TÜBİTAK BİLGEM UEKAE

NATIONAL RESEARCH INSTITUTE OF ELECTRONICS AND CRYPTOLOGY

---

eID Applications Unit

UKİS v2.2.8H

SECURITY TARGET LITE

Revision No	01
Revision Date	19.10.2016
Document Code	UKİS-228H-STlite-01
Prepared by	eID Applications Unit
Approved by	AKİS Project Manager

Revision History

Revision #	Revision Reason	Date
01	First public version of the ST created	19.10.2016

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 ST INTRODUCTION

1.1 ST REFERENCE

Title: Security Target Lite of UKİS v2.2.8H

Document Version: 01

CC Version: 3.1 (Revision 4)

Assurance Level: EAL 4+ AVA_VAN.5

1.2 TOE REFERENCE

The current Security Target refers to the product UKİS v2.2.8H

1.3 TOE OVERVIEW

1.3.1 TOE TYPE

UKİS v2.2.8H contact based smartcard is a composite product consisting of Embedded Operating System and the Security IC.

The TOE consists of:

- UKİS v2.2.8H Embedded Operating System,
- IC Dedicated Software (Test and Support Software including Libraries),
- Security IC,
- Guidance Documentation,
- Activation Data.

1.3.2 MAJOR SECURITY PROPERTIES OF THE TOE

The TOE provides:

- Protection against modification, probing, environmental stress and emanation attacks mainly by platform specification and EOS 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,
- Identification and Authentication:
    - Activation Agent Identification & Authentication by asymmetric cryptograph 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 PIN verification,

- Cryptographic Support[1]:
  - SHA-256 Operation,
  - AES Operation[2],
  - CMAC Operation,
  - TDES Operation[3],
  - Signature Generation PKCS#1 v1.5,
  - Signature Generation PKCS#1 v2.1,
  - Signature Generation ISO/IEC 9796-2 Scheme 1,
  - Signature Verification ISO/IEC 9796-2 Scheme 1[4],
  - Asymmetric Decryption PKCS#1 v1.5,
  - Asymmetric Decryption PKCS#1 v2.1,
  - Asymmetric Encryption/Decryption RAW RSA[5],
  - Random Number Generation,
- Security Management for services and data by supporting following roles: Activation Agent, Initialization Agent, Personalization Agent, and any other roles defined by the application,
- Secure Messaging services between TOE and the terminal.

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.

UKİS v2.2.8H supports two different configurations to the Application Owner:

[1] TOE also has capability for SHA-1 operation. But it is not in the scope of evaluation because of security consideration.
[2] No interface for AES and CMAC operation is present. The services start during secure messaging automatically.
[3] No interface for TDES operation is present. TDES decryption operation is provided during Initialization and Personalization Authentication automatically.
[4] No interface for ISO/IEC 9796-2 Scheme 1 signature generation and verification is present. The services start during secure messaging automatically.
[5] No interface for RAW RSA encryption/decryption is present. The service starts during secure messaging automatically.

- "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 1.3.2 for both configurations. But, there is a slight difference between two configurations in Secure Messaging properties.

In Chip Configuration, the following two secure messaging are performed.

1. Mutual authentication between card (chip) and the terminal by certificate exchange: 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.
2. First 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 is for Terminal Authentication. 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

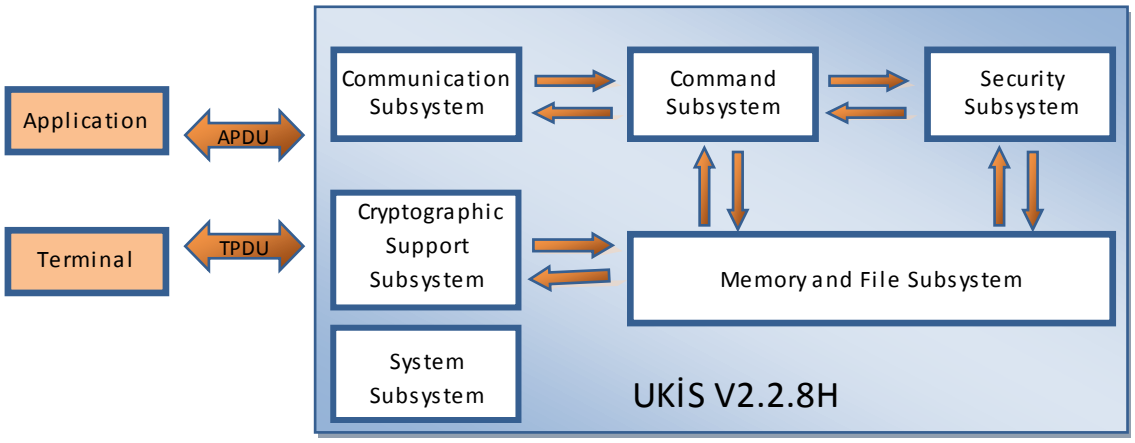


Figure 1. UKİS v2.2.8H Logical View

Communication Subsystem:

Communication subsystem manages the communication between UKİS v2.2.8H and the external world. Two layered communication takes place between the outer world and the UKİS v2.2.8H, for the transmission purposes T=1 protocol is implemented, for the application purposes APDU packets are used [8].

Cryptographic Support Subsystem:

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

Command Subsystem

Command subsystem processes the commands received from communication subsystem. It performs the commands via help of the Security Subsystem, Memory and File System Subsystem.

Security Subsystem

Access control conditions, lifecycle management and cryptographic 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 system

Memory and file system manages the non-volatile memory of the security IC. Memory and file system 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 [7] .

For the transmission:

- ISO 7816-3 Information Technology – Identification Cards – Integrated Circuits with Contacts Part 3: Electronic Signals and Transmission Protocols - T=1 Protocol [8].

For the application:

- ISO 7816-4 Information Technology – Identification Cards – Integrated Circuits with Contacts Part 4: Organization, security and commands for interchange [9].
- 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 [11].

1.5.4 LIFE CYCLE

UKİS v2.2.8H is a composite product of Security IC and Embedded Software. 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 section 1.5.4.1.

Life Cycle Phases:

Phase-1:

- Security IC Embedded Software Development

Phase-2:

- IC Development
  - IC design,
  - IC Dedicated Software development,

Phase-3:

- IC Manufacturing
  - integration and photo mask fabrication
  - IC production
  - IC testing

The Composite Product life cycle phase 4 is included in the evaluation of the IC

Phase-4:

- IC Packaging
- Security IC packaging (and testing)

Phase-5:

- Composite Product Integration
- Loading Security IC Embedded Software

Phase-6:

- Personalization Phase
  - the Composite Product personalization and testing stage where the User Data isolated into the Security IC's memory

Phase-7:

- Operational Phase
  - the Composite Product usage by its issuers and consumers (Operational Usage Phase 7) which may include loading and other management of applications in the field.

1.5.4.1 SUB PHASES OF PHASE 6 AND ADDITIONAL PHASE DEFINED FOR EMBEDDED SOFTWARE

Phase-6 is separated into three sub-phases by Embedded Software.

- Activation Sub-Phase,
- Initialization Sub-Phase,

- Personalization Sub-Phase,
- Additionally, “death phase” is added.

Activation Sub-Phase:

TOE, UKİS v2.2.8H, is activated in this phase. Initialization Key and Personalization Key are also loaded in this phase. TOE accepts only activation command and some commands that provide very limited information about TOE in this phase. The phase is ended by activation operation. It is managed by Activation Agent. A 2048 bit cryptogram is sent to TOE by Exchange Challenge Command. If the signature of sent cryptogram is verified successfully, activation is completed and 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. Listed commands 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[6]
4.	PERSONALIZATION[7]
5.	CHANGE KEY
6.	FORMAT
7.	ERASE BINARY
8.	DIR
9.	DELETE SDO
10.	GET DATA
11.	PUT DATA

6Applicapleonly in theinitializationphase
7Applicapleonly in thepersonalizationphase

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

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:

National Smartcard IC UKTÜM-H v7.01 with DES-3DES v7.01, AES256 v7.01, RSA2048 v7.01 Libraries and with IC Dedicated Software

Platform ST:

National Smartcard IC UKTÜM-H v7.01 with DES-3DES v7.01, AES256 v7.01, RSA2048 v7.01 Libraries and with IC Dedicated Software Security Target; Common Criteria CCv3.1 EAL5 augmented (EAL5+, AVA_VAN.5), Date: 2016-01-11, Version 14

Platform PP:

No PP conformance Claim, But Security IC Platform Protection Profile, Version 1.0, 15 June 2007, BSI-CC-PP-0035-2007 is used in the IC ST as guidance.

Platform Certification Report:

21.0.01/13-029-MR-01, date: 12.02.2016

Common Criteria Version:

CC v3.1 Revision 4

Platform Features:

Microprocessor: 8051 based (with 256B internal memory)

Total 192KB flash memory area configurable as

- 64KB user ROM and as 128KB data memory, or
- 128KB user ROM and as 64KB data memory

RAM: 8KB

Interfaces: Contact interface according to ISO/IEC 7816

Side Channel Resistant Crypto Engines:

- DES and 3–DES coprocessor
- AES256 coprocessor
- RSA1024 and RSA2048 coprocessor

TRNG: Physical Random Number Generator producing True Random Numbers compliant to FIBS–140–2

2.2 PLATFORM DESCRIPTION

2.2.1 PHYSICAL SCOPE OF THE PLATFORM

UKTÜM-H v7.01 IC is a contact-based smartcard IC which is designed and developed for security-based applications. It is designed by ASIC design team of YİTAL using EF250 0.25µm e-Flash CMOS process technology and design library of HHNEC. The smartcard IC is fabricated in the fab of HHNEC. The block diagram of UKTÜM-H v7.01 smartcard IC is shown in Figure 2. The hardware of the security IC consists of:

8052-type microprocessor with 256B internal RAM,
10K Test ROM storing IC Dedicated Software,
8K SRAM for volatile data storage,
3 x 64K Flash memory for non-volatile storage,
RSA2048 crypto algorithm block,
DES-3DES crypto algorithm block,
AES crypto algorithm block,
SHA-256 coprocessor block,
UART block ensuring the communication between IC and card reader according to ISO/IEC 7816-3 protocol,
Cyclic Redundancy Check module giving the opportunity to calculate 16 bit check sum according to ISO 3309 standard.
Random Number Generator block producing true random numbers,
Regulator converting external power supply of 5V to an internal supply of 2.5V,
On Chip Oscillator which produces internal clock signal,
Security sensors for sensing/preventing physical attacks,
Reset Circuitry controlling the internal reset signal production according to RESET input and security sensor outputs.

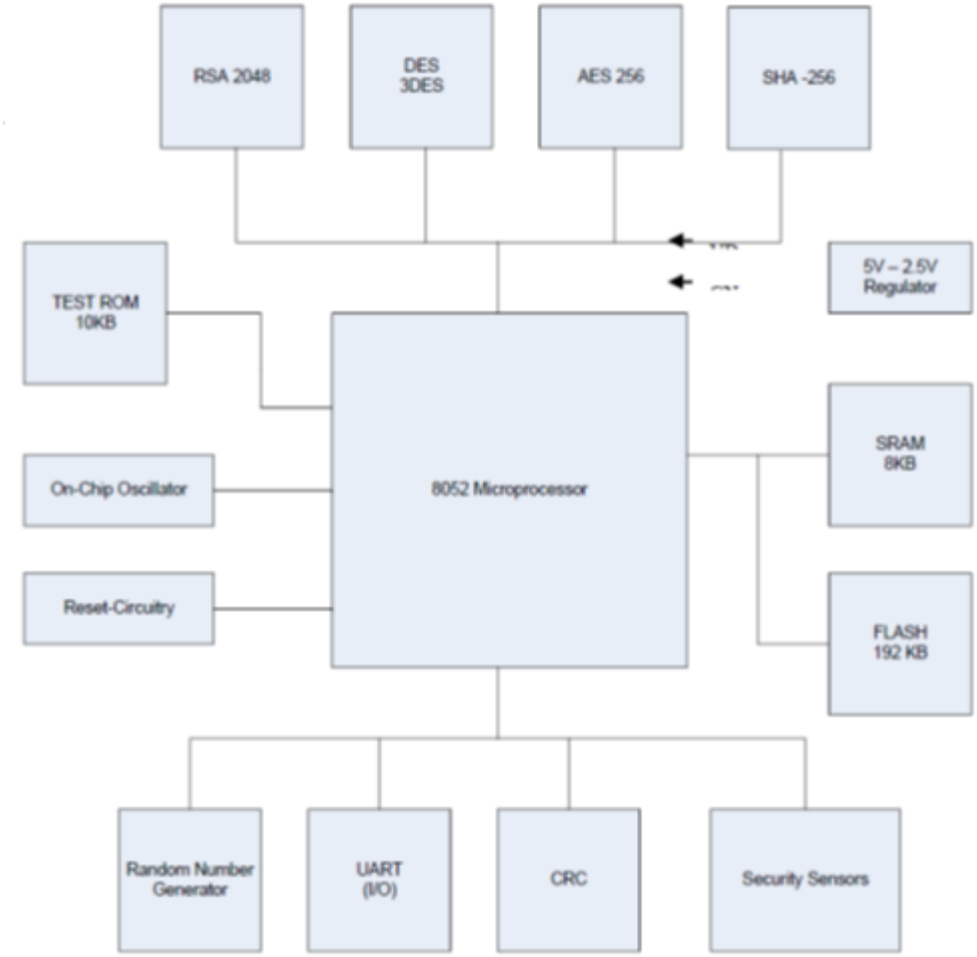


Figure 2. Block diagram of the UKİS v2.2.8H Platform

Security Sensors subsystem includes the clock frequency sensor, the internal and external supply voltage sensors and the temperature sensor which sense the operating environment. These sensors cause the smartcard IC to enter to reset state when detected environmental conditions are out of specified ranges.

The active shield and the countermeasures against the fault attacks are also parts of security circuits: The active shield, which consists of metal lines, covers the surface of the IC and prevents the attacker from probing and acquiring any useful data. In case of sensing a short-circuit or an open-circuit on the active shield, the smartcard IC enters to reset state.

The platform enters to reset state when it detects that the contents of the critical registers ensuring the proper operation of platform are corrupted due to fault attacks.

UKTÜM crypto modules have been designed to be resistant against SPA and DPA attacks. The microprocessor of the platform is equipped with additional countermeasures against power analysis attacks.

2.2.2 INTERFACES OF THE PLATFORM

- The entire surface of the IC constitutes the physical interface of the platform to the external environment.
- CLK, RESET, I/O, VDD and GND pads of the IC constitute the electrical interface of the platform to the external environment.
- The I/O pad of the IC constitutes the data input/output interface of the platform to the external environment.
- The instruction set of the platform and Special Function Registers controlling the hardware of the platform constitute platform's interface to the software environment.
- The flash memory access library constitutes the interface of the platform to flash memory access operations.
- The RSA2048 library constitutes the interface of the platform to the RSA calculations.
- The DES-3DES library constitutes the interface of the platform to the DES-3DES calculations.
- The AES library constitutes the interface of the platform to the AES calculations.

2.2.3 LOGICAL SCOPE OF THE PLATFORM

The operating system software which is loaded to the 64K/128K Flash block of the platform uses8051 instruction set to operate the smartcard IC hardware. During the platform development, driver software for accessing to special modules such as Flash memory and crypto blocks have also been developed. During the development of the operating system, Flash memory access library and, to utilize the cryptographic operations, the RSA library, the AES library, the DES and 3DES library are given to the operating system developer as source codes. The subroutines codes needed to be called as indicated in the user guidance document 'Security Requirements for Operating System' are also provided to the OS developer.

The Test ROM includes IC dedicated self-test and initialization routines and also the flash loader. The self-test software performs the operations such as the initial test after the manufacturing of the IC. Since the self-test software is deactivated after manufacturing, it is not possible to access it by the operating system. The flash loader is used by the user or/and by a subcontractor for downloading user software to Flash Memory. For these cases and whenever the user has finalized his SW-download, the user is obligated to lock the Flash loader. The final locking of the FL results in a permanent deactivation of the Flash Loader. This means that once being in the locked status, the Flash Loader cannot be reactivated anymore.

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 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 EAL 4 augmented with AVA_VAN.5 as defined in the CC, part 3.

4 SECURITY PROBLEM DEFINITION

The TOE is the embedded operating system with the security IC. Hence application 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

UKİS v2.2.8H is the composite product consisting of Embedded Operating System and the Security IC. Since the Security Target of Security IC uses the PP [1] as a guide, the assets defined in section 3.1 of the Protection Profile might be applied to this Security Target.
Additional assets are also defined below.

4.1.1 PRIMARY ASSETS

Primary assets represent User Data in the sense of the CC. They are:

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 are given in the **Table 3**:

Table 3. Secondary Assets of the TOE

Asset Name	Definition	Protection Need
PINs	TOE shall 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

Asset Name	Definition	Protection Need
Keys	Applications might need keys for their security functionality. TOE shall 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
Current Authentication Status Data	These are the data storing the current authentication status.	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
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,	Integrity, Confidentiality

Asset Name	Definition	Protection Need
	confidentiality of the TSF Code must be protected. Also its correct operation must be maintained.	
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

4.2 SUBJECTS AND EXTERNAL ENTITIES

This ST considers the following external entities and subjects:

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.
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.
ES Developer	-	The entity that designs and develops the ES.

Entity	Subject	Definition
Application Developer	-	The entity that designs and develops the Application.
IC Manufacturer	-	The entity who performs following activities: Production of the Integrated Circuit Testing the Integrated Circuit. ES 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 fulfils 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.
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 given in this section.

T.Physical_Probing

An attacker may perform physical probing of the TOE in order to disclose User Data or other critical information, to disclose/reconstruct the TOE's Embedded Software about the operation of the TOE. Physical probing requires direct interaction with the TOE's internals. Techniques commonly

employed in IC failure analysis and IC reverse engineering efforts may be used after Determination of software design including treatment of User Data, hardware security mechanisms and layout characteristics need to be identified. This pertains to "measurements" using galvanic contacts or any type of charge interaction whereas manipulations are considered under the threat "Physical Manipulation (T.Physical_Manipulation)". "Inherent Information Leakage (T.Lekage_Inherent)" and "Forced Information Leakage (T.Leakage_Forced)" may use physical probing with complex signal processing.

T.Physical_Manipulation

An attacker may physically modify the TOE in order to modify User Data/TSF Data, or TOE's hardware and Embedded Software, modify or deactivate security services of the TOE, or modify security mechanisms of the TOE to enable attacks disclosing or manipulating the User Data/TSF Data

The modification may be achieved through techniques commonly employed in IC failure analysis an d IC reverse engineering efforts. The modification may result in the deactivation of a security feature. Before that hardware security mechanisms and layout characteristics need to be identified. Determination of software design including treatment of User Data may also be a pre-requisite. Changes of circuitry or data can be permanent or temporary. In contrast to malfunctions (refer to T.Env_Malfunction) the attacker requires gathering significant knowledge about the TOE's internal construction.

T.Lekage_Inherent

An attacker may exploit information that is leaked from the TOE during usage of smart card to disclose confidential User Data or/and TSF-data. Leakage may occur through emanations, variations in power consumption, I/O characteristics, clock frequency, or by changes in processing time requirements. This leakage may be interpreted as a covert channel transmission but is more closely related to measurement of operating parameters, which may be derived either from contact or contactless interface measurements and can then be related to the specific operation being performed. Some Examples are Differential Power Analysis (DPA) and fault injection (Differential Fault Analysis).

T.Leakage_Forced

An attacker may exploit information leaking from the TOE during its usage in order to disclose confidential User/TSF Data even if the information leakage is not inherent but caused by the attacker.

This threat pertains to attacks where methods described in "Malfunction due to Environmental Stress" (refer to T.Env_Malfunction) is used to cause leakage from signals which normally do not contain significant information about secret

T.Env_Malfunction

An attacker may cause a malfunction of TSF or TOE's hardware and Embedded Software by applying environmental stress in order to modify security services of the TOE, or TOE's hardware and Embedded Software or affect security mechanisms of the TOE to enable attacks disclosing or manipulating the User Data or TSF Data. The modification of security services of the TOE may affect the quality of random numbers provided by the random number generator.

T.Abuse_Function

An attacker may use functions of the TOE which may not be used after TOE Delivery in order to (i) disclose or manipulate User Data in the TOE, (ii) manipulate (explore, bypass, deactivate or change) security services of the TOE (iii) enable an attack disclosing or manipulating the User Data or TSF Data.

T.RND

An attacker may predict or obtain information about random numbers generated by the TOE security service for instance because of a lack of entropy of the random numbers provided.

An attacker may gather information about the random numbers produced by the TOE security service. Because unpredictability is the main property of random numbers this may be a problem in case they are used to generate cryptographic keys. Here the attacker is expected to take advantage of statistical properties of the random numbers generated by the TOE. Malfunctions or premature ageing are also considered which may assist in getting information about random numbers.

4.3.2 ADDITIONAL THREATS DUE TO COMPOSITE TOE SPECIFIC FUNCTIONALITY

Terminal and communication related threats are given in this section.

T. Eavesdropping

An attacker may monitor the communication between the TOE and the terminal 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.

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.

Card Cloning and Forgery Related Threats:

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

Table 5. 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.

#	Policy Name	Definition
2.	P.PKI	There will 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 will be signed by according CA.
3.	P.Access_Control	Role attribute, PIN knowledge attribute, device authentication attribute of the user will be used as a security attribute to determine the access control behavior and security management privileges during operational phase.
4.	P.PreOperational_Security_Management	The TOE shall support • Activation Agent, • Initialization Agent, • Personalization Agent functions and roles
5.	P.Operational_Security_Management	The TOE shall support • any management function and role defined by the application

#	Policy Name	Definition

6.	P.Cryptographic_Operations	The TOE shall support following cryptographic functions:
		• RSA Key Pair Generation
		• Hash Calculation
		• eSign Operations,
		  • PKCS #1 v2.1
		  • PKCS #1 v1.5
		  • ISO/IEC 9796-2 Scheme 1
		• Asymmetric Decryption
		  • PKCS #1 v2.1
		  • PKCS #1 v1.5
		  • Raw RSA
		• Asymmetric Encryption
		  • Raw RSA
		• TDES calculation
		• AES operation
		• CMAC Operation

4.5 ASSUMPTIONS

Table 6. Composite TOE Assumptions

#	Assumption Name	Definition
1.	A.Secure_Application	Application will correctly define the access rules of the application data.
2.	A.Key_and_Certificate_Security	All keys and certificates shall be produced, stored and used securely outside of TOE.
3.	A.PIN_Handling	PINS belonging to the application shall be handled securely by PIN owner .
4.	A.Personnel_Security	Personnel who hold privileges over the TOE shall act responsibly and according to the application requirements.

#	Assumption Name	Definition
5.	A.Trusted_Parties	It is assumed that the authenticated parties that the TOE communicates act responsibly.
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 embedded operating system and the security IC. The platform (security 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.

5.1.1 PLATFORM OBJECTIVES

Platform objectives are:

- O.Physical_Probing,
- O.Physical_Manipulation,
- O.Leakage_Inherent,
- O.Leakage_Forced,
- O.Env_Malfunction,
- O.Abuse_Function,
- O.RND.

O.Physical_Probing

The TOE must provide protection against disclosure of User Data and TSF Data. This includes protection against

measuring through galvanic contacts which is direct physical probing on the chips 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 charges (using tools used in solid-state physics research and IC failure analysis)

With a prior reverse engineering to understand the design and its properties and functions. The TOE must be designed and fabricated so that it requires a high combination of complex equipment, knowledge, skill, and time to be able to derive detailed design information or other information which could be used to compromise security through such a physical attack.

O.Physical_Manipulation

The TOE must provide protection against manipulation of the TOE (including its software and Data), User Data and TSF Data. This includes protection against

reverse-engineering (understanding the design and its properties and functions),
manipulation of the hardware and any data, as well as
controlled manipulation of memory contents (Application Data).

The TOE must be designed and fabricated so that it requires a high combination of complex equipment, knowledge, skills, and time to be able to derive detailed design information or other information which could be used to compromise security through such a physical attack.

O. Leakage_Inherent

The TOE must provide protection against disclosure of confidential data stored and/or processed in the Security IC

by measurement and analysis of the shape and amplitude of signals (for example on the power, clock, or I/O lines) and
by measurement and analysis of the time between events found by measuring signals (for instance on the power, clock, or I/O lines).

This objective pertains to measurements with subsequent complex signal processing whereas O.Physical_Probing is about direct measurements on elements on the chip surface.

O. Leakage_Forced

The TOE must be protected against disclosure of confidential data processed in the TOE (using methods as described under O.Leakage_Inherent) even if the information leakage is not inherent but caused by the attacker.

by forcing a malfunction (refer to “Protection against Malfunction due to Environmental Stress (O.Env_Malfunction)” and/or
by a physical manipulation (refer to “Protection against Physical Manipulation (O.Phys-Manipulation)”.

If this is the case, signals which normally do not contain significant information about secrets could become an information channel for a leakage attack.

O.Env_Malfunction

The TOE must ensure its correct operation. The TOE must indicate or prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent malfunctions. Examples of environmental conditions are voltage, clock

frequency, temperature, or external energy fields. Remark: A malfunction of the TOE may also be caused using a direct interaction with elements on the chip surface. This is considered as being a manipulation (refer to the objective O.Phys-Manipulation) provided that detailed knowledge about the TOE´s internal construction is required and the attack is performed in a controlled manner.

O.Abuse_Function

The TOE must prevent those functions of the TOE which may not be used after TOE Delivery can be abused in order to

disclose critical User Data and TSF Data,
manipulate critical User Data and TSF Data,
bypass, deactivate, change or explore security features or security services of the TOE.

Details depend, for instance, on the capabilities of the Test Features provided by the IC Dedicated Test Software which are not specified here.

O. Identification

The TOE shall provide means to store Initialization Data and Pre-personalization Data in its non-volatile memory. The Initialization Data (or parts of them) are used for TOE identification.

O.RND

The TOE will ensure the cryptographic quality of random number generation. For instance random numbers shall not be predictable and shall have sufficient entropy. The TOE will ensure that no information about the produced random numbers is available to an attacker since they might be used for instance to generate cryptographic keys.

Note 1: Some of Platform objectives aren't included in this ST. They are either irrelevant for composite TOE or covered by other objectives. Detailed compatibility and coverage situation are defined in section 9.

5.1.2 EMBEDDED OPERATING SYSTEM OBJECTIVES

Objectives for the embedded operating system are:

O.Identification_and_Authentication,
O.Access_Control,
O.Security_Management,
O.Cryptographic_Operations,
O.Secure_Communication,
O.Storage_Integrity.

O.Identification_and_Authentication

The TOE must support following authentication mechanisms: Activation Agent Authentication, Initialization Agent Authentication, Personalization Agent Authentication, Chip Authentication, Terminal Authentication[8], Role Certificate Holder Authentication and PIN Verification.

O.Access_Control

The TOE must control the access to the user data and security services according to access control rules determined by the application. Role attribute, PIN-knowledge attribute, device authentication status and authentication shall be used as security attributes during the decision of access permission.

O.Security_Management

The TOE must support following roles: Activation Agent, Initialization Agent, Personalization Agent, and any other roles defined by the application.

O.Cryptographic_Operations

The TOE must perform following cryptographic operations: Asymmetric Key Pair Generation, Random Number Generation, Hash Calculation, eSign Operations, Symmetric Cryptographic operations.

O.Secure_Communication

The TOE must support secure communication with the terminal. TOE supports encryption, integrity and authenticity protection against attacks during communication between TOE and terminal.

O.Storage_Integrity

TOE must support storage integrity protection for User Data and TSF data.

5.2 SECURITY OBJECTIVES FOR OPERATIONAL ENVIRONMENT

Objectives for the operational environment are:

- OE.PKI,
- OE.Secure_Application,
- OE.Key_and_Certificate_Security,
- OE.PIN_Handling,
- OE.Personnel_Security,
- OE.Responsible_Parties,
- OE.Pre-Operational_Env_Sec.

[8]Provided by PIN authentication for SAM Configuration.

OE.PKI

There must 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 shall fulfill the security requirements of EOS as described in [12].

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) shall act responsibly. For example, terminal shall 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.

5.3 SECURITY OBJECTIVES RATIONALE

The justification related to the threats “Physical Probing (T.Physical_Probing)”, “Physical Manipulation (T.Physical_Manipulation)”, “Inherent Information Leakage (T.Lekage_Inherent)”, “Forced Information Leakage (T.Leakage_Forced)“, “Malfunction due to Environmental Stress (T.Env_Malfunction)”, “Abuse of Functionality (T.Abuse_Function)” and “Deficiency of Random Numbers (T.RND)” is as follows:

For all threats, the corresponding objectives in section 5.1.1 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.Physical_Manipulation and T.Env_Malfunction are also supported by additional objectives as detailed below:

T.Physical_Manipulation is mainly removed by O.Physical_Manipulation. O.Storage_Integrity also supports correspondent of the threat by detecting integrity anomalies and acting.

T.Env_Malfunction is mainly removed by O.Env_Malfunction. O.Storage_Integrity also supports correspondent of the threat by detecting integrity anomalies and acting.

Rationale for other threats describes below.

T.Eavesdropping

Eavesdropping is countered by O.Secure_Communication.

T.Session_Hijacking

Session hijacking attack is countered by O.Secure_Communication.

T.Man_in_The_Middle

Man in the middle attack is countered by O.Secure_Communication.

T.Skimmming

O.Identification_and_Authentication. O.Physical_Manipulation provides protection against physical manipulation of authenticity verification key.

T.Counterfeit

Against the Identification fraud, the TOE gives Identification and Authentication services via O.Identification_and_Authentication. Against the attacks to these services, the TOE protects the TSF data related with Identification and Authentication services. O.Physical_Probing, O.Leakage_Inherent, O.Leakage_Forced, O.Abuse_Function provides protection against disclosure of secret authentication key.

T.Unauthorised_Access

O.Access_Control handles the unauthorized access to the user data and services.

T.Unauthorised_Management

O.Security_Management put mechanisms to manage TSF data, and puts the Identification and authentication requirements for the management activities.

P.Identification_and_Authentication

O.Identification_and_Authentication covers the support for the Chip Authentication, Terminal Authentication[9], Role Holder Authentication, and PIN Verification mechanisms which are addressed by P.Identification_and_Authentication.

[9]Provided by PIN authentication for SAM Configuration.

P.PKI

OE.PKI covers the requirement of P.PKI. Additionally O.Identification_and_Authentication covers support for the Chip Authentication, Terminal Authentication[10] and Role Holder Authentication mechanisms. These authentication mechanisms include the verification of PKI hierarchy dictated by P.PKI.

P.Access_Control

O.Access_Control covers P.Access_Control.

P.PreOperational_Security_Management

O.Security_Management covers the conditions of the P.PreOperational_Security_Management.

P.Operational_Security_Management

O.Security_Management covers the conditions of the P.Operational_Security_Management.

P.Cryptographic_Operations

O.Cryptographic_Operations covers the P.Cryptographic_Operations.

A.Secure_Application

This assumption is covered by OE.Secure_Application.

A.Key_and_Certificate_Security

This assumption is covered by OE.Key_and_Certificate_Security.

A.PIN_Handling

This assumption is covered by OE.PIN_Handling

A.Personnel_Security

This assumption is covered by OE.Personnel_Security.

A.Trusted_Parties

This assumption is covered by OE.Responsible_Parties.

A.Pre-Operational_Environment

This assumption is covered by OE.Pre-Operational_Environment

Table 7. Security Objectives versus Assumptions, Threats or Policies

Threats/OSPs/Assumptions	Corresponding Objectives
T.Physical_Probing	O.Physical_Probing
T.Physical_Manipulation	O.Physical_Manipulation, O.Storage_Integrity

[10]Provided by PIN authentication for SAM Configuration.

Threats/OSPs/Assumptions	Corresponding Objectives
T.Lekage_Inherent	O.Leakage_Inherent
T.Leakage_Forced	O.Leakage_Forced
T.Env_Malfunction	O.Env_Malfunction, O.Storage_Integrity
T.Abuse_Function	O.Abuse_Function
T.RND	O.RND
T. Eavesdropping	O.Secure_Communication
T.Session_Hijacking	O.Secure_Communication
T.Man_in_The_Middle	O.Secure_Communication
T.Skimming	O.Identification_and_Authentication O.Physical_Manipulation
T.Counterfeit	O.Identification_and_Authentication O.Physical_Probing O.Leakage_Inherent O.Leakage_Forced O.Abuse_Function
T.Unauthorised_Access	O.Access_Control
T.Unauthorised_Management	O.Security_Management
P.Identification_and_Authentication	O.Identification_and_Authentication
P.PKI	O.Identification_and_Authentication, OE.PKI
P.Access_Control	O.Access_Control
P.PreOperational_Security_Management	O.Security_Management
P.Operational_Security_Management	O.Security_Management
P.Cryptographic_Operations	O.Cryptographic_Operations
A.Secure_Application	OE.Secure_Application
A.Key_and_Certificate_Security	OE.Key_and_Certificate_Security

Threats/OSPs/Assumptions	Corresponding Objectives
A.PIN_Handling	OE.PIN_Handling
A.Personnel_Security	OE.Personnel_Security
A.Trusted_Parties	OE.Responsible_Parties
A.Pre-Operational_Environment	OE.Pre-Operational_Env_Sec

6 EXTENDED COMPONENTS

There are four extended components defined and described for the TOE:

Family FAU_SAS (Audit Data Storage),

Family FMT_LIM (Limited capabilities and availability),

Family FCS_RND(Random Number Generation),

Component FPT_TST.2,

Family FIA_API (Application Proof of Identity),

Family FPT_EMSEC (TOE Emanation).

6.1 DEFINITION OF THE FAMILY FAU_SAS (AUDIT DATA STORAGE)

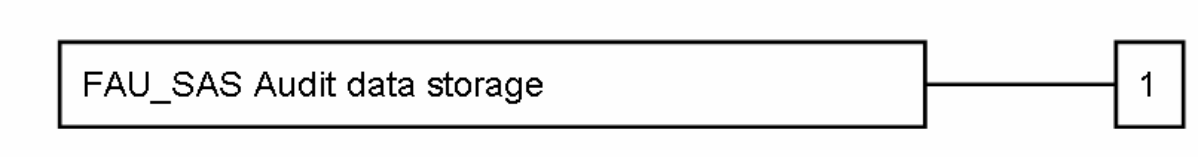
FAU_SAS family of the Class FAU (Security Audit) is defined in the platform PP document [1] and describes the functional requirements for the storage of audit data. It has a more general approach than FAU_GEN, because it does not necessarily require the data to be generated by the TOE itself and because it does not give specific details of the content of the audit records.

The family “Audit data storage (FAU_SAS)” is specified as follows.

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_RND (GENERATION OF RANDOM NUMBERS)

To define the IT security functional requirements of the TOE an additional family (FCS_RND) of the Class FCS (cryptographic support) is defined in the PP document [1]. 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 levelling:

FCS_RND.1 Generation of random numbers requires that random numbers meet a defined quality metric.

Management: FCS_RND.1

There are no management activities foreseen.

Audit:

There are no actions defined to be auditable.

6.2.1 FCS_RND.1 GENERATION OF RANDOM NUMBERS

Hierarchical to : No other components.

Dependencies : No dependencies.

FCS_RND.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_RND.1.2 The TSF shall provide random numbers that meet [assignment: a defined quality metric].

Application Note: A physical random number generator (RNG) produces the random number by a noise source based on physical random processes. A non-physical true RNG uses a noise source based on non-physical random processes like human interaction (key strokes, Mouse movement). A deterministic RNG uses a random seed to produce a pseudorandom output. A hybrid RNG combines the principles of physical and deterministic RNGs.

6.3 DEFINITION OF THE FAMILY FMT_LIM (Limited Capabilities And Availability)

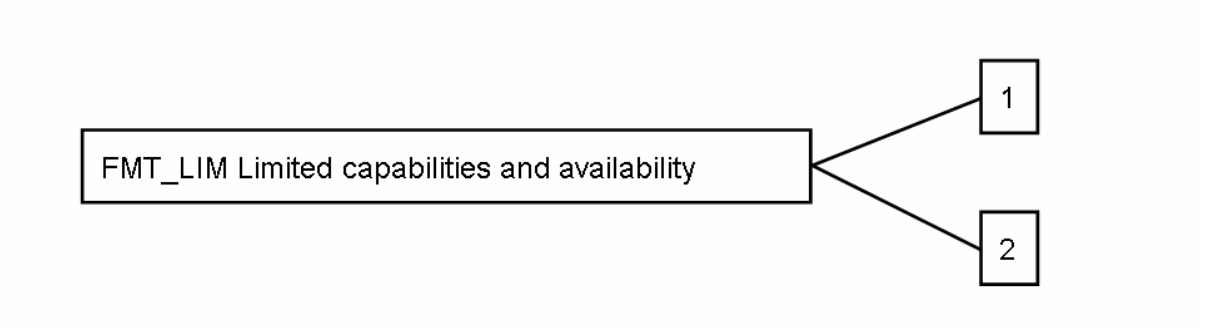
FMT_LIM of the Class FMT (Security Management) is defined as given in the IC PP [1]. This family describes the functional requirements for the Test Features of the TOE. The new functional requirements were defined in the class FMT because this class addresses the management of functions of the TSF. The 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.

The family "Limited capabilities and availability (FMT_LIM)" is specified as follows. FMT_LIM Limited capabilities and 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].

The TOE Functional Requirement "Limited availability (FMT_LIM.2)" is specified as follows.

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].

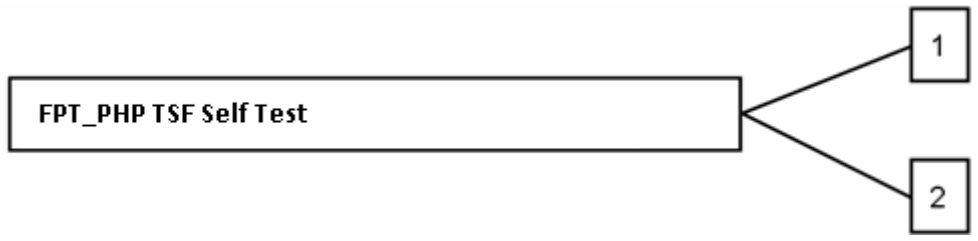
6.4 DEFINITION OF COMPONENT FPT_TST.2 (TSF Self Test)

The functional component FMT_TST.2 is defined as given in the IC ST[2].This component allows that particular parts of the security mechanisms and functions provided by the TOE can be tested after TOE Delivery or are tested automatically and continuously during normal operation transparent for the user. This security functional component is used instead of the functional component FPT_TST.1 from Common Criteria Part 2. For the user it is important to know which security functions or mechanisms can be tested. The functional component FPT_TST.1 does not mandate to explicitly specify the security functions being tested. In addition, FPT_TST.1 requires verifying the integrity of TSF data and stored TSF executable code which might violate the security policy.

Family Behavior:

The Family Behavior is defined in [3] section 15.14 (442, 443).

Component leveling:



FPT_TST.1: The component FPT_TST.1 is defined in [3] section 15.14 (444, 445, 446).

FPT_TST.2: Subset TOE security testing, provides the ability to test the correct operation of particular security functions or mechanisms. These tests may be performed at start-up, periodically, at the

request of the authorized user, or when other conditions are met. It also provides the ability to verify the integrity of TSF data and executable code.

Management: FPT_TST.2

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

Management of the conditions under which subset TSF self-testing occurs, such as during initial start-up, regular interval or under specified conditions

Management of the time interval is appropriate.

Audit: FPT_TST.2

There are no auditable events foreseen.

6.4.1 FPT_TST.2 SUBSET TOE TESTING

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_TST.2.1: The TSF shall run a suite of self-tests [selection: during initial start-up, periodically, during normal operation, at the request of the authorized user, and/or at the conditions [assignment: conditions under which self-test should occur]] to demonstrate the correct operation of [assignment: functions and/or mechanisms].

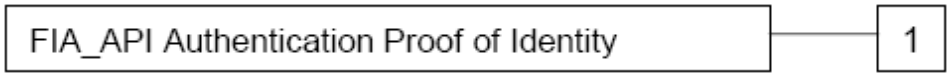
6.5 DEFINITION OF THE FAMILY FIA_API (APPLICATION PROOF OF IDENTITY)

FIA_API of the Class FIA (Identification Authentication) is defined as given in BSI-CC-PP-0056 PP [15].

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.

6.5.1 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].

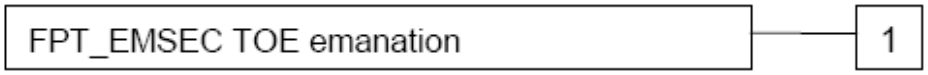
6.6 DEFINITION OF THE FAMILY FPT_EMSEC (TOE EMANATION)

FPT_EMSEC of the Class FPT (Protection of the TSF) is defined as given in BSI-CC-PP-0056 PP [15].

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 emanations 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.

6.6.1 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 type 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 Part 1 of the Common Criteria [3]. 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 by the ST author 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 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.

7.2 SECURITY FUNCTIONAL REQUIREMENTS

TOE security functional requirements of the composite product are summarized in Table 8.

Table 8. List of SFRs

CLASS FAU		
1.	FAU_SAS.1	Audit storage
CLASS FCS		
2.	FCS_COP.1/SHA	Cryptographic operation-SHA 256 Calculation
3.	FCS_COP.1/AES	Cryptographic operation-AES Calculation for Secure Messaging
4.	FCS_COP.1/TDES	Cryptographic operation- Initialization Verification with TDES
5.	FCS_COP.1/CMAC	Cryptographic operation- CMAC Calculation for Secure Messaging

6.	FCS_COP.1/SIG-GEN_PKCS#1 V1.5	Cryptographic operation-Signature Generation PKCS#1 v1.5
7.	FCS_COP.1/SIG-GEN_PKCS #1 V2.1	Cryptographic operation-Signature Generation PKCS#1 v2.1
8.	FCS_COP.1/SIG-GEN_9796	Cryptographic operation-Signature Generation ISO/IEC 9796-2 Scheme 1
9.	FCS_COP.1/SIG-VER_9796	Cryptographic operation- Signature Verification ISO/IEC 9796-2 Scheme 1
10.	FCS_COP.1/DEC_PKCS#1 V1.5	Cryptographic operation-Asymmetric Decryption PKCS#1 v.1.5
11.	FCS_COP.1/DEC_PKCS#1 V2.1 OAEP	Cryptographic operation-Asymmetric Decryption PKCS#1 v2.1
12.	FCS_COP.1/RSA_RAW	Cryptographic operation-Asymmetric Encryption/Decryption RAW RSA
13.	FCS_RND.1	Generation of random numbers
14.	FCS_CKM.1/SM	Cryptographic Key Generation - Secure Messaging Session Keys
15.	FCS_CKM.1/SM_PER-INI	Cryptographic key generation–Secure Messaging Keys for Pre-Operational Phase
16.	FCS_CKM.1/RSA_KeyPair	Cryptographic key generation- RSA Key Pair Generation
17.	FCS_CKM.2/SM	Cryptographic key distribution – Secure Messaging Keys
18.	FCS_CKM.2/SM_PER-INI	Cryptographic key distribution – Secure Messaging Keys for Pre-Operational Phases
19.	FCS_CKM.4	Cryptographic key destruction
CLASS FDP		
20.	FDP_ACC.1/Data	Subset access control-Data Access
21.	FDP_ACC.1/FUN	Subset access control-Function Access

22.	FDP_ACF.1/Data	Security attribute based access control-Data Access
23.	FDP_ACF.1/FUN	Security attribute based access control-Function Access
24.	FDP_UCT.1	Basic data exchange confidentiality
25.	FDP_UIT.1	Data exchange integrity
26.	FDP_IFC.1	Subset information flow control
27.	FDP_ITT.1	Basic internal transfer protection
28.	FDP_SDI.2/HW	Stored data integrity monitoring and action-HW Protection
29.	FDP_SDI.2/EOS	Stored data integrity monitoring and action-SW Protection
CLASS FIA		
30.	FIA_AFL.1/PIN	Authentication failure handling – PIN Verification
31.	FIA_AFL.1/ACT	Authentication failure handling – Activation
32.	FIA_AFL.1/PER	Authentication failure handling - Initialization
33.	FIA_AFL.1/INI	Authentication failure handling - Personalization
34.	FIA_API.1	Authentication Proof of Identity
35.	FIA_UAU.1	Timing of authentication
36.	FIA_UAU.4	Single use authentication mechanisms
37.	FIA_UAU.5	Multiple authentication mechanisms
38.	FIA_UAU.6	Re-authenticating
39.	FIA_UID.1	Timing of identification
CLASS FMT		
40.	FMT_LIM.1	FMT_LIM.1 Limited capabilities

41.	FMT_LIM.2	FMT_LIM.2 Limited availability
42.	FMT_SMF.1	Specification of Management Functions
43.	FMT_SMR.1	Security roles
44.	FMT_MOF.1	Management of security functions behavior
45.	FMT_MSA.1	Management of security attributes
46.	FMT_MTD.1/ INI_PER_AUTH_DATA	Management of TSF data - Initialization and Personalization Authentication Data Write
47.	FMT_MTD.1/ INI_PER_AUTH_DATA_Change	Management of TSF data - Initialization and Personalization Authentication Data Change
48.	FMT_MTD.1/ Keys_and_AC_Rules_Write_and_Change	Management of TSF data-Keys and Access Control Rules Write and Change
49.	FMT_MTD.1/PuK_Keys_Use	Management of TSF data-Public Key Usage
50.	FMT_MTD.1/PrK_Use	Management of TSF data Private Key Usage
51.	FMT_MTD.1/PIN_Management	Management of TSF data – PIN Management
CLASS FPT		
52.	FPT_EMSEC.1	TOE Emanation
53.	FPT_FLS.1	Failure with preservation of secure state
54.	FPT_ITT.1	Basic internal TSF data transfer protection
55.	FPT_PHP.3	Resistance to physical attack
56.	FPT_TST.1	TOE Testing
57.	FPT_TST.2	Subset TOE Testing

CLASS FRU		
58.	FRU_FLT.2	Limited fault tolerance

Table 9. SFRs provided by HW Document

#	Name	Title	Defined in
1.	FDP_IFC.1 [HW]	Subset information flow control	HW_ST
2.	FDP_ITT.1 [HW]	Basic internal transfer protection	HW_ST
3.	FMT_LIM.1 [HW]	Limited capabilities	HW_ST
4.	FMT_LIM.2 [HW]	Limited availability	HW_ST
5.	FPT_FLS.1 [HW]	Failure with preservation of secure state	HW_ST
6.	FPT_ITT.1 [HW]	Basic internal TSF data transfer protection	HW_ST
7.	FPT_PHP.3 [HW]	Resistance to physical attack	HW_ST
8.	FRU_FLT.2 [HW]	Limited fault tolerance	HW_ST
9.	FAU_SAS.1 [HW]	Audit storage	HW_ST
10.	FCS_RND.1 [HW]	Generation of random numbers	HW_ST
11.	FCS_CKM.1/RSA_Key Pair	Cryptographic key generation	HW_ST
12.	FCS_COP.1/DES	Cryptographic operation	HW_ST
13.	FCS_COP.1/AES	Cryptographic operation	HW_ST
14.	FCS_COP.1/RSA	Cryptographic operation	HW_ST
15.	FDP_SDI.2 [HW]	Stored data integrity monitoring and action	HW_ST
16.	FCS_RND.1 [HW]	Quality metric for random numbers	HW_ST
17.	FPT_TST.2 [HW]	Subset TOE testing	HW_ST

7.2.1 CLASS FAU: SECURITY AUDIT

The TOE shall meet the requirement "Audit storage (FAU_SAS.1)" as specified below

FAU_SAS.1 Audit storage

Hierarchical to: No other components.

Dependencies: No dependencies.

FAU_SAS.1.1 The TSF shall provide *the IC Manufacturer*[11] with the capability to store *the initialization data*[12] in the *Flash memory*[13].

7.2.2 CLASS FCS: CRYPTOGRAPHIC SUPPORT

Preface regarding Security Level related to Cryptography

The following crypto is implemented and evaluated in the TOE:

- SHA-256 Operation
- AES Operation
- 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 Verification ISO/IEC 9796-2 Scheme 1
- Asymmetric Decryption PKCS#1 v1.5
- Asymmetric Decryption PKCS#1 v2.1
- Asymmetric Encryption/Decryption RAW RSA
- Random Number Generation

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 RSA-1024 is not recommended. Therefore, for this functions it shall be checked whether the related cryptographic operations are appropriate for the intended system.

[11] [assignment: authorised users]
[12] [assignment: list of audit information]
[13] [assignment: type of persistent *memory*]

In addition, TDES 112 bit is also not recommended. Yet UKİS v2.2.8H does not supply interface for TDES 112 bit. In addition, the functions triggering TDES operations are used in the initialization and personalization sub-phases which are assumed to be carried on in physically secure environment. Therefore, no cryptographic attack due to TDES functionality is foreseen.

FCS_COP.1 /SHA Cryptographic operation-SHA 256 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*[14] in accordance with a specified cryptographic algorithm *SHA-256*[15] and cryptographic key sizes *none*[16] that meet the following: *U.S. Department of Commerce / National Institute of Standards and Technology, Secure Hash Standard (SHS), FIBS PUB 180-4, 2012-March, section 6.2 SHA-256*[17].

Application Note 2: TOE also has SHA-1 capability. But it is not in the scope of this certification.

FCS_COP.1 /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*[18] in accordance with a specified cryptographic algorithm *AES ECB and CBC Mode*[19] and cryptographic key sizes *32 bytes*[20] that meet the following:

- *AES-256: FIPS 197 Advanced Encryption Standard, NIST, November 2001,*
- *CBC Mode: Recommendation for Block Cipher Modes of Operation, NIST SP 800-38A, December 2001*[21]

[14] [assignment: list of cryptographic operations]
[15] [assignment: cryptographic algorithm]
[16] [assignment: cryptographic key sizes]
[17] [assignment: list of standards]
[18] [assignment: list of cryptographic operations]
[19] [assignment: cryptographic algorithm]
[20] [assignment: cryptographic key sizes]

Application Note 3: TOE has no interface for AES operation. It is provided automatically when secure messaging operation starts.

FCS_COP.1 /TDES Cryptographic operation-Initialization Verification with TDES

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 verification with decryption*[22] in accordance with a specified cryptographic algorithm *Triple DES*[23] and cryptographic key sizes *112 bits*[24] that meet the following:

- *U.S Department of Commerce, National Bureau of Standards Data Encryption Standard (DES), FIPS PUB 46-3, 1999 October 25. keying option 2*[25]
- *CBC Mode: Recommendation for Block Cipher Modes of Operation, NIST SP 800-38A, December 2001*[25]

Application Note 4: Applicable only for decryption form during Initialization Agent and Personalization Agent authentication.

FCS_COP.1 /CMAC Cryptographic operation-CMAC 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 but 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*[26] in accordance with a specified cryptographic algorithm *AES-CMAC*[27] and cryptographic key sizes *32 bytes*[28] that meet the following:

- *AES-256: FIPS 197 Advanced Encryption Standard, NIST, November 2001,*

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

- *NIST SP 800-38B "Recommendation For Block Cipher Modes of Operation: The CMAC Mode for Authentication" May 2005*[29]

Application Note 5: TOE has no interface for CMAC operation. It is provided automatically when secure messaging operation starts.

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] fulfilled by FCS_CKM.1/RSA_KeyPair

FCS_CKM.4 Cryptographic key destruction fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *digital signature generation*[30] in accordance with specified cryptographic algorithm *RSASSA*[31] and cryptographic key sizes 1024/*2048 bit*[32] that meet the following: *PKCS#1 v1.5, RFC 2313, March 1998.*[33]

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] fulfilled by FCS_CKM.1/RSA_KeyPair

FCS_CKM.4 Cryptographic key destruction fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *digital signature generation*[34] in accordance with a specified cryptographic algorithm *RSASSA-PSS*[35] and cryptographic key sizes *1024/2048 bit*[36] that meet the following: *PKCS#1 v2.1, RFC 3447, February 2003*[37]

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

Hierarchical to: No other components.

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

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/RSA_KeyPair
FCS_CKM.4 Cryptographic key destruction fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *digital signature generation*[38] in accordance with a specified cryptographic algorithm *RSA and SHA-256*[39] and cryptographic key sizes *1024/2048 bit*[40] that meet the following: *ISO/IEC 9796-2 Scheme 1, 2010*[41]

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

FCS_COP.1.1 The TSF shall perform *digital signature verification*[42] in accordance with a specified cryptographic algorithm *RSA and SHA-256*[43] and cryptographic key sizes *1024/2048 bit*[44] that meet the following: *ISO/IEC 9796-2 Scheme 1, December 2010*[45].

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] fulfilled by FCS_CKM.1/RSA_KeyPair
FCS_CKM.4 Cryptographic key destruction fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *asymmetric decryption*[46] in accordance with specified cryptographic algorithm *RSAES*[47] and cryptographic key sizes *1024/2048 bit*[48] that meet the following: *PKCS #1 v1.5, RFC 2313, March 1998*[49].

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

FCS_COP.1 / DEC_PKCS#1 v2.1 Cryptographic operation-Asymmetric Decryption 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] fulfilled by FCS_CKM.1/RSA_KeyPair

FCS_CKM.4 Cryptographic key destruction fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *asymmetric decryption*[50] in accordance with a specified cryptographic algorithm *RSAES-OAEP*[51] and cryptographic key sizes *1024/2048 bit*[52] that meet the following: *PKCS #1 v2.1,RFC 3447, February 2003*[53]

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] fulfilled by FCS_CKM.1/RSA_KeyPair

FCS_CKM.4 Cryptographic key destruction fulfilled by FCS_CKM.4

FCS_COP.1.1 The TSF shall perform *asymmetric encryption/decryption*[54] in accordance with a specified cryptographic algorithm *Rivest-Shamir-Adleman (RSA-Raw)*[55] and cryptographic key sizes *1024/2048 bit*[56] that meet the following: *RSA Cryptography Standard*[57]

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

FCS_RND.1 Random Number Generation

Hierarchical to: No other components

Dependencies: No dependencies

[48] [assignment: cryptographic key sizes]
[49] [assignment: list of standards]
[50] [assignment: list of cryptographic operations]
[51] [assignment: cryptographic algorithm]
[52] [assignment: cryptographic key sizes]
[53] [assignment: list of standards]
[54] [assignment: list of cryptographic operations]
[55] [assignment: cryptographic algorithm]
[56] [assignment: cryptographic key sizes]
[57] [assignment: list of standards]

FCS_RND.1.1 The TSF shall provide a physical[58] random number generator that implements *total failure test of the random source*[59]

FCS_RND.1.2 The TSF shall provide numbers that meet *the requirements of monobit, poker, runs, long run, and auto correlation tests defined in FIBS-140-1 and pass all these tests for 20.000 bit length*[60]

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/AES and FCS_COP.1/CMAC
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*[61] and specified cryptographic key sizes *32 bytes*[62] that meet the following NIST 800-56A[63].

Note 2: 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 and FCS_COP.1/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/AES, FCS_COP.1/CMAC
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 UKİS v2.2.8H*[64] and specified cryptographic key sizes *32 bytes*[65] that meet the following: *none*[66].

Application Note 7: This functionality is valid for pre-operational phases.

[58] [selection: physical, non-physical true, deterministic, hybrid]
[59] [assignment: list of security capabilities]
[60] [assignment: a defined quality metric]
[61] [assignment: cryptographic key generation algorithm]
[62] [assignment: cryptographic key sizes]
[63] [assignment: list of standards]
[64] [assignment: cryptographic key generation algorithm]
[65] [assignment: cryptographic key sizes]
[66] [assignment: list of standards]

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 OAEP, FCS_COP.1/RSA_RAW,

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 *stated as in 'TCKK Akıllı Kartlarında RSA İmzalama Anahtar Çifti Üretimi', TKRD-501-11-TA-TR01, 23 Ocak 2012, BİLGEM-UEKAE*[67] and specified cryptographic key sizes *2048 bit*[68] that meet the following: *FIPS 186-3, 2009 June*[69]

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, orFDP_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 fulfilled by FCS_CKM.4

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method *Device Authentication-Secure Messaging*[70] that meets the following: *TCKK Projesinde Kullanılan Kriptografik Algoritmalar Tanım Dokümanı, 4 Nisan 2012, v1.3, TÜBİTAK BİLGEM UEKAE Kriptoloji Birimi*[71].

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, orFDP_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 fulfilled by FCS_CKM.4

[67] [assignment: cryptographic key generation algorithm]
[68] [assignment: cryptographic key sizes]
[69] [assignment: list of standards]
[70] [assignment: cryptographic key distribution method]
[71] [assignment: list of standards]

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method: *UKİS v2.2.8H SM_PER-INI key distribution method*[72] that meets the following: *TCKK Projesinde Kullanılan Kriptografik Algoritmalar Tanım Dokümanı, 4 Nisan 2012, v1.3, TÜBİTAK BİLGEM UEKAE Kriptoloji Birimi*[73].

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 *UKİS v2.2.8H Key Destruction Method*[74] that meets the following: *none*[75].

7.2.3 CLASS FDP: USER DATA PROTECTION

FDP_ACC.1/Data Subset access control

Hierarchical to: No other components.

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

FDP_ACC.1.1: The TSF shall enforce the *Application Access Control SFP* [76] on

Subject:

- *Initialization Agent*
- *Personalization Agent*
- *Terminal*
- *Application defined and allowed role*

Objects: User data stored

Operations: write, create, read, delete[77]

FDP_ACC.1/FUN Subset access control

Hierarchical to: No other components.

[72] [assignment: cryptographic key distribution method]
[73] [assignment: list of standards]
[74] [assignment: cryptographic key destruction method]
[75] [assignment: list of standards]
[76] [assignment: access control SFP]
[77] [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]

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

FDP_ACC.1.1: The TSF shall enforce the *Application Access Control SFP*[78] on

Subject:

- *Activation Agent*
- *Initialization Agent*
- *Personalization Agent*
- *Application defined and allowed role*

Objects and operations as referred to in

- *Defined command function for Activation Sub-phase in document [12]*
- *Defined command function for Initialization Sub-phase in document [12]*
- *Defined command function for Personalization Sub-phase in document [12]*
- *Defined command function for Operation Phase in document [12]*
- *Defined command function for Death Phase in document [12]*[79]

FDP_ACF.1/Data Security attributes based access control

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control fulfilled by FDP_ACC.1

FMT_MSA.3 Static attribute initialization not fulfilled but justified

FDP_ACF.1.1 The TSF shall enforce the *Application Access Control SFP*[80] to objects based on the following:

Subject:

- *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.[81]

[78] [assignment: access control SFP]
[79] [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]
[80] [assignment: access control SFP]

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*[82] *have read, write, and change access according to rules determined by application developer*[83].

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*[84].

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*[85].

FDP_ACF.1/FUN Security attributes based access control

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control fulfilled by FDP_ACC.1

FMT_MSA.3 Static attribute initialization not fulfilled but justified

FDP_ACF.1.1 The TSF shall enforce the *Application Access Control SFP*[86] to objects based on the following:

Subject:

- *Activation Agent*
- *Initialization Agent*
- *Personalization Agent*
- *Application defined and allowed roles*

objects, and their attributes as referred to in.

- *Defined command function for Activation Sub-phase in document [12]*
- *Defined command function for Initialization Sub-phase in document [12]*

[81] [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]
[82] It means PIN authenticated terminal for SAM configuration.
[83] [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]
[84] [assignment: rules, based on security attributes that explicitly authorise access of subjects to objects]
[85] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]
[86] [assignment: access control SFP]

- *Defined command function for Personalization Sub-phase in document [12]*
- *Defined command function for Operation Phase in document [12]*
- *Defined command function for Death Phase in document [12]*[87]

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

- *Only Activation Agent access Defined command function for Activation Sub-phase in document [12]*
- *Only initialization Agent Access Defined command function for Initialization Sub-phase in document [12]*
- *Only Personalization Agent Defined command function for Personalization Sub-phase in document [12]*
- *Only Application defined and allowed roles access Defined command function for Operation Phase in document [12]*[88].

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 function for Death Phase in document [12]*[89].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *none*[90].

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] fulfilled by FDP_ACC.1

FDP_UCT.1.1 The TSF shall enforce the *Application Access Control SFP* [91] to transmit, receive[92] user data in a manner protected from unauthorized disclosure.

[87] [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]
[88] [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]
[89] [assignment: rules, based on security attributes that explicitly authorise access of subjects to objects]
[90] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]
[91] [assignment: access control SFP(s) and/or information flow control SFP(s)]
[92] [selection: transmit, receive]

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

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*[93] to transmit, receive[94] user data in a manner protected from modification, deletion, insertion, replay[95] errors.

FDP_UIT.1.2 The TSF shall be able to determine on receipt of user data, whether modification, deletion, insertion, replay[96] has occurred.

Application Note 9: 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 *Platform Data Processing Policy*[97] on *all confidential data when they are processed between the different parts of the TOE*[98].

Refinement : Platform 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

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*[99] to prevent the *disclosure*[100] of user data when it is transmitted between physically-separated parts of the TOE.

---

[93] [assignment: access control SFP(s) and/or information flow control SFP(s)]
[94] [selection: transmit, receive]
[95] [selection: modification, deletion, insertion, replay]
[96] [selection: modification, deletion, insertion, replay]
[97] [assignment: information flow control SFP]
[98] [assignment: list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP]

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/HW 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 *unmatched contents with related checksums*[101] on all objects, based on the following attributes: *checkbits for Internal RAM and SRAM*[102].

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall *produce an internal reset related to data reads with unmatched checkbits in Internal RAM and SRAM*[103].

FDP_SDI.2/EOS 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*[104] on all objects, based on the following attributes: *EDC value check of Files and File Headers stored in the EEPROM when the active file is to be changed.*[105]

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall *inform the user by a warning code*[106].

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*[107] unsuccessful authentication attempts occur related to *PIN authentication event*[108].

---

[99] [assignment: access control SFP(s) and/or information flow control SFP(s)]
[100] [selection: disclosure, modification, loss of use]
[101] [assignment: integrity errors]
[102] [assignment: user data attributes]
[103] [assignment: action to be taken]

[104] [assignment: integrity errors]
[105] [assignment: user data attributes]
[106] [assignment: action to be taken]

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met[109], the TSF shall *block the usage of PIN*[110].

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*[111] unsuccessful authentication attempts occur related to *Activation Role Authentication*[112].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met[113], the TSF shall *put the card into death phase*[114].

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*[115] unsuccessful authentication attempts occur related to *Initialization Agent Authentication*[116].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met[117], the TSF shall *put the card into death phase*[118].

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*[119] unsuccessful authentication attempts occur related to *personalization agent authentication*[120].

---

[107][selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]
[108][assignment: list of authentication events]
[109][selection: met, surpassed]
[110][assignment: list of actions]
[111][selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]
[112][assignment: list of authentication events]
[113][selection: met, surpassed]
[114][assignment: list of actions]
[115][selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]
[116][assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]
[117][selection: met, surpassed]
[118][assignment: list of actions].
[119][selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]
[120][assignment: list of authentication events].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been *met*[121], the TSF shall *put the card into death phase*[122].

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*[123] to prove the identity of the *card itself*[124].

Application Note 10: 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

FIA_UAU.1.1 The TSF shall allow

- *to read chip serial number: at pre-operational, operational and death phases*
- *to perform any application allowed actions*[125]

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*
- *Role Holder Authentication*[126]

Application Note 11: 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

[121][selection: met, surpassed]
[122][assignment: list of actions]
[123][assignment: authentication mechanism]
[124][authorised user or the role]
[125][assignment: list of TSF mediated actions]
[126][assignment: identified authentication mechanism(s)]

reused normally. But this situation does not cause a security flaw by means of secure messaging capabilities.

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*[127]
- *Role Authentication*
- *PIN Authentication*][128]

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

- *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*[129].

FIA_UAU.6 Re-authenticating

Hierarchical to: No other components.

Dependencies: No dependencies.

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

- *each reset or power-up,*

[127] It means PIN Authentication for SAM Configuration.
[128] [assignment: list of multiple authentication mechanisms]
[129] [assignment: rules describing how the multiple authentication mechanisms provide authentication]

- *each command sent to the TOE during Secure Messaging*[130].

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*
- *to perform any application allowed action*[131]

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 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: *after the submission of the TOE (After Phase 4), Self Test Software which is part of the IC Dedicated Software does not permit to collect any data which causes to disclose or change the User Data and/or the TSF data or any other.*[132]

Refinement: 'Capabilities' are the functions implemented in the Self Test Software as part of the IC Dedicated Software.

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: *after the submission of the TOE (After Phase 4), Self Test Software which is part of the IC*

[130] [assignment: list of conditions under which re-authentication is required]
[131] [assignment: list of TSF-mediated actions]
[132] [assignment: Limited capability and availability policy]

Dedicated Software does not permit to collect any data which causes to disclose or change the User Data and/or the TSF data or any other[133].

Refinement: 'Availability' is availability of the functions implemented in the Self Test Software as part of the IC Dedicated Software.

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*[134]

FMT_SMR.1 Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1/ Timing of identification 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*[135]

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

Application Note 12: The term "role" in this SFR is used as general Word in CC Part 2 and not about authenticated role holder.

FMT_MOF.1 Management of security functions behavior

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_MOF.1.1 The TSF shall restrict the ability to disable and enable[136] the functions

[133] [assignment: Limited capability and availability policy]
[134] [assignment: list of management functions to be provided by the TSF]
[135] [assignment: the authorised identified roles]
[136] [selection: determine the behaviour of, disable, enable, modify the behaviour of]

- *external interface command for operational mode listed in [13]in* [137] *to application defined roles*[138].

Application Note 13: Applicable only for operational phase. Not applicable for activation, initialization and personalization.

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] fulfilled by FDP_ACC.1

FMT_SMR.1 Security roles fulfilled by FMT_SMR.1

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

FMT_MSA.1.1 The TSF shall enforce the *Application Access Control Policy*[139] to restrict the ability to query, modify, delete[140] the security attributes *access control rules of keys, PINs, user data*[141] to *Initialization Agent, Personalization Agents and application defined roles*[142].

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[143] the *authentication reference data for Initialization and Personalization Agents*[144] to *Activation Agent*[145].

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

[137] [assignment: list of functions]
[138] [assignment: the authorised identified roles]
[139] [assignment: access control SFP(s), information flow control SFP(s)]
[140] [selection: change_default, query, modify, delete, [assignment: other operations]]
[141] [assignment: list of security attributes]
[142] [assignment: the authorised identified roles]
[143] [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
[144] [assignment: list of TSF data]
[145] [assignment: the authorised identified roles]

FMT_MTD.1.1 The TSF shall restrict the ability to change[146] the *authentication reference data for Initialization and Personalization Agents*[147] to *Initialization and Personalization Agents*[148].

FMT_MTD.1/Keys_and_AC_Rules_Write_and_ChangeManagement 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[149] the *root Certificate Authority public key, Chip Authentication PuK and PrK and Access Control Rules*[150] to *Initialization Agent, Personalization Agent any application defined and allowed role*[151].

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[152]the *Root CA PuK and Chip Authentication PuK*[153] to *application defined and allowed roles*[154].

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 fulfilled by FMT_SMF.1

FMT_MTD.1.1 The TSF shall restrict the ability to use[155]the *Chip Authentication PrK*[156]to *application defined and allowed roles*[157].

FMT_MTD.1/PIN_Management Management of TSF data PIN Management

Hierarchical to: No other components.

---

[146][selection: change_default, query, modify, delete, clear, [assignment: other operations]]
[147][assignment: list of TSF data]
[148][assignment: the authorised identified roles]
[149][selection: change_default, query, modify, delete, clear, [assignment: other operations]]
[150][assignment: list of TSF data]
[151] [assignment: the authorised identified roles]
[152] [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
[153] [assignment: list of TSF data]
[154] [assignment: the authorised identified roles]
[155] [selection: change_default, query, modify, delete, clear, [assignment: other operations]]
[156] [assignment: list of TSF data]
[157] [assignment: the authorised identified roles]

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[158] the *PIN objects*[159] to *Initialization Agent, Personalization Agents, any application defined and allowed roles*[160].

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*[161] in excess of *non-useful information*[162] enabling access to *Initialization and Personalization Keys*, *PINs used by the application*[163], and *none*[164].

FPT_EMSEC.1.2 The TSF shall ensure *any users*[165] are unable to use the following interface *contact interface and physical contacts*[166] to gain access to *none*[167].

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:

- *The temperature of the operating environment goes out of the specified ranges;*
- *The external supply voltage goes out of the specified ranges;*
- *Clock frequency goes out of the specified ranges*[168].

Refinement: The term “failure” above also covers “circumstances”. The TOE prevents failures for the circumstances” defined above. When the sensors which senses these conditions raises the ALARM signal, the device enters to reset state. [169].

[158] [selection: change, default, query, modify, delete, clear, [assignment: other operations]]
[159] [assignment: list of TSF data]
[160] [assignment: the authorised identified roles]
[161] [assignment: types of emissions]
[162] [assignment: specified limits]
[163] [assignment: list of types of TSF data]
[164] [assignment: list of types of user data].
[165] [assignment: type of users]
[166] [assignment: type of connection]
[167] [assignment: list of type of user data].
[168] [assignment: list of types of failures in the TSF]

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*[170] 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.

Application Note 14: This requirement is equivalent to FDP_ITT.1 above but refers to TSF data instead of User Data. Therefore, it should be understood as to refer to the same Data Processing Policy defined under FDP_IFC.1.

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*[171] to the *microprocessor, ciphering blocks, SRAM and Flash memories, data and address busses between microprocessor and ciphering blocks, data busses between microprocessor and SRAM and Flash memories storing the user data, data and address busses between microprocessor and Flash memories storing the Security IC Embedded Software* [172] by responding automatically such that the SFRs are always enforced.

Refinement: The TSF shall 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.

[169] [assignment: list of types of failures in the TSF]
[170] [selection: disclosure, modification]
[171] [assignment: physical tampering scenarios]
[172] [assignment: list of TSF devices/elements]

FPT_TST.1.1 The TSF shall run a suite of self-tests before critical operations[173] to demonstrate the **integrity of TSF Data except EOS Code** and correct operation of the TSF[174].

FPT_TST.1.2 The TSF shall ~~provide authenticated users with the capability to~~ verify the integrity of *TSF Data except EOS Code and Security Services*[175].

~~FPT_TST.1.3 The TSF shall provide authenticated users with the capability to verify the integrity of [selection: [assignment: parts of TSF], TSF].~~

FPT_TST.2 Subset TOE testing

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_TST.2.1 The TSF shall run a suite of self-tests during initial startup *and at the cases that the operating system requires*[176] to demonstrate the correct operation of the *active shield, security sensors, random number generator and DES-3DES, AES, RSA cryptographic operations*[177].

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 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)]*[178].

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

Application Note 15: Environmental conditions include but are not limited to power supply, clock, and other external signals (e.g. reset signal) necessary for the TOE operation.

[173][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]]
[174][selection: [assignment: parts of TSF], the TSF]
[175][selection: [assignment: parts of TSF data], TSF data]
[176][selection: during initial start-up, periodically, during normal operation, at the request of the authorized user, and/or at the conditions [assignment: conditions under which self-test should occur]]
[177][assignment: functions and/or mechanisms]
[178][assignment: list of type of failures]

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 EAL 4 augmented by the following component:

- 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 the following Table.

Table 10. Dependency of Composite TOE SFRs

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this PP
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/AES, FCS_COP.1/CMAC --- 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/AES, FCS_COP.1/CMAC --- 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.2/SM	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- FCS_CKM.1/SM --- FCS_CKM.4
6.	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
7.	FCS_CKM.4	None	----
8.	FCS_COP.1/SHA	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	--- Not fulfilled but justified. See Explanation 1

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this PP
		--- FCS_CKM.4	--- Not fulfilled but justified. See Explanation 1
9.	FCS_COP.1/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
10.	FCS_COP.1/TDES	--- FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 --- FCS_CKM.4	--- Not fulfilled but justified. See Explanation 2 --- Not fulfilled but justified. See Explanation 3
11.	FCS_COP.1/CMAC	--- 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
12.	FCS_COP.1/SIG-GEN_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
13.	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
14.	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
15.	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
16.	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
17.	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
18.	FCS_COP.1/ RSA_RAW	--- FDP_ITC.1 or FDP_ITC.2 or	--- FCS_CKM.1/RSA_KeyPair

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this PP
		FCS_CKM.1 --- FCS_CKM.4	 --- FCS_CKM.4
19.	FCS_RND.1	None	----
20.	FDP_ACC.1/Data	--- FDP_ACF.1/Data	--- FDP_ACF.1/Data
21.	FDP_ACC.1/FUN	--- FDP_ACF.1/FUN	--- FDP_ACF.1/FUN
22.	FDP_ACF.1/Data	--- FDP_ACC.1/Data --- FDP_MSA.3	--- FDP_ACC.1/Data --- Not fulfilled but justified. See Explanation 4
23.	FDP_ACF.1/FUN	--- FDP_ACC.1/Data --- FDP_MSA.3	--- FDP_ACC.1/Data --- Not fulfilled but justified. See Explanation 7
24.	FDP_UCT.1	--- FTP_ITC.1 or FTP_TRP.1 --- FDP_ACC.1 or FDP_IFC.1	--- Not fulfilled but justified. See Explanation 5 --- FDP_ACC.1
25.	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 Explanation 5
26.	FDP_IFC.1	--- FDP_IFF.1	--- Not fulfilled but justified. See Explanation 6
27.	FDP_ITT.1	--- FDP_IFC.1	--- FDP_IFC.1
28.	FDP_SDI.2/HW	None	----
29.	FDP_SDI.2/EOS	None	----
30.	FIA_AFL.1/PIN	--- FIA_UAU.1	--- FIA_UAU.1
31.	FIA_AFL.1/ACT	--- FIA_UAU.1	--- FIA_UAU.1
32.	FIA_AFL.1/PER	--- FIA_UAU.1	--- FIA_UAU.1

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this PP
33.	FIA_AFL.1/INI	--- FIA_UAU.1	--- FIA_UAU.1
34.	FIA_API.1	None	----
35.	FIA_UAU.1	--- FIA_UID.1	--- FIA_UID.1
36.	FIA_UAU.4	None	----
37.	FIA_UAU.5	None	----
38.	FIA_UAU.6	None	----
39.	FIA_UID.1	None	----
40.	FMT_LIM.1	--- FMT_LIM.2	--- FMT_LIM.2
41.	FMT_LIM.2	--- FMT_LIM.1	--- FMT_LIM.1
42.	FMT_SMF.1	None	----
43.	FMT_SMR.1	--- FIA_UID.1	--- FIA_UID.1
44.	FMT_MOF.1	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
45.	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
46.	FMT_MTD.1/INI_PER_AUTH_DATA	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
47.	FMT_MTD.1/INI_PER_AUTH_DATA_Change	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
48.	FMT_MTD.1/Keys_and_AC_Rules_Write_and_Change	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1

#	Security Functional Requirement	Dependencies	Fulfilled by security requirements in this PP
49.	FMT_MTD.1/PuK_Keys_Use	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
50.	FMT_MTD.1/PrK_Use	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
51.	FMT_MTD.1/PIN_Management	--- FMT_SMR.1 --- FMT_SMF.1	--- FMT_SMR.1 --- FMT_SMF.1
52.	FPT_EMSEC.1	None	----
53.	FPT_TST.1	None	----
54.	FPT_FLS.1	None	----
55.	FPT_ITT.1	None	----
56.	FPT_PHP.3	None	----
57.	FRU_FLT.2	--- FPT_FLS.1	--- FPT_FLS.1

Explanation 1: A key does not exist here since a hash function does not use key(s)

Explanation 2: TDES keys 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.

Explanation 3: TDES keys are used for Initialization and Personalization Agent authentication. They are written during the activation sub-phase and destruction is not needed.

Explanation 4: 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.

Explanation 5: 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.

Explanation 6: 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.

Explanation 7: 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 these security attributes (i.e. FMT_MSA.3) is necessary here.

7.4.2 SECURITY ASSURANCE REQUIREMENTS DEPENDENCIES

Security assurance level is EAL 4+ with added component AVA_VAN.5. EAL 4 is itself internally consistent. The dependencies of AVA_VAN.5 are given below

Table 11. Composite TOE SAR Dependencies

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 EAL 4.

7.5 SECURITY FUNCTIONAL REQUIREMENTS RATIONALE

O.Physical_Probing

The scenario of physical probing 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.

O.Physical_Manipulation

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

The security functional requirement FPT_TST.2 will detect attempts to conduce a physical manipulation on the monitoring functions of the TOE. The objective of FPT_TST.2 is O.Phys-Manipulation. The physical manipulation will be tried to overcome security enforcing functions.

O.Leakage_Inherent

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.Lekage_Inherent 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.

O.Leakage_Forced

This objective is 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 O.Env_Malfunction and O.Physical_Manipulation, respectively. The requirements covering O.Leakage_Inherent also support O.Leakage_Forced because they prevent the attacker from being successful if he tries the second step directly.

O.Env_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

O.Physical_Manipulation). 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.

O.Abuse_Function

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 O.Abuse_Function both security functional requirements together are suitable to meet the objective.

O.RND

FCS_RND.1 requires the TOE to provide random numbers of good quality. To specify the exact metric is left to the individual Security Target for a specific TOE. Other security functional requirements, which prevent physical manipulation and malfunction of the TOE (FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FRU_FLT.2, FPT_FLS.1, FPT_PHP.3) support this objective because they prevent attackers from manipulating or otherwise affecting the random number generator.

O.Identification_and_Authentication

O.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.1supports this objective by requiring the TOE to have unique and unchangeable serial number. UKİS v2.2.8H 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_RND.1 provides random number for key generation. They provides 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.

O.Access_Control

O.Access_Control addresses user data protection against unauthorized access through logical paths. Physical paths are covered by O.Physical_Probing and O.Physical_Manipulation objectives. 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.

O.Security_Management

Goal of O.Security_Management is only authorized entities who 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 enables the application determined entities to change to 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, 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 TOE.

O.Cryptographic_Operations

Objective O.Cryptographic_Operations covers the security services and security functions that the TOE will have. The SFRs: FCS_CKM.1/RSA_KeyPair, FCS_CKM.4, FCS_COP.1/SHA, FCS_COP.1/TDES, 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-VER_9796, FCS_COP.1/DEC_PKCS#1 V1.5, FCS_COP.1/DEC_PKCS#1 V2.1 OAEP, FCS_COP.1/RSA_RAW FCS_COP.1/AES, FCS_COP.1/CMAC, FCS_RND.1, totally cover the O.Cryptographic_Operations. Protection against SPA, DFA and DPA are addressed within the O.Leakage_Inherent.

O.Secure_Communication

Objective O.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/AES, FCS_COP.1/CMAC 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_RND.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.

O.Storage_Integrity

The security functional requirement “Stored data integrity monitoring and action” (FDP_SDI.2/HW) requires the implementation of an integrity observation which is implemented by the Error Detection (EDC) measure. The EDC is present throughout the internal RAM and SRAM of the Security IC.

Embedded OS also requires the implementation of an integrity observation mechanism which is implemented by the Error Detection (EDC) for critical user data (FDP_SDI.2/EOS). In case of any integrity anomalies, TOE detects and informs by an error code.

Therefore, the security functional requirements FDP_SDI.2/HW and FDP_SDI.2/EOS are suitable to meet the security objective.

Embedded OS provides the same mechanism for the integrity of critical TSF data. Therefore, the security functional requirement FPT_TST.1 is also suitable to meet this security objective.

Table 12. Coverage of TOE Objectives by SFRs

Security Functional Requirement	O.Physical_Probing	O.Physical_Manipulation	O.Leakage_Inherent	O.Leakage_Forced	O.Env_Malfunction	O.Abuse_Function	O.RND	O.Identification_and_Authentication	O.Access_Control	O.Security_Management	O.Cryptographic_Operations	O.Secure_Communication	O.Storage_Integrity
FAU_SAS.1								✓					
FCS_CKM.1/SM												✓	
FCS_CKM.1/SM_PER-INI												✓	
FCS_CKM.1/RSA_KeyPair											✓		
FCS_CKM.2/SM												✓	
FCS_CKM.2/SM_PER-INI												✓	
FCS_CKM.4											✓		
FCS_COP.1/SHA											✓		
FCS_COP.1/AES											✓	✓	
FCS_COP.1/TDES											✓		
FCS_COP.1/CMAC											✓	✓	
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-VER_9796								✓			✓		
FCS_COP.1/DEC_PKCS#1 V1.5											✓		
FCS_COP.1/DEC_PKCS#1 V2.1 OAEP											✓		
FCS_COP.1/RSA_RAW								✓			✓		
FCS_RND.1							✓	✓			✓	✓	
FDP_ACC.1/Data									✓				
FDP_ACC.1/Function									✓				
FDP_ACF.1/Data									✓				

Security Functional Requirement	O.Physical_Probing	O.Physical_Manipulation	O.Leakage_Inherent	O.Leakage_Forced	O.Env_Malfunction	O.Abuse_Function	O.RND	O.Identification_and_Authentication	O.Access_Control	O.Security_Management	O.Cryptographic_Operations	O.Secure_Communication	O.Storage_Integrity
FDP_ACF.1/Function									✓				
FDP_UCT.1								✓				✓	
FDP_UIT.1							✓	✓				✓	
FDP_IFC.1			✓	✓			✓						
FDP_ITT.1			✓	✓									
FDP_SDI.2/HW					✓								✓
FDP_SDI.2/EOS					✓								✓
FIA_AFL.1/PIN								✓					
FIA_AFL.1/ACT								✓					
FIA_AFL.1/PER								✓					
FIA_AFL.1/INI								✓					
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_A										✓			

Security Functional Requirement	O.Physical_Probing	O.Physical_Manipulation	O.Leakage_Inherent	O.Leakage_Forced	O.Env_Malfunction	O.Abuse_Function	O.RND	O.Identification_and_Authentication	O.Access_Control	O.Security_Management	O.Cryptographic_Operations	O.Secure_Communication	O.Storage_Integrity
C_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													✓
FPT_TST.2		✓											
FRU_FLT.2				✓	✓		✓						

7.6 SECURITY ASSURANCE REQUIREMENTS RATIONALE

An assurance level of EAL 4 with the augmentation AVA_VAN.5 is 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

UKİS v2.2.8H is the ***composite product*** consisting of Embedded Operating System and the Security IC. Some of the security features are provided mainly by Security IC and supported Embedded Operating system. Some of the security features are provided mainly by Embedded Operating system and supported by Security IC. A brief overview will be given for all Security Features. A detailed description also will be provided for the Security Features provided by Embedded Operating system. For the detailed information about security features provided by Security IC, Security IC ST [2]can be checked.

Security Features Directly Provided by IC

- SF.CO: Guarantee of Correct Operation

Security Features Provided mainly by IC and supported by Embedded OS

- SF.PM: Phase Management
- SF.PP: Physical Protection Against Physical Probing and Manipulation
- SF.LP: Logical Protection Against Data Leakage
- SF.CSUP: Cryptographic Support
- SF.TST: TSF Self-Test
- SF.RNG: Random Number Generation

Security Features Provided mainly by Embedded OS and supported by IC.

- SF.IA: Identification and Authentication
- SF.SMAC: Security Management and Access Control
- SF.SM: Secure Messaging

8.1 SF.CO: GUARANTEE OF CORRECT OPERATION

Guarantee of Correct Operation feature is provided by the Security IC For the detailed information Security IC ST can be checked. Covered SFRs are FPT_FLS.1and FRU_FLT.2

8.2 SF.PM: PHASE MANAGEMENT

Device phase management security feature is fulfilled by Security IC and Embedded Software. Covered SFRs are FAU_SAS.1, FMT_LIM.1, FMT_LIM.2.

UKİS v2.2.8H composite product may be given to Customer before personalization. TOE provides also phase management for the sub phases defined in 1.5.4.1. TSF restricts TOE functions according to

these phase management. Access policy and access functions of the sub phases are required the SFRs FDP_ACC.1/FUN and FDP_ACF.1/FUN.

8.3 SF.PP: PHYSICAL PROTECTION AGAINST PHYSICAL PROBING AND MANIPULATION

Protection against physical probing and manipulation security feature is fulfilled by the Security IC and Embedded OS together. For the detailed information on feature provided by the Security IC, Security IC ST can be checked.

Additionally, EOS controls the integrity of the user data and TSF data stored in the NVM, i.e. check the integrity of files and file headers upon selection of the corresponding file, test the integrity of the page and file information tables in the NVM and Authentication Status Data in the SRAM. In case of a EDC fault in the file headers, file information tables, page information tables and Authentication Status Data, EOS returns an error code and do not return the required data. In case of an integrity error in the Files, EOS returns a warning indicating that there could be a corruption in the file. EOS returns the required answer and warning together. FDP_SDI.2/EOS is covered with this functionality of the EOS.

When there is an authorized change in the access rights of any entity, header of the access-rules-file, where the access rules for this entity are defined is tested. In addition, upon taking a request to reach a file, integrity of current authentication status data are tested. Thus, FPT_TST.1 is covered by this feature.

Covered SFRs are: FDP_SDI.2/HW, FPT_PHP.3 (inherited from IC ST) FDP_SDI.2/EOS (defined in composite ST).

8.4 SF.LP: LOGICAL PROTECTION AGAINST DATA LEAKAGE

Logical protection against data leakage is taken by the Security IC and embedded OS. The features provided by the Security IC cover the FDP_ITT.1 “Basic Internal Transfer Protection”, FTP_ITT.1 “Basic Internal TSF Data Transfer Protection” and FDP_IFC.1 “Subset Information Flow Control. For more details, Platform ST document can be examined.

Additionally, during the usage of the functions of the platform, embedded OS takes extra actions in order not to leak data. These actions include precautions against timing attacks and DPA attacks. Thus, FPT_EMSEC.1 is covered by this feature.

8.5 SF.TST: TSF SELF-TEST

TOE Self-Test feature is provided by the Security IC. FPT_TST.2 is relevant for self-test property. This is covered by the Security IC, thus Security IC ST can be checked for detailed information. EOS calls the test function during initial startup and at the cases that the operating system requires.

8.6 SF.CSUP:CRYPTOGRAPHIC SUPPORT

The Hardware provides many cryptographic operations as detailed in HW ST. Composite TOE adds more cryptographic operations. They are RSA Key Pair Generation, SHA Calculation, Signature Verification and Generation, TDES decryption. The keys that represent confidential information are destructed after use. Covered SFRs are FCS_CKM.1/RSA_KeyPair, , FCS_COP.1/SHA, FCS_COP.1/AES, FCS_COP.1/TDES, FCS_COP.1/CMAC, 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-VER_9796, FCS_COP.1/DEC_PKCS#1 V1.5, FCS_COP.1/DEC_PKCS#1 V2.1 OAEP, FCS_COP.1/RSA_RAW.

8.7 SF.RNG: RANDOM NUMBER GENERATION

The Hardware provides true random number generation as detailed in HW ST. EOS uses hardware function to produce random numbers. With this property FCS_RND.1 is covered.

8.8 SF.IA: IDENTIFICATION AND AUTHENTICATION

The SF.IA includes the authentication mechanisms of Activation Agent Authentication, Initialization and Personalization Agent Authentication, Chip (Terminal) Authentication[179] 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 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_RND.1.

[179] Terminal authentication is provided by PIN authentication for SAM Configuration.

8.9 SF.SMAC: SECURITY MANAGEMENT AND ACCESS CONTROL

SMAC is the short form of 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. Even application may allow these rules to be modified during operational phase. UKİS v2.2.8H provides application owners 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 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. Preoperational roles are Activation Agent, Initialization Agent, and Personalization Agents. Besides supporting these roles, UKİS v2.2.8H allows application owner to define additional management roles that active in the operational phase.

8.10 SF.SM: SECURE MESSAGING

The TOE has SF.SM which allows the TOE communicates 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/AES, FCS_COP.1/CMAC, FCS_RND.1.

8.11 SECURITY FUNCTIONS RATIONALE

Table 13 shows the assignment of security functional requirements to TOE's security functionality.

Table 13. Coverage of SFRs by TOE Security Functions

Security Functional Requirement	SF.CO	SF.PM	SF.PP	SF.LP	SF.TST	SF.CSUP	SF.RNG	SF_I.A	SF.SMAC	SF.SM
FAU_SAS.1		✓						✓		
FCS_CKM.1/SM										✓
FCS_CKM.1/SM_PER-INI										✓
FCS_CKM.1/RSA_Key Pair						✓				
FCS_CKM.2/SM										✓
FCS_CKM.2/SM_PER-INI										✓
FCS_CKM.4						✓				
FCS_COP.1/SHA						✓				
FCS_COP.1/AES						✓				✓
FCS_COP.1/TDES						✓				
FCS_COP.1/CMAC						✓				✓
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-VER_9796								✓		
FCS_COP.1/DEC_PKCS#1 V1.5						✓				
FCS_COP.1/DEC_PKC						✓				

Security Functional Requirement	SF.CO	SF.PM	SF.PP	SF.LP	SF.TST	SF.CSUP	SF.RNG	SF_I.A	SF.SMAC	SF.SM
S#1 V2.1 OAEP										
FCS_COP.1/RSA_RAW								✓		
FCS_RND.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/HW			✓							
FDP_SDI.2/EOS			✓							
FIA_AFL.1/PIN								✓		
FIA_AFL.1/ACT								✓		
FIA_AFL.1/PER								✓		
FIA_AFL.1/INI								✓		
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									✓	

Security Functional Requirement	SF.CO	SF.PM	SF.PP	SF.LP	SF.TST	SF.CSUP	SF.RNG	SF_I.A	SF.SMAC	SF.SM
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			✓							
FPT_TST.2					✓					
FRU_FLT.2	✓							✓		

9 ABBREVIATIONS

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

EOS : Embedded Operating System

IC : Integrated Circuit

PP : Protection Profile

PTG2 : A class that defines the requirements for RNGs used in key generation, padding bit generation, etc. PTG2 is defined AIS31 [14]

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

UKİS : Ulusal Akıllı Kart İşletim Sistemi (National Smart Card Operating System)

10 BIBLIOGRAPHY

[1] Security IC Platform Protection Profile, Version 1.0, 15.06.2007, BSI-PP-0035

[2] National Smartcard IC UKTÜM-H v7.01 with DES-3DES v7.01, AES256 v7.01, RSA2048 v7.01 Libraries and with IC Dedicated Software Security Target, date: 2016-01-11, version 14

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

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

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

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

[7] ISO 1177 Information Processing Character Structure for start/stop and synchronous character oriented transmission

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

[9] ISO 7816-4Information 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

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

[12] AKISv2_YoneticiKullaniciKilavuzu.doc, v25, 30.05.2016

[13] AKISv2_KullaniciKilavuzu.doc, v17, 30.05.2016

[14] 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

[15] Common Criteria Protection Profile Machine Readable Travel Document with ICAO Application, Extended Access Control, Version 1.10, 25th March 2009