPair --- FCS_CKM.4
26.	FCS_COP.1/ RSA_RAW	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- FCS_CKM.1/RSA_KeyPair --- FCS_CKM.4
27.	FCS_RNG.1	None	
28.	FDP_ACC.1/Data	--- FDP_ACF.1	--- FDP_ACF.1/Data
29.	FDP_ACC.1/FUN	--- FDP_ACF.1	--- FDP_ACF.1/FUN
30.	FDP_ACF.1/Data	--- FDP_ACC.1 --- FDP_MSA.3	--- FDP_ACC.1/Data --- Not fulfilled but justified. See Note 6 below.
31.	FDP_ACF.1/FUN	--- FDP_ACC.1	--- FDP_ACC.1/Data

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
		--- FDP_MSA.3	--- Not fulfilled but justified. See Note 9 below.
32.	FDP_UCT.1	--- FTP_ITC.1 or FTP_TRP.1 --- FDP_ACC.1 orFDP_IFC.1	--- Not fulfilled but justified. See Note 7 below. --- FDP_ACC.1
33.	FDP_UIT.1	--- FDP_ACC.1 or FDP_IFC.1 --- FTP_ITC.1 or FTP_TRP.1	--- FDP_ACC.1 --- Not fulfilled but justified. See Note 7 below.
34.	FDP_IFC.1	--- FDP_IFF.1	--- Not fulfilled but justified. See Note 8 below.
35.	FDP_ITT.1	--- FDP_IFC.1	--- FDP_IFC.1
36.	FDP_SDI.2	None	
37.	FIA_AFL.1/PIN	--- FIA_UAU.1	--- FIA_UAU.1
38.	FIA_AFL.1/ACT	--- FIA_UAU.1	--- FIA_UAU.1
39.	FIA_AFL.1/INI	--- FIA_UAU.1	--- FIA_UAU.1
40.	FIA_AFL.1/PER	--- FIA_UAU.1	--- FIA_UAU.1
41.	FIA_API.1	None	----
42.	FIA_UAU.1	--- FIA_UID.1	--- FIA_UID.1
43.	FIA_UAU.4	None	----
44.	FIA_UAU.5	None	----
45.	FIA_UAU.6	None	----
46.	FIA_UID.1	None	----
47.	FMT_LIM.1	--- FMT_LIM.2	--- FMT_LIM.2

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
48.	FMT_LIM.2	--- FMT_LIM.1	--- FMT_LIM.1
49.	FMT_SMF.1	None	----
50.	FMT_SMR.1	--- FIA_UID.1	--- FIA_UID.1
51.	FMT_MOF.1	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
52.	FMT_MSA.1	--- FDP_ACC.1 or FDP_IFC.1 --- FMT_SMR.1 --- FMT_SMF.1	--- FDP_ACC.1 --- FMT_SMR.1 --- FMT_SMF.1
53.	FMT_MTD.1/INI_PER_AUTH_DATA	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
54.	FMT_MTD.1/INI_PER_AUTH_DATA_Change	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
55.	FMT_MTD.1/Keys_and_AC_Rules_Write_and_Change	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
56.	FMT_MTD.1/PuK_Keys_Use	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
57.	FMT_MTD.1/PrK_Use	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
58.	FMT_MTD.1/PIN_Management	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
59.	FPT_EMSEC.1	None	----
60.	FPT_FLS.1	None	----
61.	FPT_ITT.1	None	----
62.	FPT_PHP.3	None	----

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
63.	FPT_TST.1	None	----
64.	FRU_FLT.2	FPT_FLS.1	FPT_FLS.1

Note 2 : A key does not exist here since a hash function does not use key(s).

Note 3 : AES keys that are covered by FCS_COP.1/INIT-PERSO-VER_AES are used for initialization and personalization agent authentication. They are written to the TOE during activation phase. Activation phase takes place within the secure environment. So FDP_ITC.1 or FDP_ITC.2 is justified by environmental countermeasures.

Note 4 : AES keys that are covered by FCS_COP.1/INIT-PERSO-VER_AES are used for initialization and personalization agent authentication. They are written during the activation sub-phase and destruction is not needed.

Note 5 : TDES and AES keys that are covered by FCS_COP.1/ENC-DEC_TDES, FCS_COP.1/ENC-DEC_AES, FCS_COP.1/CMAC_AES, FCS_COP.1/CMAC_TDES, FCS_COP.1/MAC_TDES, FCS_COP.1/MAC_AES, and FCS_COP.1/Retail-MAC are written to the TOE during either personalization or operation phase. Personalization takes place within the secure environment and writing TDES/AES keys to the TOE during operation phase requires that access conditions specified by the personalization agent be satisfied. So, FDP_ITC.1 or FDP_ITC.2 is justified either by environmental countermeasures or by SF_SMAC.

Note 6 : The TSF denies access to the objects unless their security attributes are defined. So FMT_MSA.3 is not a required for SFR FDP_ACF.1/Data properly functioning.

Note 7: There is only one communication channel between the TOE and the outer world. So FDP_UIT.1 and FDP_UCT.1 does not require FTP_ITC.1 and FTP_TRC.1.

Note 8: Security attributes are necessary for making security related decisions. Since FDP_IFC.1 applies to all data, here neither decision nor a security attribute is required. Hence there is no need to FDP_IFF.1 for FDP_IFC.1 properly functioning.

Note 9: The access control TSF according to FDP_ACF.1 uses security attributes having been defined during the manufacturing and fixed over the whole life time of the TOE. No management of the security attributes (i.e. FMT_MSA.3) is necessary here.

Note 10: There are no Embedded Operating System (EOS) defined roles to change the initial values of the special function registers (SFR) including SFRs to change MMU segmentation. Since, EOS does not generate different memory segmentations having different access rights, no such roles are defined and FMT_SMR.1 is not defined. The values of the special function registers are changed by the EOS code during system run.

7.4.2 SECURITY ASSURANCE REQUIREMENTS DEPENDENCIES

Security assurance level is EAL 4+ with added components AVA_VAN.5 and ALC_DVS.2. EAL4 is itself internally consistent. The dependencies of AVA_VAN.5 and ALC_DVS.2 are given in Table 15.

Table 15: Dependencies of composite TOE SARs

Component	Dependencies	Fulfilled or not
AVA_VAN.5	ADV_ARC.1 ADV_FSP.4 ADV_TDS.3 ADV_IMP.1 AGP_OPE.1 AGD_PRE.1 ATE_DPT.1	All dependencies are fulfilled by EAL4.
ALC_DVS.2	None	----

7.5 SECURITY FUNCTIONAL REQUIREMENTS RATIONALE

OT.Prot_Phys-Tamper:

The scenario of physical tampering as described for this objective is explicitly included in the assignment chosen for the physical tampering scenarios in FPT_PHP.3. Therefore, it is clear that this security functional requirement supports the objective.

OT.Prot_Inf_Leak:

The refinements of the security functional requirements FPT_ITT.1 and FDP_ITT.1 together with the FDP_IFC.1 explicitly require the prevention of disclosure of secret data (TSF data as well as user data) when transmitted between separate parts of the TOE or while being processed. This includes that attackers cannot reveal such data by measurements of emanations, power consumption or other behavior of the TOE while data are transmitted between or processed by TOE parts.

Embedded Operating System has added operations to TOE, PIN verification and CMAC operation. T.Information_Leakage is also valid for these operations. FPT_EMSEC.1 handles these added operations and adds refinements to protect the TSF data used by cryptographic operations.

This objective is also directed against attacks, where an attacker wants to force an information leakage, which would not occur under normal conditions. In order to achieve this, the attacker has to combine a first attack step, which modifies the behavior of the TOE (either by exposing it to extreme operating conditions or by directly manipulating it) with a second attack step measuring and analyzing some output produced by the TOE. The first step is prevented by the same mechanisms which support OT.Prot_Malfunction and OT.Prot_Phys-Tamper, respectively.

The SFRs FPT_ITT.1, FDP_ITT.1, FDP_IFC.1, FPT_EMSEC.1 addressed by OT.Prot_Inf_Leak, and the SFRs FPT_PHP.3 and FRU_FLT.2 addressed by OT.Prot_Phys-Tamper and OT.Prot_Malfunction are covered.

OT.Prot_Malfunction:

The definition of this objective shows that it covers a situation, where malfunction of the TOE might be caused by the operating conditions of the TOE (while direct manipulation of the TOE is covered OT.Prot_Phys-Tamper). There are two possibilities in this situation: Either the operating conditions are inside the tolerated range or at least one of them is outside of this range. The second case is covered by FPT_FLS.1, because it states that a secure state is preserved in this case. The first case is covered by FRU_FLT.2 because it states that the TOE operates correctly under normal (tolerated) conditions.

OT.Prot_Abuse-Func:

This objective states that abuse of functions (especially provided by the IC Dedicated Test Software, for instance in order to read secret data) must not be possible in Phase 7 of the life-cycle. There are two possibilities to achieve this: (i) They cannot be used by an attacker (i. e. its availability is limited) or (ii) using them would not be of relevant use for an attacker (i. e. its capabilities are limited) since the functions are designed in a specific way. The first possibility is specified by FMT_LIM.2 and the second one by FMT_LIM.1. Since these requirements are combined to support the policy, which is suitable to fulfill OT.Prot_Abuse-Func both security functional requirements together are suitable to meet the objective.

OT.Identification_and_Authentication:

OT.Identification_and_Authentication addresses the identification and authentication mechanisms to counter masquerade attacks and implement the identification and authentication policy. FIA_UAU.5 and FIA_API.1 require the authentication mechanisms that the TOE must have. FAU_SAS.1 supports this objective by requiring the TOE to have unique and unchangeable serial number. AKiS v2.5.2N also provides an interface for the application developer to read this serial number. FIA_UAU.4 protects the role and terminal authentication mechanisms against replay attacks and iterates of FIA_AFL.1 protect against the false PIN or authentication data tries. FDP_UCT.1 and FDP_UIT.1also covers the protection of integrity and confidentiality of the data shared. FCS_RNG.1 provides random number for key generation. They provide replay protection against replay attack for PIN authentication. FIA_UAU.6 requires the TOE to re authenticate the users after each command sent and after each reset or power-up. Finally, FCS_COP.1/SIG-GEN_9796, FCS_COP.1/SIG-VER_9796 and FCS_COP.1/RSA_RAW provide cryptographic mechanism for device and role authentication.

OT.Access_Control:

OT.Access_Control addresses user data protection against unauthorized access through logical paths. Physical paths are covered by OT.Prot_Phys-Tamper objective. FIA_UID.1 and FIA_UAU.1 protects the user data from accessing without identification and authentication. FDP_ACC.1/Data, FDP_ACC.1/FUN, FDP_ACF.1/Data and FDP_ACF.1/FUN together require the enforcement of application access control policy.

OT.Security_Management:

The aim of OT.Security_Management is that only the authorized entities that are determined by application owner can manage the TSF and TSF data. FIA_UAU.1 and FIA_UID.1 limits the actions that can be done without identification and authentication. FMT_MOF.1 and FMT_MSA.1 enable the application determined entities to change the behavior of TSF and security attributes of assets.

The SFRS FMT_MTD.1/INI_PER_AUTH_DATA, FMT_MTD.1/INI_PER_AUTH_DATA_Change, FMT_MTD.1/Keys_and_AC_Rules_Write_and_Change, FMT_MTD.1/PuK_Keys_Use, FMT_MTD.1/PrK_Use, and FMT_MTD.1/PIN_Management address the mechanisms to manage the TSF Data.

FMT_SMF.1 and FMT_SMR.1 address the management functions and roles to be implemented within the EOS.

OT.Cryptographic_Operations:

OT.Cryptographic_Operations covers the security services and security functions that the TOE will have.

FCS_CKM.1/RSA_KeyPair, FCS_COP.1/DEC_PKCS#1_V1.5, FCS_COP.1/DEC_PKCS#1_V2.1 and FCS_COP.1/RSA_RAW requirements cover the (i) asymmetric RSA key pair generation part of this objective.

FCS_CKM.1/ECC_KeyPair covers (vii) the asymmetric ECC key pair generation part of this objective.

FCS_RNG.1 covers the (ii) random number generation part of this objective.

FCS_CKM.4 covers the (vi) destruction of the keys part of this objective.

FCS_COP.1/SHA covers the (iii) hash calculation part of this objective.

FCS_COP.1/INIT-PERSO-VER_AES, FCS_COP.1/SEC-MSG_AES, FCS_COP.1/ENC-DEC_TDES, FCS_COP.1/ENC-DEC_AES, FCS_COP.1/CMAC_AES, FCS_COP.1/CMAC_TDES, FCS_COP.1/MAC_TDES, FCS_COP.1/MAC_AES, and FCS_COP.1/Retail-MAC cover the (v) symmetric cryptographic operations part of this objective.

FCS_COP.1/SIG-VER_9796, FCS_COP.1/SIG-GEN_9796, FCS_COP.1/SIG-GEN_PKCS#1_V2.1, FCS_COP.1/SIG-GEN_PKCS#1_V1.5, and FCS_COP.1/SIG-GEN_ECDSA requirements cover the (iv) eSign operations part of this objective.

Protection against SPA, DFA and DPA is addressed by OT.Prot_Inf_Leak.

OT.Secure_Communication:

OT.Secure_Communication covers the protection of communication between the TOE and the external world. To fulfill this objective TOE, generates Secure Messaging Keys with the SFRs FCS_CKM.1/SM, FCS_CKM.1/SM_PER-INI and distributes them with the SFRs FCS_CKM.2/SM, FCS_CKM.2/SM_PER-INI. FCS_COP.1/SEC-MSG_AES, FCS_COP.1/CMAC_AES provides cryptographic functions for encryption and integrity/authenticity protection of messages. FDP_UCT.1 and FDP_UIT.1 covers the protection of integrity and confidentiality of the data shared. FCS_RNG.1 provides random number for key generation. And finally FIA_UAU.6 requires the authentication of each message sent between the TOE and the external world.

OT.Storage_Integrity:

Integrity control of the hardware against physical manipulation is covered by OT.Prot_Phys-Tamper which is fulfilled by FPT_PHP.3. In addition to this, Embedded Operating System also requires the implementation of an integrity observation mechanism which is implemented by the Error Detection (EDC) for critical user data. In case of any integrity anomalies, TOE detects and inform by an error code. FPT_TST.1 covers this objective.

Table 16: Coverage of TOE objectives by SFRs

Security Functional Requirement	OT.Prot_Phys-Tamper	OT.Prot_Inf_Leak	OT.Prot_Malfunction	OT.Prot_Abuse-Func	OT.Chip_Auth_Proof	OT.Identification_and_Authentication	OT.Access_Control	OT.Security_Management	OT.Cryptographic_Operations	OT.Secure_Communication	OT.Storage_Integrity
FAU_SAS.1						✓					
FCS_CKM.1/SM										✓	
FCS_CKM.1/SM_PER-INI										✓	
FCS_CKM.1/RSA_KeyPair									✓		
FCS_CKM.1/ECC_KeyPair									✓		
FCS_CKM.2/SM										✓	
FCS_CKM.2/SM_PER-INI										✓	
FCS_CKM.4									✓		
FCS_COP.1/SHA									✓		
FCS_COP.1/SEC-MSG_AES									✓	✓	
FCS_COP.1/INIT-PERSO-VER_AES									✓		
FCS_COP.1/ENC-DEC_AES									✓		
FCS_COP.1/ENC-DEC_TDES									✓		
FCS_COP.1/CMAC_AES									✓	✓	
FCS_COP.1/CMAC_TDES									✓		
FCS_COP.1/MAC_TDES									✓		

FCS_COP.1/MAC_AES									✓		
FCS_COP.1/Retail-MAC									✓		
FCS_COP.1/SIG-GEN_PKCS#1_V1.5									✓		
FCS_COP.1/SIG-GEN_PKCS#1_V2.1									✓		
FCS_COP.1/SIG-GEN_9796					✓	✓			✓		
FCS_COP.1/SIG-GEN_ECDSA									✓		
FCS_COP.1/SIG-VER_9796						✓			✓		
FCS_COP.1/DEC_PKCS#1_V1.5									✓		
FCS_COP.1/DEC_PKCS#1_V2.1									✓		
FCS_COP.1/RSA_RAW					✓	✓			✓		
FCS_RNG.1									✓	✓	
FDP_ACC.1/Data							✓				
FDP_ACC.1/FUN							✓				
FDP_ACF.1/Data							✓				
FDP_ACF.1/FUN							✓				
FDP_UCT.1										✓	
FDP_UIT.1										✓	
FDP_IFC.1		✓		✓							
FDP_ITT.1		✓		✓							
FDP_SDI.2			✓								
FIA_AFL.1/PIN						✓					
FIA_AFL.1/ACT						✓					
FIA_AFL.1/INI						✓					
FIA_AFL.1/PER						✓					
FIA_API.1						✓					
FIA_UAU.1							✓	✓			
FIA_UAU.4						✓					
FIA_UAU.5						✓					
FIA_UAU.6								✓		✓	
FIA_UID.1							✓	✓			
FMT_LIM.1				✓							

FMT_LIM.2				✓							
FMT_SMF.1								✓			
FMT_SMR.1								✓			
FMT_MOF.1								✓			
FMT_MSA.1								✓			
FMT_MTD.1/INI_PER_AUTH_DATA								✓			
FMT_MTD.1/INI_PER_AUTH_DATA_Change								✓			
FMT_MTD.1/Keys_and_AC_Rules_Write_and_Change								✓			
FMT_MTD.1/PuK_Keys_Use								✓			
FMT_MTD.1/PrK_Use								✓			
FMT_MTD.1/PIN_Management								✓			
FPT_EMSEC.1		✓									
FPT_FLS.1		✓	✓	✓							
FPT_ITT.1		✓		✓							
FPT_PHP.3	✓	✓		✓							✓
FPT_TST.1											✓
FRU_FLT.2		✓	✓	✓							

7.6 SECURITY ASSURANCE REQUIREMENTS RATIONALE

An assurance level of EAL4 with the augmentations AVA_VAN.5 and ALC_DVS.2 are required for this type of TOE since it is intended to defend against sophisticated attacks. This evaluation assurance package was selected to permit a developer to gain maximum assurance from positive security engineering based on good commercial practices. In order to provide a meaningful level of assurance that the TOE provides an adequate level of defense against such attacks, the evaluators should have access to the detailed design knowledge and source code.

8 TOE SUMMARY SPECIFICATION

AKiS v2.5.2N is the ***composite product*** consisting of Embedded Operating System, Secure Crypto Library (platform library) and the Security IC (platform IC). Security Features are provided together with the platform and the EOS. However, the contribution weight of the platform and the EOS is different in each feature. A brief overview will be given for all security features. A detailed description also will be provided for the security features mainly provided by Embedded Operating System. For the detailed information about security features mainly provided by the platform, platform IC ST [2] and platform library ST [3] can be checked.

Security features mainly provided by the platform are given as:

- SF_OPC Control of Operating Conditions
- SF_PHY Protection against Physical Modification
- SF_LOG Logical Protection
- SF_COMP Mode Management and Protection

Security features mainly provided by Embedded Operating System are given as:

- SF_CSUP Cryptographic Support
- SF_IA Identification and Authentication
- SF_SMAC Security Management and Access Control
- SF_SM Secure Messaging

8.1 SF_OPC: CONTROL OF OPERATING CONDITIONS

Control of Operating Conditions security feature is inherited from the platform IC part of composite product AKiS v2.5.2N. For the detailed information platform IC ST [2] can be checked.

Covered SFRs are FPT_FLS.1, FRU_FLT.2.

8.2 SF_PHY: PROTECTION AGAINST PHYSICAL MODIFICATION

Protection against modification attacks security feature is inherited to the composite product AKiS v2.5.2N from the platform IC. For the detailed information platform IC ST [2] can be checked.

Added SFR by the Embedded Operating System is FPT_TST.1.

Covered SFRs are: FAU_SAS.1, FCS_RNG.1, FDP_IFC.1, FDP_ITT.1, FMT_LIM.1, FMT_LIM.2, FPT_FLS.1, FRU_FLT.2, FPT_PHP.3, FPT_ITT.1, FPT_TST.1, FCS_COP.1/INIT-PERSO-VER_AES, FCS_COP.1/SEC-

MSG_AES, FCS_COP.1/ENC-DEC_TDES, FCS_COP.1/ENC-DEC_AES, FCS_COP.1/CMAC_AES, FCS_COP.1/CMAC_TDES, FCS_COP.1/MAC_TDES, FCS_COP.1/MAC_AES, FCS_COP.1/Retail-MAC, FDP_ACC.1/Data, FDP_ACC.1/FUN, FDP_ACF.1/Data, FDP_ACF.1/FUN, FMT_MSA.1, and FMT_SMF.1

8.3 SF_LOG: LOGICAL PROTECTION

Logical Protection security feature is defined in the platform IC ST and extended in the platform library ST and further extended in this ST by EOS features. Logical Protection feature supplied by the platform IC prevents the TOE from disclosure of the TSF data and user data stored and/or processed in the security IC throughout power measurement and subsequent complex signal analysis. For detailed information platform IC ST [2] should be checked.

Logical Protection feature supplied by the platform library is mainly software countermeasures against side channel attacks and fault attacks on cryptographic operations. For detailed information platform IC ST [2] should be checked.

These features are required by FDP_ITT.1, FPT_ITT.1 and FDP_IFC.1 as well as by FPT_FLS.1.

Added security feature by the EOS is protection against side channel analysis attacks due to timing leakage of EOS added functionality as required by the FPT_EMSEC.1.

8.4 SF_COMP: MODE MANAGEMENT AND PROTECTION

Mode Management and Protection security feature is fulfilled by the platform and the Embedded Operating System. In AKiS v2.5.2N, Super System Mode and System Mode provided by the platform and User mode are used (AKiS v2.5.2N executes in User Mode). Test mode is not active in the final TOE.

When the TOE is powered, the chip starts executing in the highest privileged mode, Super System Mode, and first checks the consistency of the device. When the consistency check passes, the device will leave the Super System Mode and boot in the System Mode application; the execution then commences in the NXP System Mode OS which is responsible to properly set-up the device such that the EOS is able to access its resources (the EOS only gains access to resources via the NXP System Mode OS). When the OS booting phase has finished, the NXP System Mode can hand over control to phosExecEnv which will take care of further interaction with User Mode. Platform IC features also prevent abuse of test functions after TOE delivery. It also inhibits abuse of features, which are used during start-up or reset to configure the TOE. For the detailed information related to CPU modes, platform IC ST [2] should be checked. Covered SFRs are FMT_LIM.1, FMT_LIM.2 and FAU_SAS.1.

In addition, the memory areas and register (SFR) access rights are different in different modes, which is also provided by the platform IC. There are three modes available in the final composite TOE: Super System Mode, System Mode and User Mode. How the memory areas and SFRs are separated are explained in the platform IC ST [2] document.

AKiS v2.5.2N composite product may be delivered to the consumer before personalization. TOE also provides phase management for the sub phases defined Section 1.5.4. TSF restricts TOE functions according to the phase management.

Covered SFRs are FDP_ACC.1/FUN and FDP_ACF.1/FUN

8.5 SF_CSUP: CRYPTOGRAPHIC SUPPORT

The hardware provides many cryptographic operations as detailed in the platform IC ST and platform library ST. The composite TOE defined in this ST document adds more cryptographic operations. They are RSA Key Pair Generation, Signature Verification and Generation, RSA Decryption, TDES decryption. The keys that represent confidential information are destructed after use. Covered SFRs are FCS_CKM.1/RSA_KeyPair, FCS_CKM.1/ECC_KeyPair, FCS_CKM.4, FCS_COP.1/SHA, FCS_COP.1/SEC-MSG_AES, FCS_COP.1/INIT-PERSO-VER_AES, FCS_COP.1/ENC-DEC_AES, FCS_COP.1/ENC-DEC_TDES, FCS_COP.1/CMAC_AES, FCS_COP.1/CMAC_TDES, FCS_COP.1/MAC_TDES, FCS_COP.1/MAC_AES, FCS_COP.1/Retail-MAC, FCS_COP.1/SIG-GEN_PKCS#1_V1.5, FCS_COP.1/SIG-GEN_PKCS#1_V2.1, FCS_COP.1/SIG-GEN_9796, FCS_COP.1/SIG-GEN_ECDSA, FCS_COP.1/SIG-VER_9796, FCS_COP.1/DEC_PKCS#1_V1.5, FCS_COP.1/DEC_PKCS#1_V2.1, and FCS_COP.1/RSA_RAW. The hardware provides true random number generation as detailed in HW ST. Platform library uses hardware function to produce seeds for the deterministic random number generator. With this property FCS_RNG.1 is covered. In addition, platform library provides RNG tests for the hardware random number generator. Detailed information is given in the platform library ST [3].

8.6 SF_IA: IDENTIFICATION AND AUTHENTICATION

SF.IA includes the authentication mechanisms of activation agent authentication, initialization and personalization agent authentication, chip (terminal) authentication[196] and PIN verification mechanisms. Activation agent authentication, Initialization and personalization agent authentication and PIN verification mechanisms include authentication failure handling. Role and chip (terminal) authentication mechanisms use single user authentication and therefore are protected against replay attacks. PIN authentication mechanism is protected against replay attack by secure messaging capabilities. Other authentications are performed in secure environment as assumed in section 4.5.

Covered SFRs are FIA_AFL.1/PIN, FIA_AFL.1/ACT, FIA_AFL.1/PER, FIA_AFL.1/INI, FIA_API.1, FIA_UAU.4, FIA_UAU.5, FCS_COP.1/SIG-GEN_9796, FCS_COP.1/SIG-VER_9796, FCS_COP.1/RSA_RAW, FDP_UIT.1, FDP_UCT.1 and FCS_RNG.1.

8.7 SF_SMAC: SECURITY MANAGEMENT AND ACCESS CONTROL

The TOE includes security mechanisms to control access to TSF data and user data and also controls access to the TSF Interface. Security access rules are configurable by the application which may even allow these rules to be modified during operational phase. AKiS v2.5.2N provides application owners with a flexible access control and security management mechanism. Covered SFRs are FIA_UID.1, FIA_UAU.1, FDP_ACC.1/Data, FDP_ACF.1/Data, FMT_MTD.1/INI_PER_AUTH_DATA, FMT_MTD.1/INI_PER_AUTH_DATA_Change, FMT_MTD.1/Keys_and_AC_Rules_Write_and_Change, FMT_MTD.1/PuK_Keys_Use, FMT_MTD.1/PrK_Use, FMT_MTD.1/PIN_Management, FMT_MSA.1. These SFRs arrange the access control of the TSF Data and user data.

The other SFR covered is FMT_MOF.1 which requires that the access to TSFI is also manageable by the application allowed users.

Remaining SFRs covered by SF_SMAC are FMT_SMF.1 and FMT_SMR.1 which require the management functions and management roles. Pre-operational roles are activation agent, initialization agent, and personalization agents. Besides supporting these roles, AKiS v2.5.2N allows application owners to define additional management roles that are active in the operational phase.

196 Terminal authentication is provided by PIN authentication for SAM configuration.

8.8 SF_SM: SECURE MESSAGING

The TOE has SF.SM which allows the TOE to communicate with the external world securely. SF.SM protects the confidentiality and authenticity of the messages going between the card and the external world. Covered SFRs are FCS_CKM.1/SM, FCS_CKM.1/SM_PER-INI, FCS_CKM.2/SM, FCS_CKM.2/SM_PER-INI, FDP_UCT.1, FDP_UIT.1, FIA_UAU.6, FCS_COP.1/SEC-MSG_AES, FCS_COP.1/CMAC_AES, and FCS_RNG.1.

8.9 SECURITY FUNCTIONS RATIONALE

Table 17 shows the assignment of security functional requirements to TOE's security functionality.

Table 17: Coverage of SFRs by TOE security features

Security Functional Requirement	SF_COMP	SF_OPC	SF_PHY	SF_LOG	SF_IA	SF_SMAC	SF_SM	SF_CSUP
FAU_SAS.1	✓				✓			
FCS_CKM.1/SM							✓	
FCS_CKM.1/SM_PER-INI							✓	
FCS_CKM.1/RSA_KeyPair								✓
FCS_CKM.1/ECC_KeyPair								✓
FCS_CKM.2/SM							✓	
FCS_CKM.2/SM_PER-INI							✓	
FCS_CKM.4								✓
FCS_COP.1/SHA								✓
FCS_COP.1/SEC-MSG_AES			✓				✓	✓
FCS_COP.1/INIT-PERSO-VER_AES			✓					✓
FCS_COP.1/ENC-DEC_AES			✓					✓
FCS_COP.1/ENC-DEC_TDES			✓					✓
FCS_COP.1/CMAC_AES			✓				✓	✓
FCS_COP.1/CMAC_TDES			✓					✓
FCS_COP.1/MAC_TDES			✓					✓
FCS_COP.1/MAC_AES			✓					✓
FCS_COP.1/Retail-MAC			✓					✓
FCS_COP.1/SIG-GEN_PKCS#1_V1.5								✓
FCS_COP.1/SIG-GEN_PKCS#1_V2.1								✓
FCS_COP.1/SIG-GEN_9796					✓			
FCS_COP.1/SIG-GEN_ECDSA								✓
FCS_COP.1/SIG-VER_9796					✓			
FCS_COP.1/DEC_PKCS#1_V1.5								✓
FCS_COP.1/DEC_PKCS#1_V2.1								✓

Security Functional Requirement	SF_COMP	SF_OPC	SF_PHY	SF_LOG	SF_IA	SF_SMAC	SF_SM	SF_CSUP
FCS_COP.1/RSA_RAW					✓			
FCS_RNG.1			✓				✓	✓
FDP_ACC.1/Data						✓		
FDP_ACC.1/FUN	✓							
FDP_ACF.1/Data						✓		
FDP_ACF.1/FUN	✓							
FDP_UCT.1							✓	
FDP_UIT.1							✓	
FDP_IFC.1			✓	✓				
FDP_ITT.1			✓	✓				
FDP_SDI.2			✓					
FIA_AFL.1/PIN					✓			
FIA_AFL.1/ACT					✓			
FIA_AFL.1/INI					✓			
FIA_AFL.1/PER					✓			
FIA_API.1					✓			
FIA_UAU.1						✓		
FIA_UAU.4					✓			
FIA_UAU.5					✓			
FIA_UAU.6							✓	
FIA_UID.1						✓		
FMT_LIM.1	✓		✓					
FMT_LIM.2	✓		✓					
FMT_SMF.1						✓		
FMT_SMR.1						✓		
FMT_MOF.1						✓		
FMT_MSA.1						✓		
FMT_MTD.1/INI_PER_AUTH_DATA						✓		
FMT_MTD.1/INI_PER_AUTH_DATA_Change						✓		

Security Functional Requirement	SF_COMP	SF_OPC	SF_PHY	SF_LOG	SF_IA	SF_SMAC	SF_SM	SF_CSUP
FMT_MTD.1/Keys_and_AC_Rules_Write_and_Change						✓		
FMT_MTD.1/PuK_Keys_Use						✓		
FMT_MTD.1/PrK_Use						✓		
FMT_MTD.1/PIN_Management						✓		
FPT_EMSEC.1				✓				
FPT_FLS.1		✓	✓	✓	✓			
FPT_ITT.1			✓	✓				
FPT_PHP.3			✓					
FPT_TST.1			✓					
FRU_FLT.2		✓	✓		✓			

ABBREVIATIONS AND DEFINITIONS

AES: Advanced Encryption Standard

AKİS: Akıllı Kart İşletim Sistemi (Smart Card Operating System)

APDU: Application Packet Data Unit

CPU: Central Processing Unit

DES: Data Encryption Standard

DFA: Differential Fault Analysis

DPA: Differential Power Analysis

EAL: Evaluation Assurance Level

ECC: Elliptic Curve Cryptography

EOS: Embedded Operating System

ES: Embedded Software

IC: Integrated Circuit

PP: Protection Profile

PTG2: A class that defines the requirements for RNGs used in key generation, padding bit generation, etc. PTG.2 is defined AIS31 [15]

RAM: Random Access Memory

RSA: Ron Rivest, Adi Shamir and Leonard Adleman

ROM: Read Only Memory

SAM: Secure Access Module

SHA: Secure Hash Algorithm

SPA: Simple Power Analysis

SFR: Security Functional Requirement

ST: Security Target

TPDU: Transmission Protocol Data Unit

TOE: Target of Evaluation

BIBLIOGRAPHY

[1] Security IC Platform Protection Profile with Augmentation Packages Version 1.0, Registered and Certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-CC-PP-0084-2014

[2] NXP Secure Smart Card Controller N7021 VA Security Target Lite, Rev. 1.1, 2017-05-31

[3] Crypto Library Cobalt on N7021 VA Security Target Lite, Rev. 1.1, 5 July 2017

[4] Common Criteria for Information Technology Security Evaluation Part I: Introduction and General Model; Version 3.1 Revision 4 CCMB-2012-09-001

[5] Common Criteria for Information Technology Security Evaluation Part II: Security Functional Requirements; Version 3.1 Revision 4 CCMB-2012-09-002

[6] Common Criteria for Information Technology Security Evaluation Part III: Security Assurance Requirements; Version 3.1 Revision 4 CCMB-2012-09-003

[7] Common Criteria for Information Technology Security Evaluation, Evaluation Methodology; Version 3.1, Revision 4, CCMB-2012-09-004

[8] ISO 1177 Information Processing Character Structure For Start/Stop And Synchronous Character Oriented Transmission

[9] ISO 7816-3 Information Technology – Identification Cards – Integrated Circuits with Contacts Part 3: Electronic Signals and Transmission Protocols - T=1 Protocol

[10] ISO 7816-4 Information Technology – Identification Cards – Integrated Circuits with Contacts Part 4: Organization, security and commands for interchange

[11] ISO 7816-8 Information Technology – Identification Cards – Integrated Circuits with Contacts Part 8: Commands For Security Operations

[12] ISO 7816-9 Information Technology – Identification Cards – Integrated Circuits with Contacts Part 9: Commands for card management

[13] AKİS V2.5.2N Yönetici Kullanıcı Kılavuzu, Yayın no: 36, Yayın Tarih: 15.02.2019

[14] AKİS V2.5.2N Kullanıcı Kılavuzu, Yayın no: 27, Yayın Tarihi: 15.02.2019

[15] Functionality classes and evaluation methodology for physical random number generators AIS31, Version 2.1, 2011-12-02, Bundesamt für Sicherheit in der Informationstechnik respectively —A proposal for: Functionality classes for random number generators , Version 2.0, 2011-09-18, Wolfgang Killmann, T-Systems GEI GmbH, Werner Schindler, Bundesamt für Sicherheit in der Informationstechnik

[16] FIPS PUB 180-4, Secure Hash Standard (SHS), Federal Information Processing Standards Publication, August 2015, U.S. Department of Commerce, National Institute of Standards and Technology

[17] FIPS PUB 197, Specification for the Advanced Encryption Standard (AES), November 26, 2001, Federal Information Processing Standards Publication, U.S. Department of Commerce, National Institute of Standards and Technology

[18] NIST Special Publication 800-38A, Recommendation for Block Cipher Modes of Operation, 2001, U.S. Department of Commerce, National Institute of Standards and Technology

[19] FIPS PUB 46-3, Data Encryption Standard, 1999 October 25, U.S. Department of Commerce, National Institute of Standards and Technology

[20] NIST Special Publication 800-38B, Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, May 2005, U.S. Department of Commerce, National Institute of Standards and Technology

[21] NIST Special Publication 800-90A, Revision 1, Recommendation for Random Number Generation Using Deterministic Random Bit Generators, June 2015, Elaine Barker and John Kelsey, U.S. Department of Commerce, National Institute of Standards and Technology

[22] PKCS #1 V2.2: RSA Cryptography Standard, RSA Laboratories, October 2012

[23] Bundesamt für Sicherheit in der Informationstechnik (BSI): AIS20: Anwendungshinweise und Interpretationen zum Schema (AIS), Funktionalitäts klassen und Evaluations methodologie für deterministische Zufallszahlen generatoren (AIS20), Version 1, December 2nd, 1999

[24] A proposal for: Functionality classes for random number generators, Version 2.0, 18 September 2011

[25] ISO 9797-1 Information technology — Security techniques — Message Authentication Codes (MACs) Part 1: Mechanisms using a block cipher

[26] ISO 15946-1 Information technology — Security techniques — Cryptographic techniques based on elliptic curves Part 1: General

[27] ISO 15946-2 Information technology — Security techniques — Cryptographic techniques based on elliptic curves Part 2: Digital signatures

[28] Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen: Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung (Übersicht über geeignete Algorithmen), German “Bundesanzeiger“, BAnz AT 30.01.2015 B3