Document Administration

Recipient

Department	Name

For the attention of

Department	Name

Summary

The following document comprises the Security Target Lite for a TOE evaluated according to Common Criteria Version 2.3. The TOE being subject of the evaluation is the smartcard product

MICARDO V3.0 R1.0 HPC V1.0

from Sagem Orga GmbH. The IT product under consideration shall be evaluated according to CC EAL 4 augmented with a minimum strength level for the TOE security functions of SOF-high.

Keywords

Target of Evaluation (TOE), Common Criteria, IC, Dedicated Software, Smartcard Embedded Software, Basic Software, Application Software, Security Objectives, Assumptions, Threats, TOE Security Function (TSF), TOE Security Enforcing Function (SEF), Level of Assurance, Strength of Functions (SOF), Security Functional Requirement (SFR), Security Assurance Requirement (SAR), Security Function Policy (SFP)

Responsibility for updating the document

Dr. Susanne Pingel

Sagem ORGA GmbH

MICARDO V3.0 R1.0 HPC V1.0

ST-Lite

Document Id:	3MIC3EVAL.CSL.0002
Archive:	3
Product/project/subject:	MIC3EVAL (Micardo V3 Evaluierung)
Category of document:	CSL (ST-Lite)
Consecutive number:	0002
Version:	V1.01
Date:	23 May 2007
Author:	Dr. Susanne Pingel
Confidentiality:	

Checked report:	not applicable
Authorized (Date/Signature):	not applicable
Accepted (Date/Signature):	not applicable

OdsDok V2.00

Document Organisation

i Notation

None of the notations used in this document need extra explanation.

ii Official Documents and Standards

See Bibliography.

iii Revision History

Version	Type of change	Author / team
V1.00	First edition	Dr. Susanne Pingel
V1.01	Insertion of conformance claim	Jürgen Scheffer

Table of Contents

1 ST Introduction

1.1 ST Identification

This Security Target refers to the smartcard product “MICARDO V3.0 R1.0 HPC V1.0” (TOE) provided by Sagem Orga GmbH for a Common Criteria evaluation.

Title:	ST-Lite - MICARDO V3.0 R1.0 HPC V1.0
Document Category:	Security Target for a CC Evaluation (sanitized version of the complete Security Target)
Document ID:	Refer to Document Administration
Version:	Refer to Document Administration
Publisher:	Sagem Orga GmbH
Confidentiality:	public
TOE:	“MICARDO V3.0 R1.0 HPC V1.0” (Smartcard Product containing IC with Smartcard Embedded Software, including HPC Application and SIG Application, intended to be used within the German Health Care System)
Certification ID:	BSI-DSZ-CC-0392
IT Evaluation Scheme:	German CC Evaluation Scheme
Evaluation Body:	SRC Security Research & Consulting GmbH
Certification Body:	Bundesamt für Sicherheit in der Informationstechnik (BSI)

This Security Target has been built in conformance with Common Criteria V2.3.

1.2 ST Overview

Target of Evaluation (TOE) and subject of this Security Target (ST) is the smartcard product “MICARDO V3.0 R1.0 HPC V1.0” developed by Sagem Orga GmbH.

The TOE is realised as Smartcard Integrated Circuit (IC with contacts) with Smartcard Embedded Software, consisting of the MICARDO V3.0 Operating System platform and the dedicated Health Professional Card Application (HPC Application) and Signature Application (SIG Application) as intended to be used for the German Health Care System.

The TOE`s HPC Application and SIG Application are based on the MICARDO V3.0 Operating System platform providing a wide range of functionality which can be employed for different applications. The MICARDO V3.0 platform is designed as multifunctional platform for high security applications. The Operating System platform allows for an integration of a variety of applications, in particular in the following fields: Health Systems, ID Systems, Signature Applications with and without on-card signature key pair generation, Banking Systems, Loyalty Schemes.

In particular, the TOE´s platform and its technical functionality and inherently integrated security features are designed and developed under consideration of the following specifications, standards and requirements:

- Functional and security requirements defined in the specification /eHC1/ for the electronic Health Card (eHC) as employed within the German Health Care System
- Functional and security requirements defined in the specification /HPC-SMC1/ for the Health Professional Card (HPC) and the Security Module Card (SMC) as employed within the German Health System
- Functional and security requirements drawn from the EU Directive on electronic signatures /ECDir/, the German Signature Act /SigG01/, the German Signature Ordinance /SigV01/ and the catalogue of agreed cryptographic algorithms /ALGCAT/
- Requirements from the Protection Profiles /PP9911/, /PP-eHC/, /PP-HPC/, /PP-SMC/, /PP SSCD Type3/, /PP SSCD Type2/
- Technical requirements defined in /ISO 7816/, Parts 1, 2, 3, 4, 8, 9, 15

The TOE is intended to be used as Health Professional Card (HPC) within the German Health Care System. More detailed:

The HPC Application running on the underlying MICARDO V3.0 Operating System platform is implemented according to the requirements in /HPC-SMC1/ and /HPC-SMC2/. The HPC Application in the sense of this ST covers all elementary files at the MF-level, the DF.HPA, the DF.ESIGN, the DF.CIA.ESIGN as defined in /HPC-SMC2/ and further Sagem Orga specific files.

Furthermore, the TOE is intended to be used as Secure Signature-Creation Device (SSCD) for qualified electronic signatures in view of the European Directive 1999/93/EC on electronic signatures /ECDir/, the German Signature Act /SigG01/ and the German Signature Ordinance /SigV01/. The EU compliant SIG Application of the TOE is implemented according to the requirements in /HPC-SMC2/, chap. 8 and is explicitly designed for the generation of legally binding qualified electronic signatures as defined in /ECDir/, /SigG01/ and /SigV01/. The SIG Application in the sense of this ST covers the DF.QES as defined in /HPC-SMC2/ and all elementary files at the MF-level which are accessed by the DF.QES as well as further Sagem Orga specific files.

The functional and assurance requirements and components for SSCDs as defined in /ECDir/, Annex III are mapped to three different Protection Profiles, each of it corresponding to a dedicated type of SSCD. The Sagem Orga GmbH product is designed as SSCD of the so-called Type 3, i.e. as device with *oncard* - generation of the Signature-Creation Data / Signature-Verification Data (SCD/SVD key pair), the secure storage of the SCD/SVD key pair and the secure creation of electronic signatures by using the dedicated SCD key. Hence, the Security Target for the TOE resp. its SIG Application is based on the related Protection Profile /PP SSCD Type3/.

Note: The TOE explicitly does not implement a Signature-Creation Application (SCA).

Under technical view, the TOE comprises the following components:

- Integrated Circuit (IC) "Philips SmartMX P5CC036V1D Secure Smart Card Controller" provided by Philips Semiconductors GmbH
- Smartcard Embedded Software comprising the MICARDO V3.0 Operating System platform (designed as native implementation) and the dedicated HPC Application and SIG Application for the German Health Care System provided by Sagem Orga GmbH

The configuration of the TOE as HPC will be done by Sagem Orga GmbH prior to the delivery of the product. The TOE contains at its delivery unalterable identification information on the delivered configuration. Furthermore, the TOE provides authenticity information which allow for an authenticity proof of the product.

For the delivery of the TOE two different ways are established:

- The TOE is delivered to the customer in form of a complete initialised smartcard.
- Alternatively, the TOE is delivered to the customer in form of an initialised module. In this case, the smartcard finishing process (embedding of the delivered modules, final card tests) is task of the customer.

As the form of the delivery of the TOE does not concern the security features of the TOE in any way the TOE will be named in the following with “HPC” for short, independently of its form of delivery.

In order to be compliant with the requirements from the German Health Care System and the EU Directive on electronic signatures /ECDir/, the German Signature Act /SigG01/ and the German Signature Ordinance /SigV01/ the TOE will be evaluated according to CC EAL 4 augmented with a minimum strength level for the TOE security functions of SOF-high.

The CC evaluation and certification of the TOE against the present ST serves for the security certificate as it is required for the confirmation of the TOE as SSCD according to /ECDir/ and /SigG01/ (in German: Bestätigung nach EU Direktive bzw. Signaturgesetz). Furthermore, the security certificate for the TOE contributes as necessary and essential part to the so-called prescribed licence of the TOE as technical component HPC for usage within the German Health Care System. In addition, the CC evaluation and certification of the TOE implies the proof for compliance of the TOE´s HPC Application and SIG Application with the corresponding specifications /HPC-SMC1/ and /HPC-SMC2/ and their requirements.

The main objectives of this ST are

- to describe the TOE as a smartcard product intended to be used as HPC
- to define the limits of the TOE
- to describe the assumptions, threats and security objectives for the TOE
- to describe the security requirements for the TOE
- to define the TOE security functions

1.3 CC Conformance

The CC evaluation of the TOE is based upon

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, Version 2.3, August 2005 (/CC 2.3 Part1/)
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements, Version 2.3, August 2005 (/CC 2.3 Part2/)
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements, Version 2.3, August 2005 (/CC 2.3 Part3/)

For the evaluation the following methodology will be used:

- Common Methodology for Information Technology Security Evaluation, Part 2: Evaluation Methodology, Version 2.3, August 2005 (/CEM 2.3 Part2/)

This Security Target is written in accordance with the above mentioned Common Criteria Version 2.3 and claims the following CC conformances:

- Part 2 extended
- Part 3 conformant
- conformant to the Protection Profile “Health Professional Card (HPC) – Heilberufsausweis (HBA)” /PP-HPC/

Furthermore, the Security Target takes into account the contents of the Protection Profile /PP SSCD Type3/.

The Security Target for the TOE covers all essential aspects and contents of /PP SSCD Type3/. Only the following content related differences arise:

- Communication between the TOE and the external Signature-Creation Application (SCA):

  The establishment of a trusted channel resp. trusted path for the communication between the TOE and a SCA for a secure transmission of the data to be signed (DTBS) resp. of the verification authentication data (VAD) as required within /PP SSCD Type3/ is now specified as optional. In the case that a trusted channel resp. trusted path is not used the cardholder resp. signatory is responsible for establishing a trusted environment for the communication between the TOE and the SCA.

  This extension is necessary as TOEs with mandatory use of trusted channels and trusted paths can only be used by SCAs resp. interface devices supporting trusted channels and trusted paths and would be in particular unusable for any other type of interface devices.

- Personalisation Phase of the TOE´s dedicated SIG Application:

  Related to the personalisation of the TOE´s SIG Application additional aspects concerning assets, assumptions, threats, security policies, security objectives and security functional requirements are appropriately added.

The chosen level of assurance for the TOE is **EAL 4 augmented**. The augmentation includes the assurance components ADV_IMP.2, ATE_DPT.2, AVA_MSU.3 and AVA_VLA.4.

The minimum strength level for the TOE security functions is **SOF-high**.

In order to avoid redundancy and to minimize the evaluation efforts, the evaluation of the TOE will be conducted as a delta evaluation of the CC-certified smartcard product “MICARDO V3.0 R1.0” from Sagem Orga GmbH (Certification ID BSI-DSZ-CC-0390).

Hint: The CC evaluation of the smartcard product “MICARDO V3.0 R1.0” itself has been performed as a composite evaluation with re-usage of the evaluation results of the CC evaluation of the underlying semiconductor "Philips SmartMX P5CC036V1D Secure Smart Card Controller" provided by Philips Semiconductors GmbH. The IC incl. its IC Dedicated Software has been evaluated according to Common Criteria EAL 5 augmented with a minimum strength level for its security functions of SOF-high and is listed under the Certification ID BSI-DSZ-CC-0293. The evaluation of the IC is based on the Protection Profile BSI-PP-0002 (/BSI-PP-0002/), the evaluation of the composite product “MICARDO V3.0 R1.0” is oriented on the Protection Profiles /PP9911/, /PP-eHC/, /PP-HPC/, /PP-SMC/, /PP SSCD Type3/, /PP SSCD Type2/.

2 TOE Description

2.1 TOE Definition

2.1.1 Overview

The Target of Evaluation (TOE) is the smartcard product “MICARDO V3.0 R1.0 HPC V1.0” (HPC for short in the following) intended to be used as Health Professional Card (HPC) in the German Health Care System.

In technical view the HPC is realised as a proprietary operating system with an Application Layer directly set-up on this operating system layer.

The HPC is based on the microcontroller "Philips SmartMX P5CC036V1D Secure Smart Card Controller" provided by Philips Semiconductors GmbH. The IC incl. its Dedicated Software is evaluated according to Common Criteria EAL 5 augmented with a minimum strength level for its security functions of SOF-high (refer to Certification ID BSI-DSZ-CC-0293).

Roughly spoken, the TOE is composed from the following parts:

- Integrated Circuit (IC) with its proprietary IC Dedicated Software (TOE-IC)
- Smartcard Embedded Software (TOE-ES) consisting of
  - Basic Software (TOE-ES/BS)
  - Application Software (TOE-ES/AS)

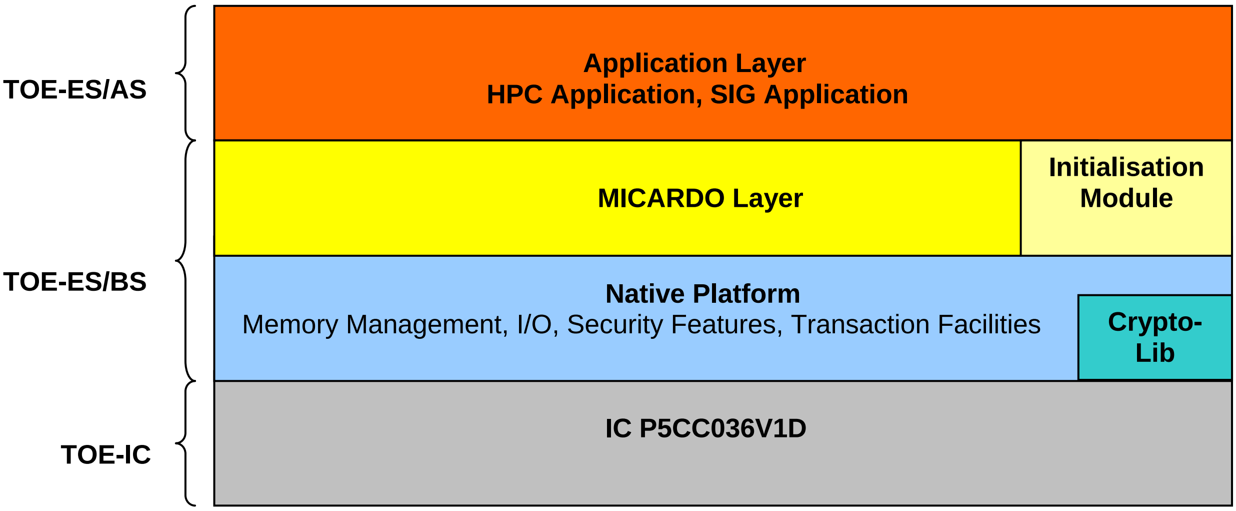
While the Basic Software consists of the MICARDO V3.0 Operating System platform of the TOE (realised as native implementation), the Application Software covers the Application Layer which is directly set-up on the MICARDO V3.0 Operating System platform and implements the specific HPC Application and SIG Application. The two pre-defined applications belonging to the TOE comprise own dedicated file and data systems with dedicated security structures, i.e. with application specific access rights for the access of subjects to objects and with application specific security mechanisms and PIN and key management. The design and implementation of the TOE´s dedicated HPC Application and SIG Application and their security structure follow the requirements in the specifications /HPC-SMC1/ and /HPC-SMC2/.

The HPC Application in the sense of this ST covers all elementary files at the MF-level, the DF.HPA, the DF.ESIGN, the DF.CIA.ESIGN as defined in /HPC-SMC2/ and further Sagem Orga specific files.

The SIG Application in the sense of this ST covers the DF.QES as defined in /HPC-SMC2/ and all elementary files at the MF-level which are accessed by the DF.QES as well as further Sagem Orga specific files.

Furthermore, the HPC itself offers the possibility to check its authenticity. For this purpose, the HPC contains the private part of a dedicated authentication key pair which depends on the configuration of the TOE and may be chosen customer specific (for more details see chap. 2.1.4.2).

The following figure shows the global architecture of the TOE and its components:



The different components of the TOE depicted in the figure above will be described more detailed in the following sections.

2.1.2 TOE Product Scope

The following table contains an overview of all deliverables associated to the TOE:

TOE component	Description / Additional Information	Type	Transfer Form
TOE-IC	Philips SmartMX P5CC036V1D Secure Smart Card Controller (incl. its IC Dedicated Software)	HW / SW	---
TOE-ES/BS	Smartcard Embedded Software / Part Basic Software (implemented in ROM/EEPROM of the microcontroller)	SW	---
TOE-ES/AS	Smartcard Embedded Software / Part Application Software (containing the HPC Application and SIG Application, implemented in the EEPROM of the microcontroller)	SW	---
Note: The TOE itself will be delivered as initialised smartcard or as initialised module.			
User Guide / User of the MICARDO platform	User guidance for the User of the MICARDO V3.0 R1.0 Operating System platform	DOC	Document in paper / electronic form
User Guide / User of the HPC Card	User guidance for the User of the HPC Card (in particular, HPC Application and SIG Application)	DOC	Document in paper / electronic form
User Guide / Personaliser of the HPC Card	User guidance for the Personaliser of the HPC Card (in particular, HPC Application and SIG Application)	DOC	Document in paper / electronic form

TOE component	Description / Additional Information	Type	Transfer Form
Identification Data Sheet of the HPC Card	Data Sheet with information on the actual identification data and configuration of the HPC Card delivered to the customer	DOC	Document in paper / electronic form
Aut-Key of the HPC Card	Public part of the authentication key pair relevant for the authenticity of the HPC Card Note: The card´s authentication key pair is generated by Sagem Orga GmbH and depends on the TOE´s configuration delivered to the customer. Furthermore, the key pair may be chosen customer specific.	KEY	Document in paper form / electronic file
Pers-Key of the HPC Card	Personalisation key relevant for the personalisation of the HPC Card Note: The card´s personalisation key pair is generated by Sagem Orga GmbH and depends on the TOE´s configuration delivered to the customer. Furthermore, the key may be chosen customer specific.	KEY	Document in paper form / electronic file

Note: Deliverables in paper form require a personal passing on or a procedure of at least the same security. For deliverables in electronic form an integrity and authenticity attribute will be attached.

2.1.3 Integrated Circuit (IC) with its Dedicated Software

Basis for the TOE´s Smartcard Embedded Software is the microcontroller “Philips SmartMX P5CC036V1D Secure Smart Card Controller”. The microcontroller and its Dedicated Software are developed and produced by Philips Semiconductors GmbH (within phase 2 and 3 of the smartcard product life-cycle, see chap. 2.2).

Detailed information on the IC Hardware, the IC Dedicated Software and the IC interfaces can be found in /ST-ICPhilips/.

2.1.4 Smartcard Embedded Software

The Smartcard Embedded Software of the TOE comprises the MICARDO V3.0 Operating System platform and applications running on this platform and is therefore divided into two parts with specific contents:

- Basic Software (MICARDO V3.0 Operating System platform)
- Application Software (Application Layer with dedicated HPC Application and SIG Application)

Each part of the Smartcard Embedded Software is designed and developed by Sagem Orga GmbH in phase 1 of the smartcard product life-cycle (see chap. 2.2). Embedding of the Smartcard Embedded Software into the TOE is performed in the later phases 3 and 5.

The main parts of the Basic Software are brought into the card by the IC manufacturer in form of the ROM mask and stored in the User-ROM of the IC (phase 3). The Application Software, and perhaps additional parts of the Basic Software, are located in the EEPROM area and are lateron loaded by specific initialisation routines of the TOE (phase 5). Hereby, the loading requires an encrypted and with a cryptographic checksum secured initialisation file. The necessary keys for securing the initialisation process are stored inside the IC during production time.

2.1.4.1 Basic Software

The Basic Software of the Smartcard Embedded Software comprises the MICARDO V3.0 Operating System platform of the TOE. Its main and security related parts are stored in the User-ROM of the underlying IC and are brought into the smartcard in form of the so-called ROM mask during the production process of the IC within phase 3 of the smartcard product life-cycle (see chap. 2.2).

The MICARDO V3.0 Operating System platform of the TOE is designed as proprietary software consisting of two layers. In detail, the integral parts of the TOE´s operating system consist of the MICARDO Layer and the Initialisation Module. Both are based on a Native Platform which serves as an abstraction layer towards the IC. On the other side, the MICARDO Layer and the Initialisation Module provide an interface between the operating system and the overlying Application Layer with the dedicated HPC Application and SIG Application.

The MICARDO Layer implements the executable code for the card commands and all general technical and security functionality of the MICARDO V3.0 Operating System platform as data objects and structures, file and object handling, security environments, security resp. cryptographic algorithms, key and PIN management, security states, access rules, secure messaging etc.

As mentioned, the Native Platform of the TOE´s operating system serves as an abstraction layer between the MICARDO Layer resp. the Initialisation Module and the IC. For this task, it provides essential operating system components and low level routines concerning memory management, I/O handling, transaction facilities, system management, security features and cryptographic mechanisms.

For the cryptographic features, the Native Platform integrates a specific module, the Crypto Library, which supports and implements the TOE´s core cryptographic functionality. In view of the Smartcard Embedded Software, the Crypto Library is accessible only via the Native Platform.

For the initialisation process of the TOE conducted within phase 5 of the smartcard product life-cycle (see chap. 2.2) the operating system of the TOE puts dedicated initialisation routines at disposal which are solely accessible during the initialisation phase and which are realised within the Initialisation Module. After the initialisation has been successfully completed these commands are no longer available. Furthermore, the functionality of deleting the complete initialisation file after the initialisation (deletion of the whole EEPROM area) is disabled for the TOE.

The Initialisation Module puts the following features at disposal:

- specific initialisation routines
- specific test routines for the EEPROM area

Loading of an initialisation file is only possible by use of the TOE´s specific initialisation routines. Hereby, the initialisation file to be loaded has to be secured before with an encryption and a cryptographic checksum, both done with dedicated keys of the TOE.

The test routines for the EEPROM area can be used for a check of the correct functioning of the memory.

Furthermore, the Initialisation Module manages the specific states of the TOE´s operating system according to specified and unalterable rules.

2.1.4.2 Application Software

The Application Software part of the TOE´s Smartcard Embedded Software comprises the Application Layer and is directly set-up on the TOE´s Basic Software. It consists of the TOE´s dedicated HPC Application and SIG Application which are implemented according to the requirements in /HPC-SMC1/ and /HPC-SMC2/.

The Application Software will be brought into the smartcard in cryptographically secured form during the initialisation process within phase 5 of the smartcard product life-cycle (see chap. 2.2). The initialisation process uses the specific initialisation routines of the TOE´s operating system, and the Application Software will be stored in the EEPROM area of the IC.

The HPC offers the capability to check its authenticity. For this purpose, the TOE contains the private part of a dedicated RSA authentication key pair over which by an internal authentication procedure the authenticity of the HPC can be proven. The authentication key pair depends on the Initialisation File (containing the Application Software to be initialised) and its configuration and may be chosen customer specific. The corresponding public part of the authentication key pair is delivered through a trusted way to the external world.

Furthermore, the TOE contains a data area for storing identification data of the TOE and its configuration. The data area will be filled in the framework of the initialisation of the TOE with a specific operating system command and can be read out with a further specific operating system command. Once the identification data have been written, there is afterwards no change possible.

2.1.4.3 TOE´s SIG Application

The TOE is a Secure Signature-Creation Device (SSCD Type 3) in view of the EU Directive /ECDir/ on electronic signatures.

The TOE as SSCD is configured software and hardware used to implement the Signature-Creation Data (SCD) and to guarantee for the secure usage of the SCD.

The TOE provides the following functions necessary for devices involved in creating qualified electronic signatures:

1. Generation of the SCD and the correspondent Signature-Verification Data (SVD)
2. Creation of qualified electronic signatures

a. after allowing for the data to be signed (DTBS) to be displayed correctly where the display function has to be provided by an appropriate environment

b. using appropriate hash functions that are, according to /ALGCAT/, agreed as suitable for qualified electronic signatures

c. after appropriate authentication of the signatory by the TOE

d. using appropriate cryptographic signature functions that employ appropriate cryptographic parameters agreed as suitable according to /ALGCAT/.

The TOE includes an automatic preceding destruction of the old SCD prior to the generation of the new SCD/SVD pair.

The TOE implements all IT security functionality which is necessary to ensure the secrecy of the SCD. To prevent the unauthorised usage of the SCD the TOE provides user authentication and access control. The user authenticates himself by supplying the verification authentication data (VAD) to the TOE which compares the VAD against the reference authentication data (RAD) securely stored inside the TOE. The TOE implements IT measures to support a trusted path to a trusted human interface device that can optionally be connected via a trusted channel with the TOE.

The TOE does not implement the Signature-Creation Application (SCA) which presents the data to be signed (DTBS) to the signatory and prepares the DTBS-representation the signatory wishes to sign for performing the cryptographic function of the signature. This ST assumes the SCA as environment of the TOE.

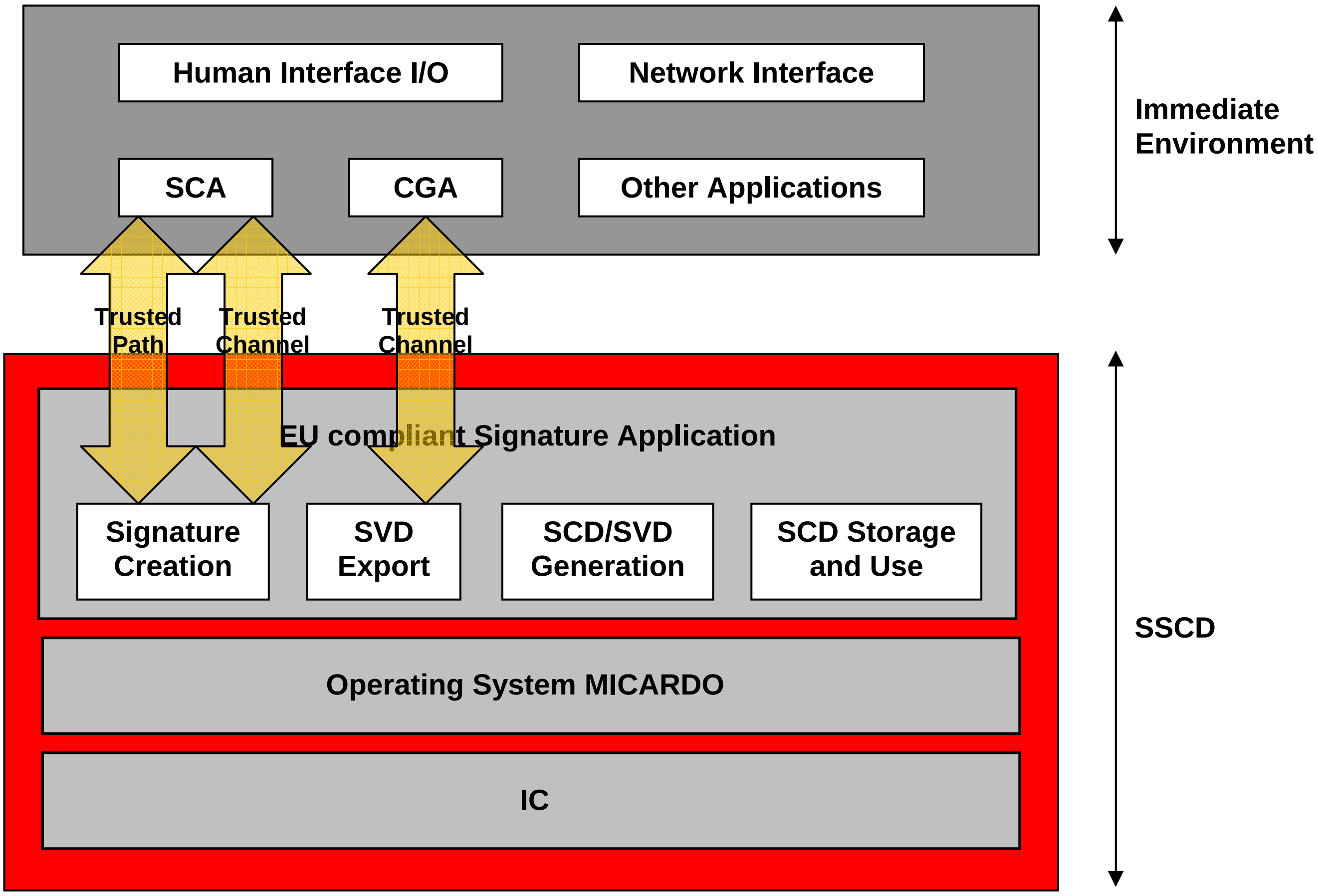
The TOE protects the SCD during the whole life-cycle as to be solely used in the signature-creation process by the legitimate signatory. The TOE as SSCD of Type 3 generates the signatory´s SCD oncard and serves for a secure storage of this data. The initialisation and personalisation of the TOE for the signatory´s use in the sense of the Protection Profile /PP SSCD Type3/ include:

1. Generation of the SCD/SVD pair
2. Personalisation for the signatory by means of the signatory's verification authentication data (VAD).

The SVD corresponding to the signatory's SCD will be included in the certificate of the signatory by the Certification-Service-Provider (CSP).

From the structural perspective, the TOE as SSCD comprises the underlying IC, the MICARDO V3.0 Operating System platform and the dedicated SIG Application with SCD/SVD generation, SCD storage and use, SVD export, and the signature-creation functionality. The SCA and the CGA (beside other applications within the German Health Care System) are part of the immediate environment of the TOE. They may communicate with the TOE over a trusted channel, a trusted path for the human interface provided by the SCA, respectively. In case a trusted channel or trusted path is not established with cryptographic means the TOE shall only be used within a Trusted Environment.

The following figure points the structural view of the TOE as SSCD and its integration into the external world out:


Human Interface I/O
Network Interface
SCA
CGA
Other Applications
Immediate Environment
Trusted Path
Trusted Channel
Trusted Channel
EU compliant Signature Application
Signature Creation
SVD Export
SCD/SVD Generation
SCD Storage and Use
Operating System MICARDO
IC
SSCD

2.2 TOE Life-Cycle

The smartcard product life-cycle of the TOE is decomposed into seven phases. In each of these phases different authorities with specific responsibilities and tasks are involved:

Phase		Description
Phase 1	**Smartcard Embedded Software Development**	The **Smartcard Embedded Software Developer (Sagem Orga GmbH)** is in charge of • the development of the Smartcard Embedded Software (Basic Software, Application Software) and • the specification of the IC initialisation and pre-personalisation requirements (though the actual data for the IC initialisation and pre-personalisation come from Phase 4, 5 resp. 6). The purpose of the Smartcard Embedded Software designed during phase 1 is to control and protect the TOE during phases 4 to 7 (product usage).The global security requirements of the TOE are such that it is mandatory during the development phase to anticipate the security threats of the other phases.
Phase 2	**IC Development**	The **IC Designer (Philips Semiconductors GmbH)** • designs the IC, • develops the IC Dedicated Software, • provides information, software or tools to the Smartcard Embedded Software Developer, and • receives the Smartcard Embedded Software (only Basic Software) from the developer through trusted delivery and verification procedures. From the IC design, IC Dedicated Software and Smartcard Embedded Software, the **IC Designer (Philips Semiconductors GmbH)** • constructs the smartcard IC database, necessary for the IC photomask fabrication.
Phase 3	**IC Manufacturing and Testing**	The **IC Manufacturer (Philips Semiconductors GmbH)** is responsible for • producing the IC through three main steps: - IC manufacturing, - IC testing, and - IC pre-personalisation. The **IC Mask Manufacturer (Philips Semiconductors GmbH)** • generates the masks for the IC manufacturing based upon an output from the smartcard IC database.
Phase 4	**IC Packaging and Testing**	The **IC Packaging Manufacturer (Sagem Orga GmbH)** is responsible for

		• the IC packaging (production of modules) and • testing.
Phase 5	**Smartcard Product Finishing Process**	The **Smartcard Product Manufacturer (Sagem Orga GmbH)** is responsible for • the initialisation of the TOE (in form of the initialisation of the modules of phase 4) and • its testing. The smartcard product finishing process comprises the embedding of the initialised modules for the TOE and the card production what is done alternatively by **Sagem Orga GmbH or by the customer.** Final card tests only aim at checking the quality of the card production, in particular concerning the bonding and implantation of the modules.
Phase 6	**Smartcard Personalisation**	The **Personaliser / Card Management System** is responsible for • the smartcard personalisation and • final tests. The personalisation of the smartcard includes the printing of the (card holder specific) visual readable data onto the physical smartcard, and the writing of (card holder specific) TOE User Data and TSF Data into the smartcard.
Phase 7	**Smartcard End-Usage**	The **Smartcard Issuer** is responsible for • the smartcard product delivery to the smartcard end-user (card holder), and the end of life process. The **authorized personalisation agents** (Card Management Systems) are allowed • to add data, modify or delete an HPC Application. The TOE is used as HPC by the smart card holder in the operational use phase.

Appropriate procedures for a secure delivery process of the TOE or parts of the TOE under construction from one development resp. production site to another site within the smartcard product life-cycle are established. This concerns any kind of delivery performed from phase 1 to 5, including:

- intermediate delivery of the TOE or parts of the TOE under construction within a phase,
- delivery of the TOE or parts of the TOE under construction from one phase to the next.

In particular, the delivery of the ROM mask and the EEPROM pre-personalisation data from Sagem Orga GmbH to Philips Semiconductors GmbH is done by using the dedicated secured delivery procedure specified by Philips Semiconductors GmbH following the so-called Philips Order Entry Form P5CC036V1D.

The IC manufacturer Philips Semiconductors GmbH delivers the IC with its IC Dedicated Software and the ROM mask supplied by Sagem Orga GmbH at the end of phase 3 in form of wafers according to /UG-ICPhilips/, chap. 2.1, Delivery Method 2, bullet point 1. The IC Dedicated Test Software stored in the Test-ROM is disabled before the delivery of the IC and cannot be used in the following phases.

The FabKey procedure described in /UG-ICPhilips/, chap. 2.1, Delivery Method 2, bullet point 2 is replaced by the following procedure which provides at least equivalent security: The TOE´s operating system puts in the non-initialised status the command “Verify ROM” at disposal, with which a SHA-1 hash value over the complete ROM and data freely chosen by the external world can be generated. Prior to the initialisation of the IC, the authenticity of the IC with its ROM mask will be proven by using the functionality “Verify ROM” and comparing the new generated hash value over the ROM data and the data freely chosen with a corresponding external reference value which is accessible only for Sagem Orga GmbH.

With regard to the smartcard product life-cycle of the TOE described above, the different development and production phases of the TOE with its IC incl. its IC Dedicated Software and with its Smartcard Embedded Software (Basic Software, Application Software) are part of the evaluation of the TOE. Two different ways for the delivery of the TOE are established:

- The TOE is delivered at the end of phase 5 in form of complete cards, i.e. after the initialisation process of the TOE has been successfully finished, final card tests have been successfully conducted and the card production has been fulfilled.
- Alternatively, the TOE is delivered in form of initialised and tested modules. In this case, the smartcard finishing process (embedding of the delivered modules, final card tests) is task of the customer.

2.3 TOE Environment

Considering the TOE and its life-cycle described above, four types of environments can be distinguished:

- development environment corresponding to phase 1 and 2,
- production environment corresponding to phase 3 to phase 5,
- personalisation environment corresponding to phase 6,
- end-user environment corresponding to phase 7.

2.3.1 Development Environment

Phase 1 - Smartcard Embedded Software Development

To assure security of the development process of the Smartcard Embedded Software, a secure development environment with appropriate personnel, organisational and technical security measures at Sagem Orga GmbH is established.

Only authorized and experienced personnel which understands the importance and the rigid implementation of the defined security procedures is involved in the development activities.

The development process comprises the specification, the design, the coding and the testing of the Smartcard Embedded Software. For design, implementation and test purposes secure computer systems preventing unauthorized access are used. For security reasons the coding and testing activities will be done independently of each other.

All sensitive documentation, data and material concerning the development process of the Smartcard Embedded Software are handled in an appropriately and sufficiently secure way. This concerns both the transfer as well as the storing of all related sensitive documents, data and material. Furthermore, all development activities run under a configuration control system which guarantees for an appropriate traceability and accountability.

The Smartcard Embedded Software of the developer, more precise the Basic Software part dedicated for the ROM of the IC, is delivered to the IC manufacturer through trusted delivery and verification procedures. The Application Software and additional parts of the Basic Software are delivered in form of a cryptographically secured initialisation file as well through trusted delivery and verification procedures to the initialisation centre.

Phase 2 – IC Development

During the design and layout process only people involved in the specific development project for the IC have access to sensitive data. Different people are responsible for the design data of the IC and for customer related data. The security measures installed at Philips Semiconductors GmbH ensure a secure computer system and provide appropriate equipment for the different development tasks.

2.3.2 Production Environment

Phase 3 - IC Manufacturing and Testing

The verified layout data are provided by the developers of Philips Semiconductors GmbH directly to the wafer fab. The wafer fab generates and forwards the layout data related to the relevant photomask to the IC mask manufacturer (Philips Semiconductors GmbH).

The photomask is generated off-site and verified against the design data of the development before usage. The accountability and traceability is ensured among the wafer fab and the photomask provider.

The production of the wafers includes two different steps regarding the production flow. In the first step the wafers are produced with the fixed mask independent of the customer. After that step the wafers are completed with the customer specific mask and the remaining mask. The computer tracking ensures the control of the complete process including the storage of the semifinished wafers.

The test process of every die is performed by a test centre of Philips Semiconductors GmbH.

Delivery processes between the involved Philips Semiconductors GmbH sites provide accountability and traceability of the produced wafers. The delivery of the ICs from Philips Semiconductors GmbH to Sagem Orga GmbH is made in form of wafers whereby non-functional ICs are marked on the wafer.

Phase 4 – IC Packaging and Testing

For security reasons the processes of IC packaging and testing at Sagem Orga GmbH are done in a secure environment with adequate personnel, organisational and technical security measures.

Only authorized and experienced personnel which understands the importance and the rigid implementation of the defined security procedures is involved in these activities.

All sensitive material and documentation concerning the production process of the TOE is handled in an appropriately and sufficiently secure way. This concerns both the transfer as well as the storing of all related sensitive material and documentation. All operations are done in such a way that appropriate traceability and accountability exist.

Phase 5 - Smartcard Product Finishing Process

To assure security of the initialisation process of the TOE, a secure environment with adequate personnel, organisational and technical security measures at Sagem Orga GmbH is established.

Only authorized and experienced personnel which understands the importance and the rigid implementation of the defined security procedures is involved in the initialisation and test activities.

The initialisation process of the TOE comprises the loading of the TOE´s Application Software and the remaining EEPROM-parts of the TOE´s Basic Software which have been specified, coded, tested and cryptographically secured in phase 1 of the product life-cycle. The TOE allows only the initialisation of the intended initialisation file with its Application Software and its parts of the Basic Software. For security reasons, secure systems within a separate network and preventing unauthorized access are used for the initialisation process.

If the TOE is delivered in form of initialised and tested modules, the smartcard finishing process, i.e. the embedding of the delivered modules and final card tests, is task of the customer.

Otherwise, the smartcard finishing process is part of the production process at Sagem Orga GmbH, and the TOE is delivered in form of complete (initialised) cards.

All sensitive documentation, data and material concerning the production processes of the TOE at Sagem Orga GmbH within phase 5 are handled in an appropriately and sufficiently secure way. This concerns both the transfer as well as the storing of all related sensitive documents, data and material. Furthermore, all operations run under a control system which supplies appropriate traceability and accountability.

At the end of this phase, the TOE is complete as smartcard and can be supplied for delivery to the personalisation centre for personalisation.

2.3.3 Personalisation Environment

Note: The phases from the TOE delivery at the end of phase 5 to phase 7 in the smartcard product life-cycle are not part of the TOE development and production process in the sense of this Security Target. Information about the phases 6 and 7 are just included to describe how the TOE is used after its development and production.

Phase 6 - Smartcard Personalisation

Central task for the personaliser is the personalisation of the initialised product, i.e the loading of card resp. card holder specific data into the dedicated HPC Application and SIG Application already existing on the initialised card.

The personalisation process and its security depends directly on the access rules which have been initialised. For instance, the already existing HPC Application and SIG Application on the card require for their personalisation a mutual authentication between the card and the personalisation unit with session key agreement and a following data transfer secured by Secure Messaging using the agreed session keys.

However, the establishment of a secure environment for the personalisation process with adequate personnel, organisational and technical security measures is in the responsibility of the personalisation centre itself. In particular, the personaliser is responsible for the set-up of a secure personalisation process and for taking into account the requirements and recommendations given in the TOE´s user guidance for the personaliser. The secure key management and handling of the cryptographic keys for securing the data transfer within the personalisation process (if applicable) and the secure handling of the personalisation data itself is task of the personalisation centre.

2.3.4 End-User Environment

Phase 7 – Smartcard End-usage

In the end-usage phase, the TOE is under control of the card holder, and the HPC Application and SIG Application with their file systems, objects and data residing on the card are used in their intended way in the German Health Care System. However, according to the card structure and the access rules set for the different objects, further card management activities (as e.g. deleting or adding applications, inserting further personalisation data) may be possible for authorised users.

2.4 TOE Intended Usage

Introducing information on the intended usage of the TOE is given within chap. 1.2. The present chapter will provide additional and more detailed information on the Operating System platform and on the HPC Application and SIG Application residing on the card at delivery time point.

In general, the MICARDO V3.0 Operating System platform is designed as multifunctional platform for high security applications. Therefore, the TOE provides an Operating System platform with a wide range of technical functionality and an adequate set of inherently integrated security features.

The MICARDO V3.0 Operating System platform supports the following services:

- Oncard-generation of RSA key pairs of high quality (with appropriate key lengths)
- Different signature schemes (based on RSA with appropriate key lengths and padding schemes)
- Different encryption schemes (based on DES and RSA with appropriate key lengths and padding schemes)
- Key derivation schemes
- PIN based authentication scheme
- Different key based authentication schemes (based on DES and RSA, with / without session key agreement)
- Hash value calculation
- Random number generation of high quality
- Calculation and verification of cryptographic checksums
- Verification of CV certificates
- Protection of the communication between the TOE and the external world against disclosure and manipulation (Secure Messaging)
- Protection of files and data by access control functionality
- Life-cycle state information related to the Operating System itself as well as to all objects processed by the card
- Confidentiality of cryptographic keys, PINs and further security critical data
- Integrity of cryptographic keys, PINs and further security critical data
- Confidentiality of operating system code and its internal data
- Integrity of operating system code and its internal data (self test functionality)
- Resistance of crypto functionality against Side Channel Analysis (SPA, DPA, TA, DFA)
- Card management functionality
- Channel management (with separation of channel related objects)

To support the security of the above mentioned features of the TOE, the MICARDO V3.0 Operating System platform provides appropriate countermeasures for resistance especially against the following attacks:

- Cloning of the product
- Unauthorised disclosure of confidential data (during generation, storage and processing)
- Unauthorised manipulation of data (during generation, storage and processing)
- Identity usurpation
- Forgery of data to be processed
- Derivation of information on the private key from the related public part for oncard-generated RSA key pairs
- Side Channel Attacks

The resistance of the TOE against such attack scenarios is reached by usage of appropriate security features already integrated in the underlying IC as well as by implementing additional appropriate software countermeasures.

The specific HPC Application of the TOE comprises a file system with objects, access rules and data according to the requirements in /HPC-SMC1/ and /HPC-SMC2/. The HPC and its dedicated HPC Application provide the following main security services:

- Authentication of the card holder by use of a PIN,
- Access control for the functions listed in the following
- Asymmetric card-to-card authentication between the HPC and the eHC without establishment of a trusted channel
- Asymmetric card-to-card authentication between the HPC and the SMC with establishment of a trusted channel
- Symmetric card-to-card authentication between the HPC and a security module with establishment of a trusted channel
- Document key decipherment
- Client-server authentication

Furthermore, the TOE is explicitly intended to be used as Secure Signature-Creation Device (SSCD) for qualified electronic signatures in view of the European Directive 1999/93/EC on electronic signatures /ECDir/, the German Signature Act /SigG01/ and the German Signature Ordinance /SigV01/. The EU compliant SIG Application of the TOE is implemented according to the requirements in /HPC-SMC2/ and is explicitly designed for the generation of legally binding qualified electronic signatures as defined in /ECDir/, /SigG01/ and /SigV01/.

The Sagem Orga product is designed as SSCD of the so-called Type 3, i.e. as device with *oncard* - generation of the Signature-Creation Data / Signature-Verification Data (SCD/SVD key pair), the secure storage and use of the SCD and the secure creation of electronic signatures using the dedicated SCD key.

The TOE´s SIG Application provides the following services:

- Oncard-generation of the SCD/SVD pair
- Signature-creation using the dedicated SCD
- Confidentiality of cryptographic keys, PINs and further security critical data
- Integrity of cryptographic keys, PINs and further security critical data
- Confidentiality of operating system code and its internal data
- Integrity of operating system code and its internal data
- Authentication of the signatory, administrator and other users
- Protection of the communication between the TOE and the external world against disclosure and manipulation
- Protection of files and data by access control

Additional detailed information on the intended usage of the TOE and its functionality is given within the chapters 1.2 and 2.1.2.

2.5 Application Note: Scope of SSCD ST Application

This ST is intended to be used for a CC evaluation of a Secure Signature-Creation Device (SSCD) in view of the requirements specified in the European Directive 1999/93/EC on electronic signatures /ECDir/, Annex III as well as to the requirements from the German Signature Act /SigG01/ and the German Signature Ordinance /SigV01/.

For the TOE´s dedicated Signature Application, this ST refers to qualified certificates as electronic attestation of the SVD corresponding to the signatory's SCD that is implemented by the TOE.

While the main application scenario of the SSCD will assume a qualified certificate to be used in combination with the SSCD, there still is a large benefit in the security when such a SSCD is applied in other areas and such application is encouraged. The SSCD may as well be applied to environments where the certificates expressed as 'qualified certificates' in the ST do not fulfil the requirements laid down in Annex I and Annex II of the Directive /ECDir/.

With this respect the notion of qualified certificates in the ST refers to the fact that when an instance of the SSCD is used with a qualified certificate, such use is from the technical point of view eligible for an electronic signature as referred to in Directive /ECDir/, article 5, paragraph 1. As a consequence, the standard /ECDir/ does not prevent a device itself from being regarded as a SSCD, even when used together with a non-qualified certificate.

3 TOE Security Environment

3.1 Assets

Assets are security–relevant elements to be directly protected by the TOE whereby assets have to be protected in terms of confidentiality and integrity. Confidentiality of assets is always intended with respect to untrusted users of the TOE and its security-critical components, whereas the integrity of assets is relevant for the correct operation of the TOE and its security-critical components.

The confidentiality of the code of the TOE is included in this ST for several reasons. First, the confidentiality is needed for the protection of intellectual/industrial property on security or effectiveness mechanisms. Second, though protection shall not rely exclusively on code confidentiality, disclosure of the code may weaken the security of the involved application. For instance, knowledge about the implementation of the operating system or the applications running on the operaing system may benefit an attacker. This also applies to internal data of the TOE, which may similarly provide leaks for further attacks.

3.1.1 General Assets of the TOE

For a detailed description of the general assets of the TOE (IC and MICARDO V3.0 Operating System platform) refer to /ST-MIC30/, chap. 3.1.

3.1.2 Specific Assets of the TOE´s HPC Application

For a detailed description of the TOE´s assets related to the TOE´s dedicated HPC Application refer to /PP-HPC/, chap. 3.1.

For the asset Card Authentication Private Key PrK.HPC.AUT the security attribute “key usage counter” is added. Refer to /PP-HPC/, chap. 10.

3.1.3 Specific Assets of the TOE´s SIG Application

For a detailed description of the TOE´s assets related to the TOE´s dedicated SIG Application refer to /PP SSCD Type3/, chap. 3.

Note: Biometric authentication is not supported by the TOE. Hence, “biometric data” and “biometric authentication references” are not applicable for the TOE.

The following asset concerning the personalisation of the TOE´s dedicated SIG Application is added:

SIG Application / Personalisation Data

Personalisation data related to the TOE´s dedicated SIG Application (integrity, authenticity and confidentiality of the personalisation data must be assured)

3.2 Assumptions

3.2.1 General Assumptions for the TOE

For a detailed description of the general assumptions for the TOE (IC and MICARDO V3.0 Operating System platform) refer to /ST-MIC30/, chap. 3.2.1.

3.2.2 Specific Assumptions for the TOE´s HPC Application

For a detailed description of the specific assumptions related to the TOE´s dedicated HPC Application refer to /PP-HPC/, chap. 3.4.

3.2.3 Specific Assumptions for the TOE´s SIG Application

For a detailed description of the specific assumptions related to the TOE´s dedicated SIG Application refer to /PP SSCD Type3/, chap. 3.1.

The following specific assumption concerning the personalisation of the TOE´s dedicated SIG Application is added:

A.SIG_PERS Security of the Personalisation Process for the SIG Application

The originator of the personalisation data and the personalisation center responsible for the personalisation of the TOE´s dedicated SIG Application handle the personalisation data in an adequate secure manner. This concerns especially the security data to be personalised as secret cryptographic keys and PINs. The storage of the personalisation data at the originator and at the personalisation center as well as the transfer of these data between the different sites is conducted with respect to data integrity, authenticity and confidentiality.

Furthermore, the personalisation center treats the data for securing the personalisation process, i.e. the personalisation keys suitably secure.

It is in the responsibility of the originator of the personalisation data to garantuee for a sufficient quality of the personalisation data, especially of the cryptographic material to be personalised. The preparation and securing of the personalisation data appropriate to the card´s structure and according to the TOE´s personalisation requirements is as well in the responsibility of the external world and is done with care.

3.3 Threats

The TOE is required to counter different type of attacks against its specific assets. A threat agent could try to threaten these assets either by functional attacks or by environmental manipulation, by specific hardware manipulation, by a combination of hardware and software manipulations or by any other type of attacks.

3.3.1 General Threats on the TOE

For the definition of the general threats related to the TOE (IC and MICARDO V3.0 Operating System platform) refer to /ST-MIC30/, chap. 3.3.1, 3.3.2 and 3.3.3.

3.3.2 Specific Threats on the TOE´s HPC Application

For a detailed description of the specific threats related to the TOE´s dedicated HPC Application refer to /PP-HPC/, chap. 3.3.

3.3.3 Specific Threats on the TOE´s SIG Application

For a detailed description of the specific threats related to the TOE´s dedicated SIG Application refer to /PP SSCD Type3/, chap. 3.2.

The following specific threats concerning the personalisation of the TOE´s dedicated SIG Application are added:

T.SIG_ PERS_Aut Authentication for Personalisation Process of SIG Application

A successful storage of personalisation data for the TOE´s dedicated SIG Application without authorisation (of the external world) would be a threat to the security of the TOE.

T.SIG_PERS_Data Modification or Disclosure of Personalisation Data of SIG Application

A successful modification or disclosure of personalisation data for the TOE´s dedicated SIG Application during the data import would be a threat to the security of the TOE.

3.4 Organisational Security Policies

3.4.1 General Organisational Security Policies for the TOE

For a detailed description of the general organisational security policies for the TOE (IC and MICARDO V3.0 Operating System platform) refer to /ST-MIC30/, chap. 3.4.

3.4.2 Specific Organisational Security Policies for the TOE´s HPC Application

For a detailed description of the organisational security policies related to the TOE´s dedicated HPC Application refer to /PP-HPC/, chap. 3.2 and 10. In particular, the organisational security policy OSP.Limit_Usage as defined in /PP-HPC/, chap. 10 is added.

3.4.3 Specific Organisational Security Policies for the TOE´s SIG Application

For a detailed description of the organisational security policies related to the TOE´s dedicated SIG Application refer to /PP SSCD Type3/, chap. 3.3.

4 Security Objectives

4.1 Security Objectives for the TOE

The security objectives for the TOE cover principally the following aspects:

- integrity and confidentiality of the TOE´s assets
- protection of the TOE and its associated documentation and environment during the development and production phases.

4.1.1 General Security Objectives for the TOE

For a detailed description of the general security objectives for the TOE (IC and MICARDO V3.0 Operating System platform) refer to /ST-MIC30/, chap. 4.1.1, 4.1.2 and 4.1.3.

4.1.2 Specific Security Objectives for the TOE´s HPC Application

For a detailed description of the specific security objectives related to the TOE´s dedicated HPC Application refer to /PP-HPC/, chap. 4.1, 4.2 and 10. In particular, the security objective OT.Limited_Key_Usage as defined in /PP-HPC/, chap. 10 is added.

4.1.3 Specific Security Objectives for the TOE´s SIG Application

For a detailed description of the specific security objectives related to the TOE´s dedicated SIG Application refer to /PP SSCD Type3/, chap. 4.1. All security objectives have been overtaken, except OT.DTBS_Integrity_TOE which has been re-defined according to the extension of the Protection Profile concerning the establishment of trusted channels / paths for the communication between the TOE and a SCA. Furthermore, a specific security objective related to the personalisation of the TOE´s dedicated SIG Application is added.

OT.DTBS_Integrity_TOE Verification of the DTBS-Representation Integrity

In the case that a trusted channel between the TOE and the SCA by cryptographic means is established the TOE shall verify that the DTBS-representation received from the SCA has not been altered in transit between the SCA and the TOE. The TOE itself shall ensure that the DTBS-representation is not altered by the TOE as well. Note, that this does not conflict with the signature-creation process where the DTBS itself could be hashed by the TOE.

OT.SIG_PERS Security of the Personalisation Process for the SIG Application

The TOE shall only load and store personalisation data for the TOE´s dedicated SIG Application after the authentication of the external world. The TOE shall only load and store unaltered and authentic personalisation data.

The TOE shall detect flaws during the personalisation process, i.e. during the loading of the personalisation data.

The TOE must be able to support secure communication protocols and procedures between the TOE and the personalisation device ensuring data integrity, authenticity and confidentiality.

4.2 Security Objectives for the Environment of the TOE

4.2.1 General Security Objectives for the Environment of the TOE

For a detailed description of the general security objectives related to the environment of the TOE (IC and MICARDO V3.0 Operating System platform) refer to /ST-MIC30/, chap. 4.2.1.

4.2.2 Specific Security Objectives for the Environment of the TOE´s HPC Application

For a detailed description of the specific security objectives related to the environment of the TOE´s dedicated HPC Application refer to /PP-HPC/, chap. 4.3.

4.2.3 Specific Security Objectives for the Environment of the TOE´s SIG Application

For a detailed description of the specific security objectives related to the environment of the TOE´s dedicated SIG Application refer to /PP SSCD Type3/, chap. 4.2. All security objectives have been taken over, with the following exceptions: OE.HI_VAD has been re-defined and the new security objective OE.Trusted_Environment has been added according to the extension of the Protection Profile concerning the establishment of trusted channels / paths for the communication between the TOE and a SCA. Furthermore, a specific security objective related to the personalisation of the TOE´s dedicated SIG Application is added.

OE.HI_VAD Protection of the VAD

If an external device provides the human interface for user authentication, this device or its environment will ensure confidentiality and integrity of the VAD as needed by the authentication method employed.

OE.Trusted_Environment Trusted Environment for SCA and TOE

In the case that a trusted channel resp. trusted path between the TOE and the SCA by cryptographic means is not established the environment for the TOE usage protects the confidentiality and integrity of the VAD as well as the integrity of the DTBS sent by the user via the SCA human interface to the TOE.

OE.SIG_PERS Security of the Personalisation Process for the SIG Application

The originator of the personalisation data and the personalisation center responsible for the personalisation of the TOE´s dedicated SIG Application handle the personalisation data in an adequate secure manner. This concerns especially the security data to be personalised as secret cryptographic keys and PINs. The storage of the personalisation data at the originator and at the personalisation center as well as the transfer of these data between the different sites is conducted with respect to data integrity, authenticity and confidentiality.

Furthermore, the personalisation center treats the data for securing the personalisation process, i.e. the personalisation keys suitably secure.

It is in the responsibility of the originator of the personalisation data to garantuee for a sufficient quality of the personalisation data, especially of the cryptographic material to be personalised. The preparation and securing of the personalisation data appropriate to the card´s structure and according to the TOE´s personalisation requirements is as well in the responsibility of the external world and is done with care.

5 IT Security Requirements

5.1 TOE Security Requirements

This section covers the subsections “TOE Security Functional Requirements” and ”TOE Security Assurance Requirements”.

5.1.1 TOE Security Functional Requirements

The TOE Security Functional Requirements (SFRs) define the functional requirements for the TOE using functional requirement components drawn directly from /CC 2.3 Part2/, functional requirement components of /CC 2.3 Part2/ with extension as well as self-defined functional requirement components. This chapter considers the SFRs concerning the IC (TOE-IC) as well as the SFRs concerning the Smartcard Embedded Software (TOE-ES).

Notes:

The SFRs for the TOE are listed in the following chapters within tables. Thereby, the tables contain in the left column the original definition of the respective SFR and its elements, dependencies, hierarchical information, management and audit functions. The right column supplies the iterations, selections, assignments and refinements chosen for the TOE.

Operations in the SFRs already carried out within the Protection Profiles are highlighted in bold face, further operations carried out in this ST are written in bold and italic face. Furthermore, extensions of the Protection Profile /PP SSCD Type3/ are marked by underlining the new text (refer to chap. 5.1.1.3).

In general, the SFRs can be categorized as follows: cryptographic support, user data protection, identification and authentication, security management, protection of the TSF, trusted paths/channels.

5.1.1.1 General TOE Security Functional Requirements for the TOE

For the definition of the general SFRs related to the TOE (IC and MICARDO V3.0 Operating System platform) refer to /ST-MIC30/, chap. 5.1.1.1 and 5.1.1.2.

5.1.1.2 TOE Security Functional Requirements for the TOE´s HPC Application

The following section gives a survey of the SFRs related to the TOE´s dedicated HPC Application as specified in the Protection Profile /PP-HPC/, chap. 6.1. The SFRs of the Protection Profile have been supplemented appropriately.

For the TOE´s dedicated HPC Application, the TOE maintains an SFP as defined as follows:

SFP HPC Access Control

Subjects:

- Card Management System (according to /PP-HPC/, chap. 3.1)
- Card Holder (according to /PP-HPC/, chap. 3.1)
- Terminal (according to /PP-HPC/, chap. 3.1)
- Secure Module Card (SMC) (according to /PP-HPC/, chap. 3.1)
- card management system (according to /PP-HPC/, chap. 10)

Security attributes for subjects:

- USER_GROUP (authorised user, non-authorised user)

Objects:

- Master File (MF), Dedicated Files (DF) and Elementary Files (EF)
- Health Professional related Data (EF.HPD)
- Global Data Object (EF.GDO)
- Card Authentication Private Keys (PrK.HPC.AUT)
- Client-Server Authentication Private Key (PrK.HP.AUT)
- Decipher Private Key (PrK.HP.ENC)
- Card Verifiable Certificates (CVC.HPC.AUT, CVC.HPC.TCE, CVC.CA_HPC.CS)
- X.509 certificates (C.HP.AUT, C.HP.ENC)
- display message in DF.ESIGN

Security attributes for objects:

- Access Rules
- Error Usage Counters and Usage Counters for Key and PIN objects

Operations (Access Modes):

- MICARDO V3.0 operating system commands

The HPC Access Control SFP controls the access of subjects to objects on the basis of security attributes. For a general description of the access rules handled by the TOE´s operating system refer to /ST-MIC30/, chap. 5.1.1.2. For a detailed description of the access rules explicitly set for the HPC Application and the handling of error usage counters and usage counters related to Key and PIN objects refer to /PP-HPC/, chap. 6.1 and 10.

FCS **Cryptographic Support**	
FCS_CKM **Cryptographic Key Management**	
FCS_CKM.1 **Cryptographic Key Generation**	PP HPC
FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *cryptographic key generation algorithm*] and specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: a) the management of changes to cryptographic key attributes. Examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption) Audit: a) Minimal: Success and failure of the activity b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	**FCS_CKM.1/ASYM** **FCS_CKM.1.1/ASYM** The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [**asymmetric card-to-card authentication with key agreement**] and specified cryptographic key sizes [**112 bit**] that meet the following: [- **/HPC-SMC1/, Annex E.3**].
	FCS_CKM.1/SYM **FCS_CKM.1.1/SYM** The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [**symmetric card-to-card authentication with key agreement**] and specified cryptographic key sizes [**112 bit**] that meet the following: [- **/HPC-SMC1/, Annex E.4**].
FCS_CKM.4 **Cryptographic Key Destruction**	PP HPC
FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction	**FCS_CKM.4** **FCS_CKM.4.1**

method [assignment: *cryptographic key destruction method*] that meets the following: [assignment: *list of standards*]. 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] - FMT_MSA.2 Secure security attributes Management: a) the management of changes to cryptographic key attributes. Examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption) Audit: a) Minimal: Success and failure of the activity b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [***erasure of a 3DES session key***] that meets the following: [***physical erasure of the key***]. **Application Note** The TOE shall destroy the Triple-DES encryption key (SMK.ENC) and the Retail-MAC message authentication keys (SMK.MAC) for secure messaging after reset or termination of secure messaging session or reaching fail secure state according to FPT_FLS.1.
FCS_COP **Cryptographic Operation**	
FCS_COP.1 **Cryptographic Operation**	PP HPC
FCS_COP.1.1 The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes	**FCS_COP.1/CSA** **FCS_COP.1.1/CSA** The TSF shall perform [**digital signature-creation**] in accordance with a specified cryptographic algorithm [**RSA**] and cryptographic key sizes [***1024, 1280, 1536, 1792 resp. 2048 bit modulus length***] that meet the following: [- **/PKCS1/, EMSA-PKCS1-v1_5**].

Management: --- Audit: a) Minimal: Success and failure, and the type of cryptographic operation b) Basic: Any applicable cryptographic mode(s) of operation, subject attributes and object attributes	
	FCS_COP.1/CCA_SIGN **FCS_COP.1.1/CCA_SIGN** The TSF shall perform [**digital signature-creation**] in accordance with a specified cryptographic algorithm [**RSA**] and cryptographic key sizes [**1024,** ***1280, 1536, 1792 resp. 2048*** **bit modulus length**] that meet the following: [- **/HPC-SMC2/, Annex E**].
	FCS_COP.1/RSA_DEC **FCS_COP.1.1/ RSA_DEC** The TSF shall perform [**decryption**] in accordance with a specified cryptographic algorithm [**RSA**] and cryptographic key sizes [***1024, 1280, 1536, 1792 resp. 2048 bit modulus length***] that meet the following: [- **/HPC-SMC2/, Annex E**].
	FCS_COP.1/CCA_VERIF **FCS_COP.1.1/ CCA_VERIF** The TSF shall perform [**digital signature-verification**] in accordance with a specified cryptographic algorithm [**RSA**] and cryptographic key sizes [**1024,** ***1280, 1536, 1792 resp. 2048*** **bit modulus length**] that meet the following: [- **/HPC-SMC2/, Annex E**].
	FCS_COP.1/TDES **FCS_COP.1.1/TDES** The TSF shall perform [**encryption and decryption**] in accordance with a specified cryptographic algorithm [**3DES in CBC mode**] and cryptographic key sizes [**112 bit**] that meet the following: [- **/FIPS 46-3/** - **/HPC-SMC1/, Annex C**].
	FCS_COP.1/MAC

	FCS_COP.1.1/MAC The TSF shall perform [**generation and verification of message authentication code**] in accordance with a specified cryptographic algorithm [**Retail MAC**] and cryptographic key sizes [**112 bit**] that meet the following: [- **/ANSI X9.19/** - **/HPC-SMC1/, Annex C**].
	FCS_COP.1/SHA **FCS_COP.1.1/SHA** The TSF shall perform [**hashing**] in accordance with a specified cryptographic algorithm [**SHA-1**] and cryptographic key sizes [**none**] that meet the following: [- **standard FIPS 180-2**].
FCS_RND **Generation of Random Numbers**	
FCS_RND.1 **Quality Metric for Random Numbers**	PP HPC
FCS_RND.1.1 The TSF shall provide a mechanism to generate random numbers that meet [assignment: a defined quality metric]. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: ---	**FCS_RND.1** **FCS_RND.1.1** The TSF shall provide a mechanism to generate random numbers that meet [***deterministic RNG of quality class K4***].

FDP **User Data Protection**	
FDP_ACC **Access Control Policy**	
FDP_ACC.2 **Complete Access Control**	PP HPC

FDP_ACC.2.1 The TSF shall enforce the [assignment: *access control SFP*] on [assignment: *list of subjects and objects*] and all operations among subjects and objects covered by the SFP. **FDP_ACC.2.2** The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP. <u>Hierarchical to:</u> FDP_ACC.1 <u>Dependencies:</u> - FDP_ACF.1 Security attribute based access control <u>Management:</u> --- <u>Audit:</u> ---	**FDP_ACC.2** **FDP_ACC.2.1** The TSF shall enforce the [**HPC Access Control SFP**] on [**1. the subjects** - **Card Management System** - **Card Holder** - **Terminal** - **Secure Module Card** - **card management system** **2. the objects** - **Master File (MF), Dedicated Files (DF) and Elementary Files (EF)** - **Health Professional related Data (EF.HPD)** - **Global Data Object (EF.GDO)** - **Card Authentication Private Keys (PrK.HPC.AUT)** - **Client-Server Authentication Private Key (PrK.HP.AUT)** - **Decipher Private Key (PrK.HP.ENC)** - **Card Verifiable Certificates (CVC.HPC.AUT, CVC.HPC.TCE, CVC.CA_HPC.CS)** - **X.509 certificates (C.HP.AUT, C.HP.ENC)** - **display message in DF.ESIGN**] and all operations among subjects and objects covered by the SFP. **FDP_ACC.2.2** The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.
FDP_ACF **Access Control Functions**	
FDP_ACF.1 **Security Attribute Based Access Control**	PP HPC
FDP_ACF.1.1 The TSF shall enforce the [assignment: *access control SFP*] to objects based on the following: [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*]. **FDP_ACF.1.2** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*].	**FDP_ACF.1** **FDP_ACF.1.1** The TSF shall enforce the [**HPC Access Control SFP**] to objects based on the following: [**authentication status of user**]. **FDP_ACF.1.2** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**1. the Card Management System is allowed** **(a) to load applications and to create Dedicated Files (DF) and Elementary Files (EF) in the Master File (MF) or Dedicated Files (DF) using**

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]. **FDP_ACF.1.4** The TSF shall explicitly deny access of subjects to objects based on the [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]. Hierarchical to: No other components Dependencies: - FDP_ACC.1 Subset access control - FMT_MSA.3 Static attribute initialisation Management: a) Managing the attributes used to make explicit access or denial based decisions Audit: a) Minimal: Successful requests to perform an operation on an object covered by the SFP b) Basic: All requests to perform an operation on an object covered by the SFP c) Detailed: The specific security attributes used in making an access check	**the Service_Asym_Mut_Auth_with_SM and PrK.HPC.AUT** **(b) to create the Health Professional related Data (EF.HPD)** **(c) to create and to write the Global Data Object (EF.GDO)** **(d) to create and to write Card Authentication Private Key (PrK.HPC.AUT)** **(e) to create and to write Client-Server Authentication Private Key (Pr.HP.AUT)** **(f) to create and to write Decipher Private Key (PrK.HP.ENC)** **(g) to create, to write and to read Card Verifiable Certificates (CVC.HPC.AUT, CVC.HPC.TCE, CVC.CA_HPC.CS)** **(h) to create, to write and to read X.509 certificates (C.HP.AUT, C.HP.ENC)** **(i) to create the display message in DF.ESIGN;** **2. the Card Holder is allowed** **(a) to read and to update the Health Professional related Data (EF.HPA)** **(b) to read Global Data Object (EF.GDO)** **(c) to read the Card Verifiable Certificates (CVC.HPC.AUT, CVC.HPC.TCE, CVC.CA-HPC.CS)** **(d) to read the X.509 certificates (C.HP.AUT, C.HP.ENC)** **(e) to update the display message (DM) in DF.ESIGN** **(f) to execute the card-to-card authentication Service_Asym_Mut_Auth_w/o_SM using PrK.HPC.AUT in security environment #1** **(g) to execute the card-to-card authentication Service_Asym_Mut_Auth_with_SM using PrK.HPC.AUT in security environment #2** **(h) to execute the document key decipherment Service_Data_Decryption using PrK.HP.ENC** **(i) to execute the client-server authentication Service_Client_Server_Auth using PrK.HP.AUT** **(j) to write the display message (DM) in DF.ESIGN;** **3. a Terminal is allowed** **(a) to read the Health Professional related Data (EF.HPD)** **(b) to read Global Data Object (EF.GDO)** **(c) to read the Card Verifiable Certificates (CVC.HPC.AUT, CVC.HPC.TCE, CVC.CA_HPC.CS)** **(d) to read the X.509 certificates (C.HP.AUT, C.HP.ENC)** **(e) to execute the card-to-card authentication Service_Asym_Mut_Auth_with_SM using PrK.HPC.AUT in security environment #2** **(f) to execute the card-to-card authentication Service_Sym_Mut_Auth_with_SM** **(g) to read the display message (DM) after estab-**

	lishing secure messaging]. **FDP_ACF.1.3** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**none**]. **FDP_ACF.1.4** The TSF shall explicitly deny access of subjects to objects based on the [**1. the Card Management System is not allowed** **(a) to execute the card-to-card authentication Service_Asym_Mut_Auth_w/o_SM with PrK.HPC.AUT** **(b) to execute the document key decipherment Service_Data_Decryption with PrK.HP.ENC** **(c) to execute the client-server authentication Service_Client_Server_Auth with PrK.HP.AUT** **(d) to read the display message (DM) in DF.ESIGN;** **2. the Terminal is not allowed** **(a) to execute the card-to-card authentication Service_Asym_Mut_Auth_w/o_SM with PrK.HPC.AUT** **(b) to execute the document key decipherment Service_Data_Decryption with PrK.HP.ENC** **(c) to execute the client-server authentication Service_Client_Server_Auth with PrK.HP.AUT** **(d) to read the display message (DM) in DF.ESIGN;** **3. no subject is allowed** **(a) to read any private key PrK.HPC.AUT, PrK.HP.AUT, and PrK.HP.ENC** **(b) to update the Card Verifiable Certificates (CVC.HPC.AUT, CVC.HPC.TCE, CVC.CA_HPC.CS)** **(c) to update the X.509 certificates (C.HP.AUT, C.HP.ENC)** **(d) to update the Global Data Object (EF.GDO)**].
FDP_RIP **Residual Information Protection**	
FDP_RIP.1 **Subset Residual Information Protection**	PP HPC
FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection: *allocation of the resource to*, *deallocation of the resource from*] the following objects: [assignment: *list of objects*].	**FDP_RIP.1** **FDP_RIP.1.1** The TSF shall ensure that any previous information content of a resource is made unavailable upon the [***deallocation of the resource from***] the following

Hierarchical to: No other components Dependencies: No dependencies Management: a) The choice of when to perform residual information protection (i.e. upon allocation or deallocation) could be made configurable within the TOE Audit: ---	objects: [***security relevant material (as secret and private cryptographic keys, PINs, PUCs, data in all files which are not freely accessible, ...)***].
FDP_SDI **Stored Data Integrity**	
FDP_SDI.2 **Stored Data Integrity Monitoring and Action**	PP HPC
FDP_SDI.2.1 The TSF shall monitor user data stored within the TSC for [assignment: *integrity errors*] on all objects, based on the following attributes: [assignment: *user data attributes*]. **FDP_SDI.2.2** Upon detection of a data integrity error, the TSF shall [assignment: *action to be taken*]. Hierarchical to: FDP_SDI.1 Dependencies: No dependencies Management: a) The actions to be taken upon the detection of an integrity error could be configurable Audit: a) Minimal: Successful attempts to check the integrity of user data, including an indication of the results of the check b) Basic: All attempts to check the integrity of user data, including an indication of the results of the check, if performed c) Detailed: The type of integrity error that occurred d) Detailed: The action taken upon detection of an integrity error	**FDP_SDI.2/Int-PersData** **FDP_SDI.2.1/Int-PersData** The TSF shall monitor user data ***and specific TSF data*** stored within the TSC for [**integrity errors**] on all objects, based on the following attributes: [***checksum secured persistently stored data***]. ***Application Note*** *The following data persistently stored by the TOE have the attribute „checksum secured persistently stored data“:* - *User / application data (e.g. in files of the card)* - *Keys (incl. attributes)* - *PINs / PUCs (incl. attributes)* - *File and object management information (as e.g. access rules, object life cycle states)* - *Card life cycle status information* ***Refinement*** *The check for integrity errors shall be done before usage resp. processing of the data. The checksum securing shall concern the data objects as well as the data values themselves.* **FDP_SDI.2.2/Int-PersData** Upon detection of a data integrity error, the TSF shall [- **prohibit the use of the altered data** - **inform the connected entity about integrity error**].
	FDP_SDI.2/Int-TempData **FDP_SDI.2.1/Int-TempData**

	The TSF shall monitor user data ***and specific TSF data*** stored within the TSC for [**integrity errors**] on all objects, based on the following attributes: [***checksum secured temporarily stored data***]. ***Application Note*** *The following data temporarily stored by the TOE have the attribute „checksum secured temporarily stored data“:* - *User / application data (as hash values, ...)* - *Keys (incl. attributes)* - *Card Context including different Channel Contexts (actual Security Environment, status information as the actual security status for Key and PIN based authentication, information on the availability of session keys, ...)* - *Input data for electronic signatures* ***Refinement*** *The check for integrity errors shall be done before usage resp. processing of the data. The checksum securing shall concern the data objects as well as the data values themselves.* **FDP_SDI.2.2/Int-TempData** Upon detection of a data integrity error, the TSF shall [- **prohibit the use of the altered data** - **inform the connected entity about integrity error**].
FDP_UCT **Inter-TSF User Data Confidentiality Transfer Protection**	
FDP_UCT.1 **Basic Data Exchange Integrity**	PP HPC
FDP_UCT.1.1 The TSF shall enforce the [assignment: *access control SFP(s) and/or information flow control SFP(s)*] to be able to [selection*: transmit, receive*] objects in a manner protected from unauthorised disclosure. Hierarchical to: No other components Dependencies: - [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] - [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] Management:	**FDP_UCT.1** **FDP_UCT.1.1** The TSF shall enforce the [**HPC Access Control SFP**] to be able to [**transmit and receive**] objects in a manner protected from unauthorised disclosure.

--- Audit: a) Minimal: The identity of any user or subject using the data exchange mechanisms b) Basic: The identity of any unauthorised user or subject attempting to use the data exchange mechanisms c) Basic: A reference to the names or other indexing information useful in identifying the user data that was transmitted or received. This could include security attributes associated with the information	
FDP_UIT **Inter-TSF User Data Integrity Transfer Protection**	
FDP_UIT.1 **Data Exchange Integrity**	PP HPC
FDP_UIT.1.1 The TSF shall enforce the [assignment: *access control SFP(s) and/or* i*nformation flow control SFP(s)*] to be able to [selection*: transmit, receive*] user data in a manner protected from [selection: *modification, deletion, insertion, replay*] errors. **FDP_UIT.1.2** The TSF shall be able to determine on receipt of user data, whether [selection: *modification*, *deletion, insertion, replay*] has occurred. Hierarchical to: No other components Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] - [FTP_ITC.1 Inter-TSF trusted channel or FTP_TRP.1 Trusted path] Management: --- Audit: a) Minimal: The identity of any user or subject using the data exchange mechanisms b) Basic: The identity of any user or subject attempting to use the user data exchange mechanisms, but who is unauthorised to do so c) Basic: A reference to the names or other indexing information useful in identifying the user data that was transmitted or received; this could include security attributes associated with the user data d) Basic: Any identified attempts to block transmission of user data e) Detailed: The types and/or effects of any detected	**FDP UIT.1** **FDP_UIT.1.1** The TSF shall enforce the [**HPC Access Control SFP**] to be able to [**transmit and receive**] user data in a manner protected from [**modification, deletion, insertion and replay**] errors. **FDP_UIT.1.2** The TSF shall be able to determine on receipt of user data, whether [**modification, deletion, insertion and replay**] has occurred.

modifications of transmitted user data	

FIA **Identification and Authentication**	
FIA_AFL **Authentication Failures**	
FIA_AFL.1 **Authentication Failure Handling**	PP HPC
FIA_AFL.1.1 The TSF shall detect when [selection: [assignment: *positive integer number*], “*an administrator configurable positive integer within* [assignment: *range of acceptable values*]“] unsuccessful authentication attempts occur related to [assignment: *list of authentication events*]. **FIA_AFL.1.2** When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [assignment: *list of actions*]. Hierarchical to: No other components Dependencies: - FIA_UAU.1 Timing of authentication Management: a) management of the threshold for unsuccessful authentication attempts b) management of actions to be taken in the event of an authentication failure Audit: a) Minimal: the reaching of the threshold for the unsuccessful authentication attempts and the actions (e.g. disabling of a terminal) taken and the subsequent, if appropriate, restoration to the normal state (e.g. re-enabling of a terminal)	**FIA_AFL.1/HPC-PIN** **FIA_AFL.1.1/HPC-PIN** The TSF shall detect when [***3***] unsuccessful authentication attempts occur related to [**consecutive failed human user authentication for the health care application**]. **FIA_AFL.1.2/HPC-PIN** When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [- **block the PIN for authentication until successful unblock with resetting code**].
	FIA_AFL.1/C2C **FIA_AFL.1.1/C2C** The TSF shall detect when ***[“an administrator configurable positive integer within [0 and 65334]“***] **successful or** unsuccessful authentication attempts occur related to [**key usage of the PrK.HPC.AUT**]. **FIA_AFL.1.2/C2C** When the defined number of **successful or** unsuccessful authentication attempts has been met or sur-

	passed, the TSF shall [- ***warn the entity connected*** - ***not set the actual security state for the key PrK.HPC.AUT*** - **block the key PrK.HPC.AUT resp. the authentication mechanism for this key such that any subsequent authentication attempt with this key will fail** - ***be able to indicate to subsequent users the reason for the blocking of the key PrK.HPC.AUT***].
FIA_ATD **User Attribute Definition**	
FIA_ATD.1 **User Attribute Definition**	PP HPC
FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: [assignment: *list of security attributes*]. Hierarchical to: No other components Dependencies: No dependencies Management: a) if so indicated in the assignment, the authorised administrator might be able to define additional security attributes for users Audit: ---	**FIA_ATD.1** **FIA_ATD.1.1** The TSF shall maintain the following list of security attributes belonging to individual users: [**identity and role**].
FIA_UAU **User Authentication**	
FIA_UAU.1 **Timing of Authentication**	PP HPC
FIA_UAU.1.1 The TSF shall allow [assignment: *list of TSF mediated actions*] 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. Hierarchical to: No other components	**FIA_UAU.1** **FIA_UAU.1.1** The TSF shall allow [**reading the ATR, reading data with access condition ALWAYS, identification by providing the users certificate, execution of the command INTERNAL AUTHENTICATE with PrK.HPC.AUT, algorithm '1F' in SE#2, *execution of commands allowed without preceding successful authentication due to the access rules which are set***] on behalf of the user to be performed before the user is authenticated.

<u>Dependencies:</u> - FIA_UID.1 Timing of identification <u>Management:</u> a) management of the authentication data by an administrator b) management of the authentication data by the associated user c) managing the list of actions that can be taken before the user is authenticated <u>Audit:</u> a) Minimal: Unsuccessful use of the authentication mechanism b) Basic: All use of the authentication mechanism c) Detailed: All TSF mediated actions performed before authentication of the user	**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**	PP HPC
FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to [assignment: *identified authentication mechanism(*s)]. <u>Hierarchical to:</u> No other components <u>Dependencies:</u> No dependencies <u>Management:</u> --- <u>Audit:</u> a) Minimal: Attempts to reuse authentication data	**FIA_UAU.4** **FIA_UAU.4.1** The TSF shall prevent reuse of authentication data related to [**Card-to-Card Authentication Mechanism** **(1) execution of the command EXTERNAL AUTHENTICATE as part of the Service_Asym_Mut_Auth_w/o_SM with PrK.HPC.AUT in SE#1,** **(2) execution of the command EXTERNAL AUTHENTICATE as part of the Service_Asym_Mut_Auth_with_SM with PrK.HPC.AUT in SE#2,** **(3) execution of the command EXTERNAL AUTHENTICATE as part of the Service_Sym_Mut_Auth_with_SM,** **(4) execution of the command EXTERNAL AUTHENTICATE**].
FIA_UAU.6 **Re-Authenticating**	PP HPC
FIA_UAU.6.1 The TSF shall re-authenticate the user under the conditions [assignment: *list of conditions under which re-authentication is required*]. <u>Hierarchical to:</u> No other components <u>Dependencies:</u> No dependencies <u>Management:</u>	**FIA_UAU.6** **FIA_UAU.6.1** The TSF shall re-authenticate the user under the conditions [**successful established secure messaging**].

a) if an authorised administrator could request re-authentication, the management includes a re-authentication request. Audit: a) Minimal: Failure of reauthentication b) Basic: All reauthentication attempts	
FIA_UID **User Identification**	
FIA_UID.1 **Timing of Identification**	PP HPC
FIA_UID.1.1 The TSF shall allow [assignment: *list of TSF-mediated actions*] 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. Hierarchical to: No other components Dependencies: No dependencies Management: a) the management of the user identities b) if an authorised administrator can change the actions allowed before identification, the managing of the action lists Audit: a) Minimal: Unsuccessful use of the user identification mechanism, including the user identity provided b) Basic: All use of the user identification mechanism, including the user identity provided	**FIA_UID.1** **FIA_UID.1.1** The TSF shall allow [**reading the ATR, reading data with access condition ALWAYS, *execution of commands allowed without preceding successful authentication due to the access rules which are set***] 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.

FMT **Security Management**	
FMT_LIM **Limited capabilities and availability**	
FMT_LIM.1 **Limited capabilities**	PP HPC
FMT_LIM.1.1 The TSF shall be designed in a manner that limits	**FMT_LIM.1**

their capabilities so that in conjunction with “Limited availability (FMT_LIM.2)” the following policy is enforced [assignment: *Limited capability and availability policy*]. Hierarchical to: No other components Dependencies: - FMT_LIM.2 Limited availability Management: --- Audit: ---	**FMT_LIM.1.1** The TSF shall be designed in a manner that limits their capabilities so that in conjunction with “Limited availability (FMT_LIM.2)” the following policy is enforced [**Deploying Test Features after TOE Delivery does not allow User Data to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks**].
FMT_LIM.2 **Limited availability**	PP HPC
FMT_LIM.2.1 The TSF shall be designed in a manner that limits their availability so that in conjunction with “Limited capabilities (FMT_LIM.1)” the following policy is enforced [assignment: *Limited capability and availability policy*]. Hierarchical to: No other components Dependencies: - FMT_LIM.1 Limited capability Management: --- Audit: ---	**FMT_LIM.2** **FMT_LIM.2.1** The TSF shall be designed in a manner that limits their availability so that in conjunction with “Limited capabilities (FMT_LIM.1)” the following policy is enforced [**Deploying Test Features after TOE Delivery does not allow User Data to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks**].
FMT_MTD **Management of TSF Data**	
FMT_MTD.1 **Management of TSF Data**	PP HPC
FMT_MTD.1.1 The TSF shall restrict the ability to [selection: *change_default, query, modify, delete, clear*, [assignment: *other operations*]] the [assignment: *list of TSF data*] to [assignment: *the authorised identified roles*]. Hierarchical to: No other components Dependencies: - FMT_SMF.1 Specification of management func-	**FMT_MTD.1/INI** **FMT_MTD.1.1/INI** The TSF shall restrict the ability to [**write**] the [**initialisation data and pre-personalisation data**] to [**the Manufacturer**].

tions - FMT_SMR.1 Security roles <u>Management:</u> a) managing the group of roles that can interact with the TSF data <u>Audit:</u> a) Basic: All modifications to the values of TSF data	
	FMT_MTD.1/RAD_WR **FMT_MTD.1.1/RAD_WR** The TSF shall restrict the ability to [**write**] the [**user authentication reference data, public keys of the root for CV certificate verification**] to [**the Card Management System**].
	FMT_MTD.1/RAD_MOD **FMT_MTD.1.1/RAD_MOD** The TSF shall restrict the ability to [**modify**] the [**public keys of the root for CV certificate verification**] to [**Card Management System**].
	FMT_MTD.1/PIN **FMT_MTD.1.1/PIN** The TSF shall restrict the ability to [**modify and unblock**] the [**PIN**] to [**the Card Holder**].
	FMT_MTD.1/RAD_CH **FMT_MTD.1.1/RAD_CH** The TSF shall restrict the ability to [**read**] the [**PIN and PUC**] to [**none**].
	FMT_MTD.1/C2C **FMT_MTD.1.1/C2C** The TSF shall restrict the ability to [**reset to *default value***] the [**key usage counter of PrK.HPC.AUT**] to [**card management system**].
FMT_SMF **Specification of Management Functions**	
FMT_SMF.1 **Specification of Management Functions**	PP HPC
FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions: [assignment: *list of security management functions to be provided by the TSF*]. <u>Hierarchical to:</u>	**FMT_SMF.1** **FMT_SMF.1.1** The TSF shall be capable of performing the following security management functions: [**initialisation, personalisation, card management, modification of the PIN, unblocking the PrK.HPC.AUT**].

No other components Dependencies: No dependencies Management: --- Audit: a) Minimal: Use of the management functions.	
FMT_SMR **Security Management Roles**	
FMT_SMR.1 **Security Roles**	PP HPC
FMT_SMR.1.1 The TSF shall maintain the roles [assignment: *the authorised identified roles*]. **FMT_SMR.1.2** The TSF shall be able to associate users with roles. Hierarchical to: No other components Dependencies: - FIA_UID.1 Timing of identification Management: a) managing the group of users that are part of a role Audit: a) Minimal: modifications to the group of users that are part of a role b) Detailed: every use of the rights of a role	**FMT_SMR.1** **FMT_SMR.1.1** The TSF shall maintain the roles [**Manufacturer, Card Management System, Card Holder, Terminals and card management system, SMC**]. **FMT_SMR.1.2** The TSF shall be able to associate users with roles.

FPT **Protection of the TSF**	
FPT_EMSEC **TOE Emanation**	
FPT_EMSEC.1 **TOE Emanation**	PP HPC
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** **FPT_EMSEC.1.1** The TOE shall not emit [***information on IC power consumption, information on command execution time, information on electromagnetic emanations***]

FPT_EMSEC.1.2 The TSF shall ensure [assignment: type of users] are unable to use the following interface [assignment: type of connection] to gain access to [assignment: list of types of TSF data] and [assignment: list of types of user data]. <u>Hierarchical to:</u> No other components <u>Dependencies:</u> No dependencies <u>Management:</u> --- <u>Audit:</u> ---	in excess of [***non useful information***] enabling access to [***security critical data as* PIN and PUC**] and [***security critical data as cryptographic keys, in particular* Card Authentication Private Keys, Client-Server Authentication Private Key, Document Cipher Key Decipher Key, secure messaging keys**]. **FPT_EMSEC.1.2** The TSF shall ensure [**any user**] are unable to use the following interface [**smart card circuit contacts**] to gain access to [***security critical data as* PIN and PUC**] and [***security critical data as cryptographic keys, in particular* Card Authentication Private Keys, Client-Server Authentication Private Key, Document Cipher Key Decipher Key, secure messaging keys**]. **Application Note** The TOE shall prevent attacks against secret data where the attack is based on external observable physical phenomena of the TOE. Such attacks may be observable at the interfaces of the TOE or may origin from internal operation of the TOE or may origin by an attacker that varies the physical environment under which the TOE operates. The set of measurable physical phenomena is influenced by the technology employed to implement the TOE. Examples of measurable phenomena are variations in the power consumption, the timing of transitions of internal states, electromagnetic radiation due to internal operation, radio emission. Due to the heterogeneous nature of the technologies that may cause such emanations, evaluation against state-of-the-art attacks applicable to the technologies employed by the TOE is assumed. Examples of such attacks are, but are not limited to, evaluation of TOE’s electromagnetic radiation**,** simple power analysis (SPA), differential power analysis (DPA), timing attacks, etc.
FPT_FLS **Fail Secure**	
FPT_FLS.1 **Failure with Preservation of Secure State**	PP HPC
FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [assignment: *list of types of failures in the TSF*]. <u>Hierarchical to:</u> No other components <u>Dependencies:</u> - ADV_SPM.1 Informal TOE security policy model	**FPT_FLS.1** **FPT_FLS.1.1** The TSF shall preserve a secure state when the following types of failures occur: [- **Exposure to operating conditions where therefore a malfunction could occur** - **Failure detected by TSF according to FPT_TST.1**

Management: --- Audit: a) Basic: Failure of the TSF	].
FPT_PHP **Physical Protection**	
FPT_PHP.3 **Resistance to Physical Attack**	PP HPC
FPT_PHP.3.1 The TSF shall resist [assignment: *physical tampering scenarios*] to the [assignment: *list of TSF devices / elements*] by responding automatically such that the TSP is not violated. Hierarchical to: No other components Dependencies: No dependencies Management: a) management of the automatic responses to physical tampering Audit: ---	**FPT_PHP.3** **FPT_PHP.3.1** The TSF shall resist [**physical manipulation and physical probing**] to the [**TSF**] by responding automatically such that the TSP is not violated. **Application Note** The TOE will implement appropriate measures to continuously counter physical manipulation and physical probing. Due to the nature of these attacks (especially manipulation) the TOE can by no means detect attacks on all of its elements. Therefore, permanent protection against these attacks is required ensuring that the TSP could not be violated at any time. 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_RVM **Reference Mediation**	
FPT_RVM.1 **Non-Bypassability of the TSP**	PP HPC
FPT_RVM.1.1 The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: ---	**FPT_RVM.1** **FPT_RVM.1.1** The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

FPT_SEP **Domain Separation**	
FPT_SEP.1 **TSF Domain Separation**	PP HPC
FPT_SEP.1.1 The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects. **FPT_SEP.1.2** The TSF shall enforce separation between the security domains of subjects in the TSC. <u>Hierarchical to:</u> No other components <u>Dependencies:</u> No dependencies <u>Management:</u> --- <u>Audit:</u> ---	**FPT_SEP.1** **FPT_SEP.1.1** The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects. **FPT_SEP.1.2** The TSF shall enforce separation between the security domains of subjects in the TSC. **Application Note** Those parts of the TOE which support the security functional requirements “TSF testing (FPT_TST.1)” and “Failure with preservation of secure state (FPT_FLS.1)” shall be protected from interference of the other security enforcing parts of the HPC chip Embedded Software. The security enforcing functions and health application data shall be separated in a way preventing any interference.
FPT_TST **TSF Self Test**	
FPT_TST.1 **TSF Testing**	PP HPC
FPT_TST.1.1 The TSF shall run a suite of self tests [selection: *during initial start-up, periodically during normal operation, at the request of the authorised user, at the conditions* [assignment: *conditions under which self test should occur*]] to demonstrate the correct operation of [selection: [assignment: *parts of TSF*], *the TSF*]. **FPT_TST.1.2** The TSF shall provide authorised users with the capability to verify the integrity of [selection: [assignment: *parts of TSF data*], *TSF data*]. **FPT_TST.1.3** The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code. <u>Hierarchical to:</u> No other components <u>Dependencies:</u> - FPT_AMT.1 Abstract machine testing <u>Management:</u>	**FPT_TST.1** **FPT_TST.1.1** The TSF shall run a suite of self tests [***during initial start-up, periodically during normal operation***] to demonstrate the correct operation of [***the TSF***]. ***Note*** *During initial start-up means before code execution.* ***Refinement*** *The TOE's self tests shall include the verification of the integrity of any software code (incl. patches) stored outside of the ROM. Upon detection of a self test error the TSF shall warn the entity connected. After OS testing is completed, all testing-specific commands and actions shall be disabled or removed. It shall not be possible to override these controls and restore them for use. Command associated exclusively with one life cycle state shall never be accessed during another state.* **FPT_TST.1.2** The TSF shall provide authorised users with the capability to verify the integrity of [***TSF data***].

a) management of the conditions under which TSF self testing occurs, such as during initial start-up, regular interval, or under specified conditions b) management of the time interval if appropriate Audit: a) Basic: Execution of the TSF self tests and the results of the tests	***Refinement*** *In this framework, the OS (i.e. the Smartcard Embedded Software of the TOE (TOE-ES)) itself is understood as „authorised user“.* **FPT_TST.1.3** The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code. ***Refinement*** *The integrity check over the executable code stored outside the ROM area is covered by FPT_TST.1.1 and the related refinement.* *The requirement for checking the integrity of the ROM-code shall concern only the production phase, more precise the initialisation phase of the TOE´s life-cycle. Prior to the initialisation of the TOE, the ROM-code of the TOE shall be verifiable by authorised users as the OS developer. The integrity of the ROM-code shall be provable only during the initialisation process.*

FTP **Trusted Path/Channels**	
FTP_ITC **Inter-TSF Trusted Channel**	
FTP_ITC.1 **Inter-TSF Trusted Channel**	PP HPC
FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. **FTP_ITC.1.2** The TSF shall permit [selection: *the TSF, the remote trusted IT product*] to initiate communication via the trusted channel. **FTP_ITC.1.3** The TSF shall initiate communication via the trusted channel for [assignment: *list of functions for which a trusted channel is required*]. Hierarchical to: No other components	**FTP_ITC.1** **FTP_ITC.1.1** The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. **FTP_ITC.1.2** The TSF shall permit [**the remote trusted IT product**] to initiate communication via the trusted channel. **FTP_ITC.1.3** The TSF shall initiate communication via the trusted channel for [**commands and responses after successful card-to-card authentication with algorithm ´1F´**].

Dependencies: No dependencies Management: a) Configuring the actions that require trusted channel, if supported Audit: a) Minimal: Failure of the trusted channel functions b) Minimal: Identification of the initiator and target of failed trusted channel functions c) Basic: All attempted uses of the trusted channel functions d) Basic: Identification of the initiator and target of all trusted channel functions	

5.1.1.3 TOE Security Functional Requirements for the TOE´s SIG Application

The following section gives a survey of the SFRs related to the TOE´s dedicated SIG Application as specified in the Protection Profile /PP SSCD Type3/, chap. 5.1. The SFRs of the Protection Profile have been supplemented appropriately.

For the TOE´s dedicated SIG Application, the TOE maintains an SFP as defined as follows:

SFP SIG Access Control

Subjects:

- User

Security attributes for subjects:

- General Attribute Role (Administrator, Signatory)
- Initialisation Attribute SCD/SVD Management (authorised, not authorised)

Objects:

- SCD
- DTBS

Security attributes for objects:

- For object SCD: SCD Operational (no, yes)
- For object DTBS: Sent by an authorised SCA (no, yes)

Operations (Access Modes):

- Signature key pair generation
- Export of SVD
- Creation and import of RAD
- Generation of electronic signatures

The SFP SIG Access Control is subdivided into four SFPs according to /PP SSCD Type3/, chap. 5.1.2:

- SFP Initialisation (for the generation of SCD/SVD)
- SFP SVD Transfer (for the export of SVD)
- SFP Personalisation (for the creation and import of RAD)
- SFP Signature-Creation (for the generation of electronic signatures)

The related access rules for the TOE´s dedicated SIG Application are specified in detail within /PP SSCD Type3/, chap. 5.1.2.

For the personalisation of the TOE´s dedicated SIG Application in the sense of loading the personalisation data by usage of the applicable commands of the MICARDO V3.0 operating system platform, the TOE maintains an SFP as defined as follows:

SFP SIG Personalisation

Subjects:

- Card Management System (for personalisation of the SIG Application)

Security attributes for subjects:

- USER_GROUP (authorised user, non-authorised user)

Objects:

- Personalisation data

Security attributes for objects:

- Access Rules

Operations (Access Modes):

- Loading of personalisation data by usage of the MICARDO V3.0 operating system commands

The SIG Access Control SFP controls the access of subjects to objects on the basis of security attributes. For a general description of the access rules handled by the TOE´s operating system refer to /ST-MIC30/, chap. 5.1.1.2. The access rules for the personalisation of the TOE´s SIG Application are explicitly set in such a manner that personalisation requires a preceding mutual authentication between the TOE and the external world.

Hint: The export of SVD is part of the above defined SFP SVD Transfer. The generation and personalisation of RAD is part of the above defined SFP SIG Personalisation.

FCS **Cryptographic Support**	
FCS_CKM **Cryptographic Key Management**	
FCS_CKM.1 **Cryptographic Key Generation**	PP SSCD Type3
FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *cryptographic key generation algorithm*] and specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: a) the management of changes to cryptographic key attributes. Examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption) Audit: a) Minimal: Success and failure of the activity b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	**FCS_CKM.1** **FCS_CKM.1.1** The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [***RSA key pair generation with randomly generated resp. externally chosen public exponent (up to 64 bit) (command GENERATE ASYMMETRIC KEY PAIR)***] and specified cryptographic key sizes [***1024, 1280, 1536, 1792 resp. 2048 bit modulus length***] that meet the following: [- ***/ALGCAT/, chap. 1.3, 3.1, 4***].
FCS_CKM.4 **Cryptographic Key Destruction**	PP SSCD Type3
FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: *cryptographic key destruction*	**FCS_CKM.4** **FCS_CKM.4.1** The TSF shall destroy cryptographic keys in accor-

method] that meets the following: [assignment: *list of standards*]. 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] - FMT_MSA.2 Secure security attributes Management: a) the management of changes to cryptographic key attributes. Examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption) Audit: a) Minimal: Success and failure of the activity b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	dance with a specified cryptographic key destruction method [***erasure of a private RSA key***] that meets the following: [***physical erasure of the key***]. **Application Note** The cryptographic key SCD will be destroyed on demand of the Signatory or Administrator. The destruction of the SCD is mandatory before the SCD/SVD pair is re-generated by the TOE.
FCS_COP **Cryptographic Operation**	
FCS_COP.1 **Cryptographic Operation**	PP SSCD Type3
FCS_COP.1.1 The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management:	**FCS_COP.1/CORRESP** **FCS_COP.1.1/CORRESP** The TSF shall perform [**SCD/SVD correspondence verification**] in accordance with a specified cryptographic algorithm [***generation of an RSA digital signature***] and cryptographic key sizes [***1024, 1280, 1536, 1792 resp. 2048 bit modulus length***] that meet the following: [- ***RSA signature scheme with appendix according to PKCS #1 (based on SHA-1, SHA-2 (224, 256, 384 resp. 512 bit) resp. RIPEMD-160 as hash algorithm): /PKCS1/, chap. 8.2.1 without hash value calculation inside step 1 of chap. 9.2; /HPC-SMC1/, chap. 11, /eHC1/, chap. 10*** ***or alternatively*** - ***RSA signature scheme with appendix according to ISO/IEC 9796-2 with random number (based on SHA-1, SHA-2 (224, 256, 384 resp. 512 bit) resp. RIPEMD-160 as***

--- Audit: a) Minimal: Success and failure, and the type of cryptographic operation b) Basic: Any applicable cryptographic mode(s) of operation, subject attributes and object attributes	***hash algorithm): /ISO 9796-2/ without hash value calculation; /HPC-SMC1/, chap. 11, /eHC1/, chap. 10***]. ***Note*** *The SCD/SVD correspondence verification shall be realised by the generation of a digital signature using the SCD (to be done by the signatory resp. the TOE) followed by the verification of the supplied signature by the external world using the corresponding SVD.*
	FCS_COP.1/SIGNING-PKCS1: **FCS_COP.1.1/SIGNING-PKCS1** The TSF shall perform [**digital signature-generation *(command PSO COMPUTE DIGITAL SIGNATURE)***] in accordance with a specified cryptographic algorithm [***RSA***] and cryptographic key sizes [***1024, 1280, 1536, 1792 resp. 2048 bit modulus length***] that meet the following: [- ***RSA signature scheme with appendix according to PKCS #1 (based on SHA-1, SHA-2 (224, 256, 384 resp. 512 bit) resp. RIPEMD-160 as hash algorithm): /PKCS1/, chap. 8.2.1 without hash value calculation inside step 1 of chap. 9.2; /HPC-SMC1/, chap. 11, /eHC1/, chap. 10***].
	FCS_COP.1/SIGNING-ISO9796-2: **FCS_COP.1.1/SIGNING-ISO9796-2** The TSF shall perform [**digital signature-generation *(command PSO COMPUTE DIGITAL SIGNATURE)***] in accordance with a specified cryptographic algorithm [***RSA***] and cryptographic key sizes [***1024, 1280, 1536, 1792 resp. 2048 bit modulus length***] that meet the following: [- ***RSA signature scheme with appendix according to ISO/IEC 9796-2 with random number (based on SHA-1, SHA-2 (224, 256, 384 resp. 512 bit) resp. RIPEMD-160 as hash algorithm): /ISO 9796-2/ without hash value calculation; /HPC-SMC1/, chap. 11, /eHC1/, chap. 10***].

FDP **User Data Protection**	
FDP_ACC	

Access Control Policy	
FDP_ACC.1 **Subset Access Control**	PP SSCD Type3
FDP_ACC.1.1 The TSF shall enforce the [assignment: *access control SFP*] on [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]. Hierarchical to: No other components Dependencies: - FDP_ACF.1 Security attribute based access control Management: --- Audit: ---	**FDP_ACC.1/SVD Transfer SFP** **FDP_ACC.1.1/SVD Transfer SFP** The TSF shall enforce the [**SVD Transfer SFP**] on [**export of SVD by User**].
	FDP_ACC.1/Initialisation SFP **FDP_ACC.1.1/Initialisation SFP** The TSF shall enforce the [**Initialisation SFP**] on [**generation of SCD/SVD pair by User**].
	FDP_ACC.1/Personalisation SFP **FDP_ACC.1/Personalisation SFP** The TSF shall enforce the [**Personalisation SFP**] on [**creation of RAD by Administrator**].
	FDP_ACC.1/Signature-Creation SFP **FDP_ACC.1/Signature-Creation SFP** The TSF shall enforce the [**Signature-Creation SFP**] on [**1. sending of DTBS-representation by SCA, 2. signing of DTBS-representation by Signatory**].
	FDP_ACC.1/SIG Personalisation SFP ***FDP_ACC.1.1/SIG Personalisation SFP*** *The TSF shall enforce the [**SIG Personalisation SFP**] on [**import of personalisation data by Administrator**].*
FDP_ACF **Access Control Functions**	
FDP_ACF.1 **Security Attribute Based Access Control**	PP SSCD Type3
FDP_ACF.1.1	**FDP_ACF.1/SVD Transfer SFP**

The TSF shall enforce the [assignment: *access control SFP*] to objects based on the following: [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*]. **FDP_ACF.1.2** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]. **FDP_ACF.1.3** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]. **FDP_ACF.1.4** The TSF shall explicitly deny access of subjects to objects based on the [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]. Hierarchical to: No other components Dependencies: - FDP_ACC.1 Subset access control - FMT_MSA.3 Static attribute initialisation Management: a) Managing the attributes used to make explicit access or denial based decisions Audit: a) Minimal: Successful requests to perform an operation on an object covered by the SFP b) Basic: All requests to perform an operation on an object covered by the SFP c) Detailed: The specific security attributes used in making an access check	**FDP_ACF.1.1/SVD Transfer SFP** The TSF shall enforce the [**SVD Transfer SFP**] to objects based on the following: [**General attribute**]. **FDP_ACF.1.2/SVD Transfer SFP** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**The user with the security attribute “role” set to “Administrator” or to “Signatory” is allowed to export SVD**]. **FDP_ACF.1.3/SVD Transfer SFP** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**none**]. **FDP_ACF.1.4/SVD Transfer SFP** The TSF shall explicitly deny access of subjects to objects based on the [**none**].
	FDP_ACF.1/Initialisation SFP **FDP_ACF.1.1/Initialisation SFP** The TSF shall enforce the [**Initialisation SFP**] to objects based on the following: [**General attribute and Initialisation attribute**]. **FDP_ACF.1.2/Initialisation SFP** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**The user with the security attribute “role” set to “Administrator” or set to “Signatory” and with the security attribute “SCD /**

	SVD management" set to " authorised" is allowed to generate SCD/SVD pair]. **FDP_ACF.1.3/Initialisation SFP** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**none**]. **FDP_ACF.1.4/Initialisation SFP** The TSF shall explicitly deny access of subjects to objects based on the [**rule: The user with the security attribute "role" set to "Administrator" or set to "Signatory" and with the security attribute "SCD / SVD management" set to "not authorised" is not allowed to generate SCD/SVD pair**].
	FDP_ACF.1/Personalisation SFP **FDP_ACF.1.1/Personalisation SFP** The TSF shall enforce the [**Personalisation SFP**] to objects based on the following: [**General attribute**]. **FDP_ACF.1.2/Personalisation SFP** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**User with the security attribute "role" set to "Administrator" is allowed to create the RAD**]. **FDP_ACF.1.3/Personalisation SFP** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**none**]. **FDP_ACF.1.4/Personalisation SFP** The TSF shall explicitly deny access of subjects to objects based on the [**none**].
	FDP_ACF.1/Signature-Creation SFP **FDP_ACF.1.1/Signature-Creation SFP** The TSF shall enforce the [**Signature-creation SFP**] to objects based on the following: [**General attribute and Signature-creation attribute group**]. **FDP_ACF.1.2/Signature-Creation SFP** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**User with the security attribute "role" set to "Signatory" is allowed to create electronic signatures for DTBS sent by an authorised SCA with SCD by the Signatory which security attribute "SCD operational" is set to "yes"**]. **FDP_ACF.1.3/Signature-Creation SFP** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**none**].

	FDP_ACF.1.4/Signature-Creation SFP The TSF shall explicitly deny access of subjects to objects based on the [**rule: (a) User with the security attribute "role" set to "Signatory" is not allowed to create electronic signatures for DTBS which is not sent by an authorised SCA with SCD by the Signatory which security attribute "SCD operational" is set to "yes"; (b) User with the security attribute "role" set to "Signatory" is not allowed to create electronic signatures for DTBS sent by an authorised SCA with SCD by the Signatory which security attribute "SCD operational" is set to "no"**]. Application Note A SCA is authorised to send the DTBS-representation if it is actually used by the Signatory to create an electronic signature. The Signatory controls wether a trusted channel to the SSCD by cryptographic means as required by FTP_ITC.1.3/SCA DTBS is established or a channel to the SSCD within a trusted environment is set-up.
	FDP_ACF.1/SIG Personalisation SFP ***FDP_ACF.1.1/SIG Personalisation SFP*** *The TSF shall enforce the [**SIG Application Personalisation SFP**] to objects based on the following: [**authentication status of user**].* ***FDP_ACF.1.2/SIG Personalisation SFP*** *The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**The Card Management System is allowed to perform the smartcard personalisation process (loading of the personalisation data related to the TOE´s SIG Application)**].* ***FDP_ACF.1.3/SIG Personalisation SFP*** *The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**none**].* ***FDP_ACF.1.4/SIG Personalisation SFP*** *The TSF shall explicitly deny access of subjects to objects based on the [**none**].*
FDP_ETC **Export to Outside TSF Control**	
FDP_ETC.1 **Export of User Data without Security Attributes**	PP SSCD Type3
FDP_ETC.1.1 The TSF shall enforce the [assignment: *access control SFP(s) and/or information flow control SFP(s)*] when exporting user data, controlled under the	**FDP_ETC.1/SVD Transfer** **FDP_ETC.1.1/SVD Transfer** The TSF shall enforce the [**SVD Transfer SFP**] when

SFP(s), outside of the TSC. **FDP_ETC.1.2** The TSF shall export the user data without the user data's associated security attributes. Hierarchical to: No other components Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] Management: --- Audit: a) Minimal: Successful export of information b) Basic: All attempts to export information	exporting user data, controlled under the SFP(s), outside of the TSC. **FDP_ETC.1.2/SVD Transfer** The TSF shall export the user data without the user data's associated security attributes.
FDP_ITC **Import from Outside TSF Control**	
FDP_ITC.1 **Import of User Data without Security Attributes**	PP SSCD Type3
FDP_ITC.1.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] when importing user data, controlled under the SFP, from outside of the TSC. **FDP_ITC.1.2** The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC. **FDP_ITC.1.3** The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: [assignment: *additional importation control rules*]. Hierarchical to: No other components Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] - FMT_MSA.3 Static attribute initialisation Management: a) The modification of the additional control rules used for import Audit:	**FDP_ITC.1/DTBS** **FDP_ITC.1.1/DTBS** The TSF shall enforce the [**Signature-Creation SFP**] when importing user data, controlled under the SFP, from outside of the TSC. **FDP_ITC.1.2/DTBS** The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC. **FDP_ITC.1.3/DTBS** The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: [**DTBS-representation shall be sent by an authorised SCA**]. **Application Note** A SCA is authorised to send the DTBS-representation if it is actually used by the Signatory to create an electronic signature. The Signatory controls wether a trusted channel to the SSCD by cryptographic means as required by FTP_ITC.1.3/SCA DTBS is established or a channel to the SSCD within a trusted environment is set-up.

a) Minimal: Successful import of user data, including any security attributes b) Basic: All attempts to import user data, including any security attributes c) Detailed: The specification of security attributes for imported user data supplied by an authorised user	
FDP_RIP **Residual Information Protection**	
FDP_RIP.1 **Subset Residual Information Protection**	PP SSCD Type3
FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection: *allocation of the resource to*, *deallocation of the resource from*] the following objects: [assignment: *list of objects*]. Hierarchical to: No other components Dependencies: No dependencies Management: a) The choice of when to perform residual information protection (i.e. upon allocation or deallocation) could be made configurable within the TOE Audit: ---	**FDP_RIP.1** **FDP_RIP.1.1** The TSF shall ensure that any previous information content of a resource is made unavailable upon the [**deallocation of the resource from**] the following objects: [**SCD, VAD, RAD**].
FDP_SDI **Stored Data Integrity**	
FDP_SDI.2 **Stored Data Integrity Monitoring and Action**	PP SSCD Type3
FDP_SDI.2.1 The TSF shall monitor user data stored within the TSC for [assignment: *integrity errors*] on all objects, based on the following attributes: [assignment: *user data attributes*]. **FDP_SDI.2.2** Upon detection of a data integrity error, the TSF shall [assignment: *action to be taken*]. Hierarchical to: FDP_SDI.1 Dependencies: No dependencies	**Note** The following data persistently stored by TOE have the user data attribute “integrity checked persistent stored data”: 1. SCD, 2. RAD, 3. SVD (if persistent stored by TOE). **FDP_SDI.2/Persistent** **FDP_SDI.2.1/Persistent** The TSF shall monitor user data stored within the TSC for [**integrity error**] on all objects, based on the following attributes: [**integrity checked persistent stored data**]. **FDP_SDI.2.2/Persistent** Upon detection of a data integrity error, the TSF shall

Management: a) The actions to be taken upon the detection of an integrity error could be configurable Audit: a) Minimal: Successful attempts to check the integrity of user data, including an indication of the results of the check b) Basic: All attempts to check the integrity of user data, including an indication of the results of the check, if performed c) Detailed: The type of integrity error that occurred d) Detailed: The action taken upon detection of an integrity error	[**1. prohibit the use of the altered data, 2. inform the Signatory about integrity error**].
	Note The DTBS-representation temporarily stored by TOE has the user data attribute “integrity checked stored data”. **FDP_SDI.2/DTBS** **FDP_SDI.2.1/DTBS** The TSF shall monitor user data stored within the TSC for [**integrity error**] on all objects, based on the following attributes: [**integrity checked stored data**]. **FDP_SDI.2.2/DTBS** Upon detection of a data integrity error, the TSF shall [**1. prohibit the use of the altered data, 2. inform the Signatory about integrity error**].
FDP_UIT **Inter-TSF User Data Integrity Transfer Protection**	
FDP_UIT.1 **Data Exchange Integrity**	PP SSCD Type3
FDP_UIT.1.1 The TSF shall enforce the [assignment: *access control SFP(s) and/or* i*nformation flow control SFP(s)*] to be able to [selection*: transmit, receive*] user data in a manner protected from [selection: *modification, deletion, insertion, replay*] errors. **FDP_UIT.1.2** The TSF shall be able to determine on receipt of user data, whether [selection: *modification*, *deletion, insertion, replay*] has occurred. Hierarchical to: No other components Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] - [FTP_ITC.1 Inter-TSF trusted channel	**FDP_UIT.1/SVD Transfer** **FDP_UIT.1.1/SVD Transfer** The TSF shall enforce the [**SVD Transfer SFP**] to be able to [**transmit**] user data in a manner protected from [**modification and insertion**] errors. **FDP_UIT.1.2/SVD Transfer** The TSF shall be able to determine on receipt of user data, whether [**modification and insertion**] has occurred.

or FTP_TRP.1 Trusted path] Management: --- Audit: a) Minimal: The identity of any user or subject using the data exchange mechanisms b) Basic: The identity of any user or subject attempting to use the user data exchange mechanisms, but who is unauthorised to do so c) Basic: A reference to the names or other indexing information useful in identifying the user data that was transmitted or received; this could include security attributes associated with the user data d) Basic: Any identified attempts to block transmission of user data e) Detailed: The types and/or effects of any detected modifications of transmitted user data	
	FDP_UIT.1/TOE DTBS **FDP_UIT.1.1/TOE DTBS** The TSF shall enforce the [**Signature-Creation SFP**] to be able to [**receive**] user data in a manner protected from [**modification, deletion and insertion**] errors. **FDP_UIT.1.2/TOE DTBS** The TSF shall be able to determine on receipt of user data, whether [**modification, deletion and insertion**] has occurred. **Application Note** Protection for FDP_UIT.1.1/TOE DTBS can either be assured by a trusted channel to the SSCD by cryptographic means or by a channel to the SSCD within a trusted environment.

FIA **Identification and Authentication**	
FIA_AFL **Authentication Failures**	
FIA_AFL.1 **Authentication Failure Handling**	PP SSCD Type3
FIA_AFL.1.1 The TSF shall detect when [selection: [assignment: *positive integer number*], “*an administrator configurable positive integer within* [assignment: *range of acceptable values*]“] unsuccessful authentication	**FIA_AFL.1** **FIA_AFL.1.1** The TSF shall detect when [***3***] unsuccessful authentication attempts occur related to [**consecutive failed**

attempts occur related to [assignment: *list of authentication events*]. **FIA_AFL.1.2** When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [assignment: *list of actions*]. <u>Hierarchical to:</u> No other components <u>Dependencies:</u> - FIA_UAU.1 Timing of authentication <u>Management:</u> a) management of the threshold for unsuccessful authentication attempts b) management of actions to be taken in the event of an authentication failure <u>Audit:</u> a) Minimal: the reaching of the threshold for the unsuccessful authentication attempts and the actions (e.g. disabling of a terminal) taken and the subsequent, if appropriate, restoration to the normal state (e.g. re-enabling of a terminal)	**authentication attempts**]. **FIA_AFL.1.2** When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [**block RAD**].
FIA_ATD **User Attribute Definition**	
FIA_ATD.1 **User Attribute Definition**	PP SSCD Type3
FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: [assignment: *list of security attributes*]. <u>Hierarchical to:</u> No other components <u>Dependencies:</u> No dependencies <u>Management:</u> a) if so indicated in the assignment, the authorised administrator might be able to define additional security attributes for users <u>Audit:</u> ---	**FIA_ATD.1** **FIA_ATD.1.1** The TSF shall maintain the following list of security attributes belonging to individual users: [**RAD**].
FIA_UAU **User Authentication**	
FIA_UAU.1 **Timing of Authentication**	PP SSCD Type3

FIA_UAU.1.1 The TSF shall allow [assignment: *list of TSF mediated actions*] 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. Hierarchical to: No other components Dependencies: - FIA_UID.1 Timing of identification Management: a) management of the authentication data by an administrator b) management of the authentication data by the associated user c) managing the list of actions that can be taken before the user is authenticated Audit: a) Minimal: Unsuccessful use of the authentication mechanism b) Basic: All use of the authentication mechanism c) Detailed: All TSF mediated actions performed before authentication of the user	**FIA_UAU.1** **FIA_UAU.1.1** The TSF shall allow [**1. identification of the user by means of TSF required by FIA_UID.1, 2. establishing a trusted path between local user and the TOE by means of TSF required by FTP_TRP.1 / TOE, 3. establishing a trusted channel between the SCA and the TOE by means of TSF required by FTP_ITC.1 / DTBS import**] on behalf of the user to be performed before the user is authenticated. **FIA_UAU.1.2** The TSF shall require each user to be successfully authenticated before allowing any other TSF- mediated actions on behalf of that user. **Application Note** “Local user” mentioned in component FIA_UAU.1.1 is the user using the trusted path provided between the SCA in the TOE environment and the TOE as indicated by FTP_TRP.1/SCA and FTP_TRP.1/TOE.
FIA_UID **User Identification**	
FIA_UID.1 **Timing of Identification**	PP SSCD Type3
FIA_UID.1.1 The TSF shall allow [assignment: *list of TSF-mediated actions*] 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. Hierarchical to: No other components Dependencies: No dependencies Management: a) the management of the user identities b) if an authorised administrator can change the actions allowed before identification, the managing of	**FIA_UID.1** **FIA_UID.1.1** The TSF shall allow [**1. establishing a trusted path between local user and the TOE by means of TSF required by FTP_TRP.1 / TOE, 2. establishing a trusted channel between the SCA and the TOE by means of TSF required by FTP_ITC.1 / DTBS import**] on behalf of the user to be performed before the user is identified. **FIA_UID.1.2** The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

the action lists Audit: a) Minimal: Unsuccessful use of the user identification mechanism, including the user identity provided b) Basic: All use of the user identification mechanism, including the user identity provided	

FMT **Security Management**	
FMT_MOF **Management of Functions in TSF**	
FMT_MOF.1 **Management of Security Functions Behaviour**	PP SSCD Type3
FMT_MOF.1.1 The TSF shall restrict the ability to [selection: *determine the behaviour of, disable, enable, modify the behaviour of*] the functions [assignment: *list of functions*] to [assignment: *the authorised identified roles*]. Hierarchical to: No other components Dependencies: - FMT_SMF.1 Specification of management functions - FMT_SMR.1 Security roles Management: a) managing the group of roles that can interact with the functions in the TSF Audit: a) Basic: All modifications in the behaviour of the functions in the TSF	**FMT_MOF.1** **FMT_MOF.1.1** The TSF shall restrict the ability to [**enable**] the functions [**signature-creation function**] to [**Signatory**].
FMT_MSA **Management of Security Attributes**	
FMT_MSA.1 **Management of Security Attributes**	PP SSCD Type3
FMT_MSA.1.1 The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] to restrict the ability to [selection: *change_default, query, modify, delete,* [assignment: *other operations*]] the security attributes [assignment: *list of security attributes*] to [assignment: *the authorised identified roles*].	**FMT_MSA.1/Administrator** **FMT_MSA.1.1/Administrator** The TSF shall enforce the [**Initialisation SFP**] to restrict the ability to [**modify**] the security attributes [**SCD/SVD management**] to [**Administrator**].

Hierarchical to: No other components Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] - FMT_SMF.1 Specification of management functions - FMT_SMR.1 Security roles Management: a) managing the group of roles that can interact with the security attributes Audit: a) Basic: All modifications of the values of security attributes	
	FMT_MSA.1/Signatory **FMT_MSA.1.1/Signatory** The TSF shall enforce the [**Signature-Creation SFP**] to restrict the ability to [**modify**] the security attributes [**SCD operational**] to [**Signatory**].
	FMT_MSA.1/SIG Personalisation ***FMT_MSA.1.1/SIG Personalisation*** *The TSF shall enforce the [**SIG Personalisation SFP**] to restrict the ability to [**modify**] the security attributes [**access rules**] to [**none**].*
FMT_MSA.2 **Secure Security Attributes**	PP SSCD Type3
FMT_MSA.2.1 The TSF shall ensure that only secure values are accepted for security attributes. Hierarchical to: No other components Dependencies: - ADV_SPM.1 Informal TOE security policy model - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] - FMT_MSA.1 Management of security attributes - FMT_SMR.1 Security roles Management: --- Audit: a) Minimal: All offered and rejected values for a security attribute b) Detailed: All offered and accepted secure values	**FMT_MSA.2** **FMT_MSA.2.1** The TSF shall ensure that only secure values are accepted for security attributes.

for a security attribute	
FMT_MSA.3 **Static Attribute Initialisation**	PP SSCD Type3
FMT_MSA.3.1 The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] to provide [selection: *choose one of: restrictive, permissive, [assignment: other property]*] default values for security attributes that are used to enforce the SFP. **FMT_MSA.3.2** The TSF shall allow the [assignment: *the authorised identified roles*] to specify alternative initial values to override the default values when an object or information is created. Hierarchical to: No other components Dependencies: - FMT_MSA.1 Management of security attributes - FMT_SMR.1 Security roles Management: a) managing the group of roles that can specify initial values b) managing the permissive or restrictive setting of default values for a given access control SFP Audit: a) Basic: Modifications of the default setting of permissive or restrictive rules b) Basic: All modifications of the initial values of security attributes	**FMT_MSA.3** **FMT_MSA.3.1** The TSF shall enforce the [**Initialisation SFP and Signature-Creation SFP**] to provide [**restrictive**] default values for security attributes that are used to enforce the SFP. **Refinement** The security attribute of the SCD “SCD operational” is set to “no” after generation of the SCD. **FMT_MSA.3.2** The TSF shall allow the [**Administrator**] to specify alternative initial values to override the default values when an object or information is created.
FMT_MTD **Management of TSF Data**	
FMT_MTD.1 **Management of TSF Data**	PP SSCD Type3
FMT_MTD.1.1 The TSF shall restrict the ability to [selection: *change_default, query, modify, delete, clear*, [assignment: *other operations*]] the [assignment: *list of TSF data*] to [assignment: *the authorised identified roles*]. Hierarchical to: No other components Dependencies: - FMT_SMF.1 Specification of management functions - FMT_SMR.1 Security roles	**FMT_MTD.1** **FMT_MTD.1.1** The TSF shall restrict the ability to [**modify**] the [**RAD**] to [**Signatory**].

Management: a) managing the group of roles that can interact with the TSF data Audit: a) Basic: All modifications to the values of TSF data	
FMT_SMF **Specification of Management Functions**	
FMT_SMF.1 **Specification of Management Functions**	PP SSCD Type3
FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions: [assignment: *list of security management functions to be provided by the TSF*]. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: a) Minimal: Use of the management functions	***FMT_SMF.1*** ***FMT_SMF.1.1*** *The TSF shall be capable of performing the following security management functions: [**security function management, security attribute management, TSF data management**].* ***Note*** *This SFR has been added to the SFRs defined in the SSCD Protection Profile due to /AIS 32/.*
FMT_SMR **Security Management Roles**	
FMT_SMR.1 **Security Roles**	PP SSCD Type3
FMT_SMR.1.1 The TSF shall maintain the roles [assignment: *the authorised identified roles*]. **FMT_SMR.1.2** The TSF shall be able to associate users with roles. Hierarchical to: No other components Dependencies: - FIA_UID.1 Timing of identification Management: a) managing the group of users that are part of a role Audit: a) Minimal: modifications to the group of users that	**FMT_SMR.1** **FMT_SMR.1.1** The TSF shall maintain the roles [**Administrator, Signatory, Card Management System**]. **FMT_SMR.1.2** The TSF shall be able to associate users with roles.

are part of a role b) Detailed: every use of the rights of a role	

FPT **Protection of the TSF**	
FPT_AMT **Underlying Abstract Machine Test**	
FPT_AMT.1 **Abstract Machine Testing**	PP SSCD Type3
FPT_AMT.1.1 The TSF shall run a suite of tests [selection: *during initial start-up, periodically during normal operation, at the request of an authorised user, other conditions*] to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF. <u>Hierarchical to:</u> No other components <u>Dependencies:</u> No dependencies <u>Management:</u> a) management of the conditions under which abstract machine test occurs, such as during initial start-up, regular interval, or under specified conditions b) management of the time interval if appropriate <u>Audit:</u> a) Basic: Execution of the tests of the underlying machine and the results ofthe tests	**FPT_AMT.1** **FPT_AMT.1.1** The TSF shall run a suite of tests [***during initial start-up, periodically during normal operation***] to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF. ***Application Note*** *The test of the underlying abstract machine is performed in the framework of the self test functionality of the TOE (refer to SFR FPT_TST.1).*
FPT_EMSEC **TOE Emanation**	
FPT_EMSEC.1 **TOE Emanation**	PP SSCD Type3
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*	**FPT_EMSEC.1** **FPT_EMSEC.1.1** The TOE shall not emit [***information on IC power consumption, information on command execution time, information on electromagnetic emanations***] in excess of [***non useful information***] enabling access to [**RAD**] and [**SCD**]. **FPT_EMSEC.1.2**

of types of TSF data] and [assignment: *list of types of user data*]. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: ---	The TSF shall ensure [***S.OFFCARD***] are unable to use the following interface [***IC contacts as Vcc, I/O and GND, IC surface***] to gain access to [**RAD**] and [**SCD**]. **Application Note** The TOE shall prevent attacks against the SCD and other secret data where the attack is based on external observable physical phenomena of the TOE. Such attacks may be observable at the interfaces of the TOE or may origin from internal operation of the TOE or may origin by an attacker that varies the physical environment under which the TOE operates. The set of measurable physical phenomena is influenced by the technology employed to implement the TOE. Examples of measurable phenomena are variations in the power consumption, the timing of transitions of internal states, electromagnetic radiation due to internal operation, radio emission. Due to the heterogeneous nature of the technologies that may cause such emanations, evaluation against state-of-the-art attacks applicable to the technologies employed by the TOE is assumed. Examples of such attacks are, but are not limited to, evaluation of TOE's electromagnetic radiation**,** simple power analysis (SPA), differential power analysis (DPA), timing attacks, etc.
FPT_FLS **Fail Secure**	
FPT_FLS.1 **Failure with Preservation of Secure State**	PP SSCD Type3
FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [assignment: *list of types of failures in the TSF*]. Hierarchical to: No other components Dependencies: - ADV_SPM.1 Informal TOE security policy model Management: --- Audit: a) Basic: Failure of the TSF	**FPT_FLS.1** **FPT_FLS.1.1** The TSF shall preserve a secure state when the following types of failures occur: [- ***HW and/or SW induced reset*** - ***Power supply cut-off or variations*** - ***Unexpected abortion of the execution of the TSF due to external or internal events (in particular, break of a transaction before completion)*** - ***System breakdown*** - ***Internal HW and/or SW failure*** - ***Manipulation of executable code*** - ***Corruption of status information (as e.g. card status information, object life cycle state, actual security state related to key and PIN based authentication, ...)*** - ***Environmental stress*** - ***Input of inconsistent or improper data*** - ***Tampering*** - ***Manipulation resp. insufficient quality of the***

	HW-RNG resp. SW-RNG - ***Fault injection attacks*** - ***Exposure to operating conditions where therefore a malfunction could occur*** - ***Failure detected by TSF according to FPT_TST.1***]. ***Refinements*** *The TOE shall preserve a secure state during power supply cut-off or variations. If power is cut or if power variations occur from the TOE, or if a transaction is stopped before completion, or on any other reset conditions, the TOE shall be reset cleanly.*
FPT_PHP **Physical Protection**	
FPT_PHP.1 **Passive Detection of Physical Attack**	PP SSCD Type3
FPT_PHP.1.1 The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF. **FPT_PHP.1.2** The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: a) Minimal: if detection by IT means, detection of intrusion.	**FPT_PHP.1** **FPT_PHP.1.1** The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF. **FPT_PHP.1.2** The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.
FPT_PHP.3 **Resistance to Physical Attack**	PP SSCD Type3
FPT_PHP.3.1 The TSF shall resist [assignment: *physical tampering scenarios*] to the [assignment: *list of TSF devices / elements*] by responding automatically such that the TSP is not violated. Hierarchical to: No other components Dependencies:	**FPT_PHP.3** **FPT_PHP.3.1** The TSF shall resist [***physical manipulation and physical probing (e.g. tampering of the specified physical and technical operating conditions of the IC as voltage supply, clock frequency and temperature out of the valid limits)***] to the [***TSF***] by responding automatically such that the TSP is not violated.

No dependencies Management: a) management of the automatic responses to physical tampering Audit: ---	
FPT_TST **TSF Self Test**	
FPT_TST.1 **TSF Testing**	PP SSCD Type3
FPT_TST.1.1 The TSF shall run a suite of self tests [selection: *during initial start-up, periodically during normal operation, at the request of the authorised user, at the conditions* [assignment: *conditions under which self test should occur*]] to demonstrate the correct operation of [selection: [assignment: *parts of TSF*], *the TSF*]. **FPT_TST.1.2** The TSF shall provide authorised users with the capability to verify the integrity of [selection: [assignment: *parts of TSF data*], *TSF data*]. **FPT_TST.1.3** The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code. Hierarchical to: No other components Dependencies: - FPT_AMT.1 Abstract machine testing Management: a) management of the conditions under which TSF self testing occurs, such as during initial start-up, regular interval, or under specified conditions b) management of the time interval if appropriate Audit: a) Basic: Execution of the TSF self tests and the results of the tests	**FPT_TST.1** **FPT_TST.1.1** The TSF shall run a suite of self tests [***during initial start-up, periodically during normal operation***] to demonstrate the correct operation of [***the TSF***]. ***Note*** *During initial start-up means before code execution.* ***Refinements*** *The TOE's self tests shall include the verification of the integrity of any software code (incl. patches) stored outside of the ROM. Upon detection of a self test error the TSF shall warn the entity connected. After OS testing is completed, all testing-specific commands and actions shall be disabled or removed. It shall not be possible to override these controls and restore them for use. Command associated exclusively with one life cycle state shall never be accessed during another state.* **FPT_TST.1.2** The TSF shall provide authorised users with the capability to verify the integrity of [***TSF data***]. ***Refinement*** *In this framework, the OS (i.e. the Smartcard Embedded Software of the TOE (TOE-ES)) itself is understood as „authorised user“.* **FPT_TST.1.3** The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code. ***Refinement*** *The integrity check over the executable code stored outside the ROM area is covered by FPT_TST.1.1 and the related refinement.* *The requirement for checking the integrity of the ROM-code shall concern only the production phase,*

	more precise the initialisation phase of the TOE´s lifecycle. Prior to the initialisation of the TOE, the ROM-code of the TOE shall be verifiable by authorised users as the OS developer. The integrity of the ROM-code shall be provable only during the initialisation process.

FTP **Trusted Path/Channels**	
FTP_ITC **Inter-TSF Trusted Channel**	
FTP_ITC.1 **Inter-TSF Trusted Channel**	PP SSCD Type3
FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. **FTP_ITC.1.2** The TSF shall permit [selection: *the TSF, the remote trusted IT product*] to initiate communication via the trusted channel. **FTP_ITC.1.3** The TSF shall initiate communication via the trusted channel for [assignment: *list of functions for which a trusted channel is required*]. Hierarchical to: No other components Dependencies: No dependencies Management: a) Configuring the actions that require trusted channel, if supported Audit: a) Minimal: Failure of the trusted channel functions b) Minimal: Identification of the initiator and target of failed trusted channel functions c) Basic: All attempted uses of the trusted channel functions d) Basic: Identification of the initiator and target of all trusted channel functions	**FTP_ITC.1/SVD Transfer** **FTP_ITC.1.1/SVD Transfer** The TSF shall provide a communication channel between itself and a remote trusted IT product **CGA** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. **FTP_ITC.1.2/SVD Transfer** The TSF shall permit [***the remote trusted IT product CGA***] to initiate communication via the trusted channel. **FTP_ITC.1.3/SVD Transfer** The TSF **or the CGA** shall initiate communication via the trusted channel for [**export SVD**].
	FTP_ITC.1/DTBS Import

	FTP_ITC.1.1/DTBS Import The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. **FTP_ITC.1.2/DTBS Import** The TSF shall permit [**the remote trusted IT product SCA**] to initiate communication via the trusted channel. **FTP_ITC.1.3/DTBS Import** The TSF **or the SCA** shall initiate communication via the trusted channel for [**signing DTBS-representation**]. **Application Note** For the communication channel either a trusted channel to the SSCD by cryptographic means or a channel to the SSCD within a trusted environment can be used. In the latter case the TOE identifies the establishment of a trusted environment by a successful user authentication.
	FTP_ITC.1/SIG Personalisation ***FTP_ITC.1.1/ SIG Personalisation*** *The TSF shall provide a communication channel between itself and a remote trusted IT product* ***Card Management System*** *that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.* ***FTP_ITC.1.2/SIG Personalisation*** *The TSF shall permit [***the remote trusted IT product (Card Management System)****] to initiate communication via the trusted channel.* ***FTP_ITC.1.3/SIG Personalisation*** *The TSF* ***or the Card Management System*** *shall initiate communication via the trusted channel for [***import of personalisation data****].*
FTP_TRP **Trusted Path**	
FTP_TRP.1 **Trusted Path**	PP SSCD Type3
FTP_TRP.1.1 The TSF shall provide a communication path between itself and [selection: *remote, local*] users that is logically distinct from other communication paths and provides assured identification of its end points and	**FTP_TRP.1/TOE** **FTP_TRP.1.1/TOE** The TSF shall provide a communication path between itself and [**local**] users that is logically distinct from

protection of the communicated data from modification or disclosure. **FTP_TRP.1.2** The TSF shall permit [selection: *the TSF, local users, remote users*] to initiate communication via the trusted path. **FTP_TRP.1.3** The TSF shall require the use of the trusted path for [selection: *initial user authentication*, [assignment: *other services for which trusted path is required*]]. Hierarchical to: No other components Dependencies: No dependencies Management: a) Configuring the actions that require trusted path, if supported Audit: a) Minimal: Failures of the trusted path functions b) Minimal: Identification of the user associated with all trusted path failures, if available c) Basic: All attempted uses of the trusted path functions d) Basic: Identification of the user associated with all trusted path invocations, if available	other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure. **FTP_TRP.1.2/TOE** The TSF shall permit [***local users***] to initiate communication via the trusted path. **FTP_TRP.1.3/TOE** The TSF shall require the use of the trusted path for [***none***]. **Application Note** For the communication path either a trusted path to the SSCD by cryptographic means or a path to the SSCD within a trusted environment can be used. In the latter case the TOE identifies the establishment of a trusted environment by a successful user authentication.

5.1.2 SOF Claim for TOE Security Functional Requirements

The required level for the Strength of Function of the TOE security functional requirements listed in the preceding chap. 5.1.1 is “SOF-high”. This correlates to the claimed assurance level with its augmentation by the assurance component AVA_VLA.4 (refer to the following chap. 5.1.3).

5.1.3 TOE Security Assurance Requirements

The TOE security assurance level is fixed as

EAL4 augmented by ADV_IMP.2, ATE_DPT.2, AVA_MSU.3 and AVA_VLA.4.

The assurance level with its augmentations is chosen in view of the requirements in the Protection Profiles /PP-HPC/ and /PP SSCD Type3/ and in correspondence with the CC evaluation level of the underlying smartcard product “MICARDO V3.0 R1.0” (Certification ID BSI-DSZ-CC-0390).

The following table lists the security assurance requirements (SARs) for the TOE:

SAR	
Class ACM **Configuration Management**	ACM_AUT.1 Partial CM Automation
	ACM_CAP.4 Generation Support and Acceptance Procedures
	ACM_SCP.2 Problem Tracking CM Coverage
Class ADO **Delivery and Operation**	ADO_DEL.2 Detection of Modification
	ADO_IGS.1 Installation, Generation, and Start-up Procedures
Class ADV **Development**	ADV_FSP.2 Fully Defined External Interfaces
	ADV_HLD.2 Security Enforcing High-Level Design
	ADV_IMP.**2** Implementation of the TSF
	ADV_LLD.1 Descriptive Low-Level Design
	ADV_RCR.1 Informal Correspondence Demonstration
	ADV_SPM.1 Informal TOE Security Policy Model
Class AGD **Guidance Documents**	AGD_ADM.1 Administrator Guidance
	AGD_USR.1 User Guidance
Class ALC **Life Cycle Support**	ALC_DVS.1 Identification of Security Measures
	ALC_LCD.1 Developer Defined Life-Cycle Model
	ALC_TAT.1 Well-defined Development Tools
Class ATE **Tests**	ATE_COV.2 Analysis of Coverage
	ATE_DPT.**2** Testing: Low-Level Design

	ATE_FUN.1 Functional Testing
	ATE_IND.2 Independent Testing – Sample
Class AVA Vulnerability Assessment	AVA_MSU.**3** Analysis and Testing for Insecure States
	AVA_SOF.1 Strength of TOE Security Function Evaluation
	AVA_VLA.**4** Highly Resistant

5.1.4 Refinements of the TOE Security Assurance Requirements

All assurance components given in the table of chap. 5.1.3 are used as defined in /CC 2.3 Part3/ and /CEM 2.3 Part2/.

5.2 Security Requirements for the Environment of the TOE

5.2.1 Security Requirements for the IT-Environment

The following sections cover the security requirements specified for the IT-environment of the TOE. Only the TOE´s dedicated SIG Application is affected.

5.2.1.1 Certification Generation Application (CGA)

For the Certification Generation Application (CGA), the following SFRs are defined according to /PP SSCD Type3/, chap. 5.3.1:

FCS Cryptographic Support	
FCS_CKM Cryptographic Key Management	
FCS_CKM.2 Cryptographic Key Distribution	
FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [assignment: *cryptographic key distribution method*] that meets the following: [assignment: *list of*	**FCS_CKM.2/CGA** **FCS_CKM.2.1/CGA** The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution

standards]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: a) the management of changes to cryptographic key attributes. Examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption) Audit: a) Minimal: Success and failure of the activity b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	method [**qualified certificate**] that meets the following: [***/ECDir/***].
FCS_CKM.3 **Cryptographic Key Access**	
FCS_CKM.3.1 The TSF shall perform [assignment: *type of cryptographic key access*] in accordance with a specified cryptographic key access method [assignment: *cryptographic key access method*] that meets the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: a) the management of changes to cryptographic key attributes; examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption) Audit: a) Minimal: Success and failure of the activity b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or	**FCS_CKM.3/CGA** **FCS_CKM.3.1/CGA** The TSF shall perform [**import the SVD**] in accordance with a specified cryptographic key access method [**import through a secure channel**] that meets the following: [***none***].

private keys)	

FDP **User Data Protection**	
FDP_UIT **Inter-TSF User Data Integrity Transfer Protection**	
FDP_UIT.1 **Data Exchange Integrity**	
FDP_UIT.1.1 The TSF shall enforce the [assignment: *access control SFP(s) and/or* i*nformation flow control SFP(s)*] to be able to [selection*: transmit, receive*] user data in a manner protected from [selection: *modification, deletion, insertion, replay*] errors. **FDP_UIT.1.2** The TSF shall be able to determine on receipt of user data, whether [selection: *modification*, *deletion, insertion, replay*] has occurred. Hierarchical to: No other components Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] - [FTP_ITC.1 Inter-TSF trusted channel or FTP_TRP.1 Trusted path] Management: --- Audit: a) Minimal: The identity of any user or subject using the data exchange mechanisms b) Basic: The identity of any user or subject attempting to use the user data exchange mechanisms, but who is unauthorised to do so c) Basic: A reference to the names or other indexing information useful in identifying the user data that was transmitted or received; this could include security attributes associated with the user data d) Basic: Any identified attempts to block transmission of user data e) Detailed: The types and/or effects of any detected modifications of transmitted user data	**FDP_UIT.1/SVD Import** **FDP_UIT.1.1/SVD Import** The TSF shall enforce the [**SVD Import SFP**] to be able to [**receive**] user data in a manner protected from [**modification and insertion**] errors. **FDP_UIT.1.2/SVD Import** The TSF shall be able to determine on receipt of user data, whether [**modification and insertion**] has occurred.

FTP **Trusted Path/Channels**	
FTP_ITC **Inter-TSF Trusted Channel**	
FTP_ITC.1 **Inter-TSF Trusted Channel**	
FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. **FTP_ITC.1.2** The TSF shall permit [selection: *the TSF, the remote trusted IT product*] to initiate communication via the trusted channel. **FTP_ITC.1.3** The TSF shall initiate communication via the trusted channel for [assignment: *list of functions for which a trusted channel is required*]. Hierarchical to: No other components Dependencies: No dependencies Management: a) Configuring the actions that require trusted channel, if supported Audit: a) Minimal: Failure of the trusted channel functions b) Minimal: Identification of the initiator and target of failed trusted channel functions c) Basic: All attempted uses of the trusted channel functions d) Basic: Identification of the initiator and target of all trusted channel functions	**FTP_ITC.1/SVD Import** **FTP_ITC.1.1/SVD Import** The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. **FTP_ITC.1.2/SVD Import** The TSF shall permit [***the TSF***] to initiate communication via the trusted channel. **FTP_ITC.1.3/SVD Import** The TSF **or the TOE** shall initiate communication via the trusted channel for [**import SVD**].

5.2.1.2 Signature Creation Application (SCA)

For the Signature Creation Application (SCA), the following SFRs are defined according to /PP SSCD Type3/, chap. 5.3.2:

FCS **Cryptographic Support**	
FCS_COP **Cryptographic Operation**	
FCS_COP.1 **Cryptographic Operation**	
FCS_COP.1.1 The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: --- Audit: a) Minimal: Success and failure, and the type of cryptographic operation b) Basic: Any applicable cryptographic mode(s) of operation, subject attributes and object attributes	**FCS_COP.1/SCA Hash** **FCS_COP.1.1/SCA Hash** The TSF shall perform [**hashing the DTBS**] in accordance with a specified cryptographic algorithm [***SHA-1***] and cryptographic key sizes [**none**] that meet the following: [**FIPS 180-2**].

FDP **User Data Protection**	
FDP_UIT **Inter-TSF User Data Integrity Transfer Protection**	
FDP_UIT.1 **Data Exchange Integrity**	
FDP_UIT.1.1 The TSF shall enforce the [assignment: *access control SFP(s) and/or information flow control SFP(s)*] to be able to [selection: *transmit, receive*] user data in a manner protected from [selection: *modification, deletion, insertion, replay*] errors.	**FDP_UIT.1/SCA DTBS** **FDP_UIT.1.1/SCA DTBS** The TSF shall enforce the [**Signature-Creation SFP**] to be able to [**transmit**] user data in a manner protected from [**modification, deletion and insertion**] errors.

FDP_UIT.1.2 The TSF shall be able to determine on receipt of user data, whether [selection: *modification*, *deletion, insertion, replay*] has occurred. Hierarchical to: No other components Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] - [FTP_ITC.1 Inter-TSF trusted channel or FTP_TRP.1 Trusted path] Management: --- Audit: a) Minimal: The identity of any user or subject using the data exchange mechanisms b) Basic: The identity of any user or subject attempting to use the user data exchange mechanisms, but who is unauthorised to do so c) Basic: A reference to the names or other indexing information useful in identifying the user data that was transmitted or received; this could include security attributes associated with the user data d) Basic: Any identified attempts to block transmission of user data e) Detailed: The types and/or effects of any detected modifications of transmitted user data	**FDP_UIT.1.2/SCA DTBS** The TSF shall be able to determine on receipt of user data, whether [**modification, deletion and insertion**] has occurred.

FTP **Trusted Path/Channels**	
FTP_ITC **Inter-TSF Trusted Channel**	
FTP_ITC.1 **Inter-TSF Trusted Channel**	
FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. **FTP_ITC.1.2** The TSF shall permit [selection: *the TSF, the remote trusted IT product*] to initiate communication via the	**FTP_ITC.1/SCA DTBS** **FTP_ITC.1.1/SCA DTBS** The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. **FTP_ITC.1.2/SCA DTBS**

trusted channel. **FTP_ITC.1.3** The TSF shall initiate communication via the trusted channel for [assignment: *list of functions for which a trusted channel is required*]. Hierarchical to: No other components Dependencies: No dependencies Management: a) Configuring the actions that require trusted channel, if supported Audit: a) Minimal: Failure of the trusted channel functions b) Minimal: Identification of the initiator and target of failed trusted channel functions c) Basic: All attempted uses of the trusted channel functions d) Basic: Identification of the initiator and target of all trusted channel functions	The TSF shall permit [**the TSF**] to initiate communication via the trusted channel. **FTP_ITC.1.3/SCA DTBS** The TSF **or the TOE** shall initiate communication via the trusted channel for [**signing DTBS-representation by means of the SSCD**].
FTP_TRP **Trusted Path**	
FTP_TRP.1 **Trusted Path**	
FTP_TRP.1.1 The TSF shall provide a communication path between itself and [selection: *remote, local*] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure. **FTP_TRP.1.2** The TSF shall permit [selection: *the TSF, local users, remote users*] to initiate communication via the trusted path. **FTP_TRP.1.3** The TSF shall require the use of the trusted path for [selection: *initial user authentication*, [assignment: *other services for which trusted path is required*]]. Hierarchical to: No other components Dependencies: No dependencies Management: a) Configuring the actions that require trusted path, if	**FTP_TRP.1/SCA** **FTP_TRP.1.1/SCA** The TSF shall provide a communication path between itself and [**local**] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure. **FTP_TRP.1.2/SCA** The TSF shall permit [***local users***] to initiate communication via the trusted path. **FTP_TRP.1.3/SCA** The TSF shall require the use of the trusted path for [***none***].

supported Audit: a) Minimal: Failures of the trusted path functions b) Minimal: Identification of the user associated with all trusted path failures, if available c) Basic: All attempted uses of the trusted path functions d) Basic: Identification of the user associated with all trusted path invocations, if available	

5.2.2 Security Requirements for the Non-IT-Environment

The following section covers the security requirements specified for the Non-IT-environment of the TOE. Only the TOE´s dedicated SIG Application is affected.

The specific security requirements for the Non-IT-environment of the TOE are defined according to /PP SSCD Type3/, chap. 5.4, with the following exception: the new security requirement R.Trusted_Environment has been added according to the extension of the Protection Profile concerning the establishment of trusted channels / paths for the communication between the TOE and a SCA. Furthermore, a specific security requirement related to the personalisation of the TOE´s dedicated SIG Application is added.

R.Trusted_Environment Trusted Environment for SCA and TOE

In the case that a trusted channel resp. trusted path between the TOE and the SCA by cryptographic means is not established the environment for the TOE usage shall be secured with the target to keep confidentiality and integrity of the VAD and integrity of the DTBS.

R.SIG_PERS Security of the Personalisation Process for the SIG Application

The originator of the personalisation data and the personalisation center responsible for the personalisation of the TOE´s dedicated SIG Application shall handle the personalisation data in an adequate secure manner. This concerns especially the security data to be personalised as secret cryptographic keys and PINs. The storage of the personalisation data at the originator and at the personalisation center as well as the transfer of these data between the different sites shall be conducted with respect to data integrity, authenticity and confidentiality.

Furthermore, the personalisation center shall treat the data for securing the personalisation process, i.e. the personalisation keys suitably secure.

It is in the responsibility of the originator of the personalisation data to garantuee for a sufficient quality of the personalisation data, especially of the cryptographic material to be personalised. The preparation and securing of the personalisation data appropriate to the card´s structure and according to the TOE´s personalisation requirements shall be as well in the responsibility of the external world and shall be done with care.

6 TOE Summary Specification

6.1 TOE Security Functions

6.1.1 TOE Security Functions / TOE-IC

For the definition of the TOE Security Functions (TSF) related to the TOE-IC refer to /ST-MIC30/, chap. 6.1.1.

The TSFs defined for the TOE-IC cover the following functions which are relevant for the TOE: F.RNG, F.HW_DES, F.OPC, F.PHY, F.LOG, F.COMP, F.MEM_ACC, F.SFR_ACC.

6.1.2 TOE Security Functions / TOE-ES

The following section gives a survey of the TSFs of the TOE´s Smartcard Embedded Software. All TSFs of /ST-MIC30/, chap. 6.1.2 are overtaken without any change except the TSF F.ACS which is suitably extended and replaced by the new TSF F.ACS_SFP, the TSF F.IA_PWD which is supplemented by a reference to the specification /HPC-SMC2/ and the TSF F.GEN_DIGSIG which is extended for the explicit generation of digital signatures to further hash algorithms.

TOE Security Functions / TOE-ES	
Access Control	
F.ACS_SFP	**Security Attribute Based Access Control**
	The TSF enforces the SFPs HPC Access Control, SIG Access Control and SIG Personalisation as defined in chap. 5.1.1.2 und 5.1.1.3. The TSF extends the TSF F.ACS of /ST-MIC30/, chap. 6.1.2. The TSF controls the access to data stored in the TOE and to functionality provided by the TOE. The access control is realised by usage of access rules as security attributes. Access to a DF, an EF, a key, a PIN or other user data is only possible if the related access rule is fulfilled. In particular, the TSF checks prior to command execution if the command specific requirements concerning user authentication and secure communication are satisfied. For SIG Access Control, the TSF covers especially the following functionality: • The TSF manages the following security attributes: - For subject User: General Attribute “Role” (Administrator, Signatory), Initialisation Attribute “SCD/SVD Management” (authorised, not authorised) - For object SCD: “SCD Operational” (no, yes) - For object DTBS: “Sent by an authorised SCA” (no, yes)

	• The user with the security attribute “Role” set to “Administrator” or set to “Signatory” is allowed to export the SVD. Establishment and usage of a trusted channel for the export of the SVD is required. • The user with the security attribute “Role” set to “Administrator” or set to “Signatory” is allowed to generate the SCD/SVD pair if the security attribute “SCD / SVD management” is set to “authorised”. • The user with the security attribute “Role” set to “Signatory” is allowed to create electronic signatures if the security attributes “Sent by an authorised SCA” and “SCD operational” are both set to “yes”. This is only allowed during the end-usage phase of the TOE. • Establishment of a trusted path or trusted channel is allowed prior to identification and authentication of the user. Other TSF mediated actions explicitly require a preceding successful authentication. • The user with the security attribute “Role” set to “Signatory” is allowed to enable the signature-creation function. Required is a preceding authentication of the Signatory. • The user with the security attribute “Role” set to “Signatory” is allowed to modify the security attribute “SCD operational”. • The user with the security attribute “Role” set to “Signatory” is allowed to modify RAD. • The user with the security attribute “Role” set to “Administrator” is allowed to modify the security attribute “SCD/SVD management”. • The user with the security attribute “Role” set to “Administrator” is allowed to create the RAD. This is only allowed during the personalisation phase of the TOE. • The TSF provides an authentication mechanism for the Administrator. • The user with the security attribute “Role” set to “Administrator” is allowed to perform a secure modification of the security attributes “Role” and “SCD/SVD management”. • The Security Attribute “SCD operational” is set to “no” after generation of the SCD. The user with the security attribute “Role” set to “Administrator” is allowed to specify an alternative value. • The SVD is exported without associated security attributes.
Identification and Authentication	
F.IA_AKEY	**Key Based User / TOE Authentication Based on Asymmetric Cryptography**
	The TSF provides the functionality of a key based external and internal authentication on the base of asymmetric cryptography. By an external authentication, users of the TOE can be authenticated with regard to the TOE. Vice versa, by an internal authentication, the TOE itself can be authenticated with regard to the external world. Both authentication mechanisms base on a challenge-response procedure using random numbers. The TSF enforces the following different internal and external authentication mechanisms: - Internal authentication without session key agreement according to /ISO 9796-2/, /HPC-SMC1/, chap. 11, Annex E.2, E.3, /eHC1/, chap. 10, Annex E.2, E.3 - External authentication without session key agreement according to /ISO 9796-2/, /HPC-SMC1/, chap. 11, Annex E.2, E.3, /eHC1/, chap. 10, Annex E.2, E.3 - Internal authentication including one step of session key and send sequence

	counter agreement according to /ISO 9796-2/, /HPC-SMC1/, chap. 11, Annex E.2, E.3, /eHC1/, chap. 10, Annex E.2, E.3, /eHC2/, chap. 3.6 - External authentication including one step of session key and send sequence counter agreement according to /ISO 9796-2/, /HPC-SMC1/, chap. 11, Annex E.2, E.3, /eHC1/, chap. 10, Annex E.2, E.3, /eHC2/, chap. 3.6 - Internal authentication according to /HPC-SMC1/, chap. 11, Annex E.6, /eHC1/, chap. 10, Annex E.6 Note: Each external authentication process requires a preceding Get Challenge – operation. The private and public keys necessary on the card´s side for authentication purposes are either generated on-card (with support by the TSF F.RSA_KEYGEN) or imported during the initialisation, personalisation or end-usage phase of the TOE. In particular, the import of public keys can be performed in the form of CV certificates what is connected with the verification of the respective CV certificate under usage of the TSF F.VER_DIGSIG. In each case, the keys involved on the card´s side in the authentication processes have to be explicitly referenced prior to their usage. The access to the keys necessary for the authentication processes is controlled by the specific SFP which is defined for the respective application using the authentication keys. The execution of the specific SFP is task of the TSF F.ACS_SFP for access control. In the case of a successful external authentication attempt the TSF sets a corresponding actual security state for key based user authentication. The TSF makes use of asymmetric cryptography with generation and verification of RSA digital signatures resp. RSA encryption and decryption and is therefore directly connected with the TSF F.CRYPTO. Depending on the type of authentication mechanism, the combination of a successful internal and external authentication process can include the generation of session keys (incl. send sequence counter). Depending on the type of authentication mechanism, the TSF stores the generated session keys volatile and on demand as well persistently on the card. The generated keys can be used for securing the following data exchange between the TOE and the external world (in the current or a later session) with the objective of data confidentiality and data integrity and authenticity (Secure Messaging). In addition, as well depending on the type of authentication mechanism, the generated keys can be used further on for authentication processes based on symmetric cryptography.
F.IA_SKEY	**Key Based User / TOE Authentication Based on Symmetric Cryptography**
	The TSF provides the functionality of a key based external and internal authentication on the base of symmetric cryptography. By an external authentication, users of the TOE can be authenticated with regard to the TOE. Vice versa, by an internal authentication, the TOE itself can be authenticated with regard to the external world. Both authentication mechanisms base on a challenge-response procedure using random numbers. The TSF enforces the following different internal and external authentication mechanisms: - Internal authentication with / without individual key derivation and without session key generation according to /HPC-SMC1/, chap. 11, Annex E.4, /eHC1/, chap.10, Annex E.4, /ISO 9796-2/ - External authentication with / without individual key derivation and without session key generation according to /HPC-SMC1/, chap. 11, Annex E.4, /eHC1/, chap.10,

	Annex E.4, /ISO 9796-2/ - Mutual authentication with / without individual key derivation and without session key generation according /HPC-SMC1/, chap. 11, Annex E.4, /eHC1/, chap.10, Annex E.4, /ISO 9796-2/ - Internal authentication with / without individual key derivation and including the first step of session key and send sequence counter generation according to /HPC-SMC1/, chap. 11, Annex E.4, /eHC1/, chap.10, Annex E.4, /eHC2/, chap. 3.7, /ANSI X9.63/, /ISO 9796-2/ - External authentication with / without individual key derivation and including the last step of session key and send sequence counter generation according to /HPC-SMC1/, chap. 11, Annex E.4, /eHC1/, chap.10, Annex E.4, /eHC2/, chap. 3.7, /ANSI X9.63/, /ISO 9796-2/ - Mutual authentication with / without individual key derivation and including session key and send sequence counter generation according to /HPC-SMC1/, chap. 11, Annex E.4, /eHC1/, chap.10, Annex E.4, /eHC2/, chap. 3.7, /ANSI X9.63/, /ISO 9796-2/ Note: Each external authentication process requires a preceding Get Challenge – operation. The symmetric keys necessary on the card´s side for the authentication mechanisms can either be generated on-card by a derivation process for deriving individual keys before the main authentication process starts. This key derivation process is performed by the TSF F.CRYPTO. Alternatively, symmetric keys imported during the initialisation, personalisation or end-usage phase of the TOE or agreed within a preceding authentication process can be used. The access to the keys necessary for the authentication processes is controlled by the specific SFP which is defined for the respective application using the authentication keys. The execution of the specific SFP is task of the TSF F.ACS_SFP for access control. In the case of a successful external authentication attempt the TSF sets a corresponding actual security state for key based user authentication. The TSF makes use of symmetric cryptography with DES based encryption, decryption, MAC generation resp. MAC verification. Hence, the TSF F.IA_SKEY is directly connected with the TSF F.CRYPTO. Depending on the type of authentication mechanism, the combination of a successful internal and external authentication process can include the generation of session keys (incl. send sequence counter). Depending on the type of authentication mechanism, the TSF stores the generated session keys volatile and on demand as well persistently on the card. The generated keys can be used for securing the following data exchange between the TOE and the external world (in the current or a later session) with the objective of data confidentiality and data integrity and authenticity (Secure Messaging). In addition, as well depending on the type of authentication mechanism, the generated keys can be used further on for authentication processes based on symmetric cryptography.
F.IA_PWD	**Password Based User Authentication**
	Users of the TOE can be authenticated (towards the TOE) by means of a card holder authentication process. For the card holder authentication process, the TSF compares the card holder verification information, here a password (PIN), provided by a subject with a corresponding secret reference value stored permanently on the card. The TSF uses for the authentication process the password referenced by the external world. The access to the relevant password resp. its reference value is controlled by the specific SFP which

is defined for the respective application using the password. The execution of the specific SFP is task of the TSF F.ACS_SFP for access control.

The card holder authentication process can be performed by usage of the command Verify or Change Reference Data (whereat the latter command makes a password change possible).

Each password used for authentication purposes is connected with an own error usage counter and an own usage counter. Furthermore, each password is connected with an own resetting code (PUK) whereat the resetting code itself is connected with an own usage counter (but no error usage counter).

The number of applications of a password for authentication purposes with the command Verify is limited by its usage counter. The TSF allows at maximum for a number of authentication attempts with a password as restricted by its usage counter. The value for the usage counter can be predefined as infinite, i.e. the password can be used without any limit. A password with an expired usage counter cannot be longer used for authentication purposes with the command Verify (but with the command Change Reference Data).

In the case of a password with a finite usage counter, each authentication attempt with the command Verify decrements the usage counter of the password, independently whether the authentication attempt succeeds or fails. A successful authentication attempt with the command Change Reference Data re-initialises the usage counter to its predefined initial value.

The TSF detects for a password when a predefined number of consecutive unsuccessful authentication attempts occurs related to the card holder authentication process. Each consecutive unsuccessful comparison of the presented password with the reference value stored on the card is recorded by the TSF in order to limit the number of further authentication attempts with this password.

In the case of a successful authentication attempt a corresponding actual security state for the password is set and the error usage counter of the password is re-initialised to its predefined initial value.

If an authentication attempt with the password fails, the corresponding actual security state is reset and the error usage counter of the password is decreased. When the defined maximum number of unsuccessful authentication attempts has been met or surpassed, the TSF blocks the corresponding password for any further authentication attempt.

A password with an expired error usage counter can be unblocked by usage of the related resetting code, provided that the usage counters of the password and of the resetting code are not expired. Otherwise, there is no way to unblock the password so that this password is invalid for each further authentication attempt.

The unblocking of a blocked password can be performed by usage of the command Reset Retry Counter only. In the case of a successful authentication attempt with the resetting code related to the blocked password, the expired error usage counter is re-initialised to its initial value (as well as for the usage counter of the password) and hence, the password can be used further on for authentication attempts.

The number of applications of a resetting code for authentication purposes is limited by its usage counter. The TSF allows at maximum for a number of authentication attempts with the resetting code as restricted by its usage counter. Each unblocking attempt with the command Reset Retry Counter decrements the usage counter of the resetting code, independently whether the authentication attempt with the resetting code succeeds or fails. The unblocking process for a blocked password can be combined with a change of this password. However, even if the command Reset Retry Counter resp. the authentication

	with the resetting code succeeds, the actual security state for the password will not be set. For security reasons, a password shall be connected with an error usage counter with a sufficiently small value as initial value. Furthermore, the usage of the related resetting code itself shall be limited by an usage counter with a sufficiently small initial value. In general, a security state set due to a successful authentication attempt can be valid for several following TOE commands. However, as well, it is possible to restrict the validity of such an authentication state to one single following TOE command, i.e. after the next command has accessed this security state it will be reset by the TSF. The TSF does not check the quality of passwords or resetting codes used. The sufficient quality of passwords and resetting codes lies in the responsibility of the external world only. The transfer of passwords and resetting codes to the TOE can be executed in unsecured mode, i.e. without usage of Secure Messaging, or alternatively in secured mode, i.e. with usage of Secure Messaging. In the latter case, the TSFs F.EX_CONF and F.EX_INT are involved. For the TOE´s HPC Application and SIG Application, the concrete usage of PIN and PUK, in particular the definition of error usage counters and usage counters and their initial values, the (minimal) lengths of PIN and PUK and the access to the commands Verify, Change Reference Data and Reset Retry Counter is regulated by the specification /HPC-SMC2/.
Integrity of Stored Data	
F.DATA_INT	**Stored Data Integrity Monitoring and Action**
	The TSF monitors data stored within the TOE for integrity errors. This concerns all - DFs - EFs - Passwords incl. related attributes - Cryptographic keys incl. related attributes - Security critical data stored within the card and channel context (session keys incl. attributes, status information as actual security states for key and password based authentication, hash values, further security relevant card and channel information) The monitoring is based on the following attributes: - Checksum (CRC) attached to the header of a file - Checksum (CRC) attached to the data body of a file - Checksums (CRC) attached to each secret (password, cryptographic key) and its related attributes stored in the EEPROM - Checksums (CRC) attached to card and channel context related security critical information Each access of the TOE to a DF, to an EF, to a secret (password or cryptographic key incl. its related attributes) or to security critical card resp. channel context data the TSF is secured with an integrity check on base of the mentioned attributes. Upon detection of a

	data integrity error, the TSF informs the user about this fault (output of a warning). If the checksum of the header of a file has been detected as corrupted, the data contained in the affected file are no longer accessible. If the data contained in a file are not of integrity, the affected data will be treated in the following way: - For the Read access, the affected data will be exported, but the data export will be connected with a warning. - For the Update access, the integrity error of the affected data will be ignored, and the data imported by the command will be stored and a new checksum will be computed. - For all remaining access modes, the affected data will not be used for data processing. If a secret (password, cryptographic key) and its related attributes are corrupted, the secret and its related data will not be processed. If security critical card or channel context data are not of integrity, the Smartcard Embedded Software immediately jumps into an endless-loop (re-activation by reset possible).
Data Exchange	
F.EX_CONF	**Confidentiality of Data Exchange**
	The TSF provides the capability to ensure that secret data which is exchanged between the TOE and the external world remains confidential during transmission. For this purpose, encryption based on symmetric cryptography is applied to the secret data. The TSF ensures that the user and the user data's access condition have indicated confidentiality for the data exchange. Securing the data transfer with regard to data confidentiality is done by Secure Messaging according to the standard ISO/IEC 7816-4. The cryptographic key used for securing the data transfer is either a symmetric session or static key. In case of a session key, the key is negotiated during a preceding mutual authentication process (based on a random challenge and response procedure) between the TOE and the external world (realised by the TSFs F.IA_SKEY, F.IA_AKEY, F.CRYPTO). For encryption and decryption, the TSF makes use of the TSF F.CRYPTO for DES functionality.
F.EX_INT	**Integrity and Authenticity of Data Exchange**
	The TSF provides the capability to ensure that data which is exchanged between the TOE and the external world remains integer and authentic during transmission. For this purpose, cryptographic checksums based on symmetric cryptography are applied to the data. The TSF ensures that the user and the user data's access condition have indicated integrity and authenticity for the data exchange. Securing the data transfer with regard to data integrity and authenticity is done by Secure Messaging according to the standard ISO/IEC 7816-4. The cryptographic key used for securing the data transfer is either a symmetric session or

	static key. In case of a session key, the key is negotiated during a preceding mutual authentication process (based on a random challenge and response procedure) between the TOE and the external world (realised by the TSFs F.IA_SKEY, F.IA_AKEY, F.CRYPTO). For checksum securing and verification, the TSF makes use of the TSF F.CRYPTO for DES functionality.
Object Reuse	
F.RIP	**Residual Information Protection**
	The TSF ensures that any previous information content of a resource is explicitly erased upon the deallocation of the resource used for any of the following components: - All volatile and non-volatile memory areas used for operations in which security relevant material (as e.g. cryptographic data, passwords or other security critical data) is involved. Explicit erasure is defined as physical erasure.
Protection	
F.FAIL_PROT	**Hardware and Software Failure Protection**
	The TSF preserves a secure operation state of the TOE when the following types of failures and attacks occur: - HW and/or SW induced reset - Power supply cut-off - Power supply variations - Unexpected abortion of the execution of the TSF due to external or internal events (in particular, break of a transaction before completion) - System breakdown - Internal HW and/or SW failure - Manipulation of executable code - Corruption of status information (as e.g. card status information, object life cycle state, actual security state related to key and password based authentication, ...) - Environmental stress - Input of inconsistent or improper data - Tampering - Manipulation resp. insufficient quality of the HW-RNG The TSF makes use of HW and SW based security features and corresponding mechanisms to monitor and detect induced HW and SW failures and tampering attacks. In particular, the TSF is supported by the IC specific TSFs F.OPC and F.PHY. Upon the detection of a failure of the above mentioned type the TSF reacts in such a way that the TSP is not violated. The TOE changes immediately to a locked state and cannot be used any longer within the actual session. Depending on the type of the detected attack to the underlying IC (incl. its Dedicated Software) or to the Smartcard Embedded Software code the TOE will be irreversible locked resp. can be reactivated by a reset.

F.SIDE_CHAN	**Side Channel Analysis Control**
	The TSF provides suitable HW and SW based mechanisms to prevent attacks by side channel analysis like Simple Power Analysis (SPA), Differential Power Analysis (DPA), Differential Fault Analysis (DFA) and Timing analysis (TA). The TSF ensures that all countermeasures available are used in such a way that they support each other. In particular, the TSF is supported by the TSF F.LOG of the underlying IC and its Dedicated Support Software. The TSF acts in such a manner that all security critical operations of the TOE, in particular the TOE´s cryptographic operations, are suitably secured by these HW and SW countermeasures. The TSF guarantees that information on IC power consumption, information on command execution time and information on electromagnetic emanations do not lead to useful information on processed security critical data as secret cryptographic keys or passwords. In particular, the IC contacts as Vcc, I/O and GND or the IC surface do not make it possible for an attacker to gain access to security critical data as secret cryptographic keys or passwords. The TSF enforces the installation of a secure session before any cryptographic operation is started. In particular, the installation of a secure session does not only concern the core cryptographic operation itself. All preparing security relevant actions performed prior to the core cryptographic operation as e.g. the generation of session keys, the process of loading keys into the dedicated IC cryptographic modules and the data preparation as reformatting or padding are involved as well. Furthermore, the secure session covers all security relevant actions which follow the core cryptographic operation as e.g. the post-processing of the output data.
F.SELFTEST	**Self Test**
	The TSF covers different types of self tests whereat each self test consists of a check of a dedicated integrity attribute related to (parts of) the TOE´s code resp. data. The TSF integrates self tests with the following objectives: The TSF provides the capability of conducting a self test during initial start-up, i.e. after each reset, to demonstrate the correct operation of its TSFs. This self test is performed automatically by the TOE and consists of the verification of the integrity of any software code stored in the EEPROM area. Furthermore, the TSF provides authorised users - here the Smartcard Embedded Software of the TOE (TOE-ES) itself - with the capability to verify the integrity of TSF data during run-time. The self test is performed automatically by the TOE and is supported by the TSF F.DATA_INT. Additionally, the TSF provides authorised users with the capability to verify the integrity of stored TSF executable code. This concerns only the production phase, more precise the initialisation phase of the TOE (phase 5 of the product´s life cycle). Prior to the initialisation of the TOE, the ROM-code of the TOE can be verified on demand by the Smartcard Embedded Software developer. The integrity of the whole EEPROM-code is checked automatically by the TOE during the storage of the initialisation file in the framework of the TOE´s initialisation. These self tests are supported by the TSF F.CRYPTO (SHA-1 hash value calculation, MAC verification). The TSF supports all other TSFs defined for the Smartcard Embedded Software (TOE-ES).

Cryptographic Operations	
F.CRYPTO	**Cryptographic Support**
	The TSF provides cryptographic support for the other TSFs using cryptographic mechanisms. The TSF supports: - DES/3DES algorithm according to the standard /FIPS 46-3/ resp. /ANSI X9.52/ with a key length of 56 resp. 112 bit entropie (used for encryption, decryption, MAC generation and verification according to /FIPS 46-3/, /ANSI X9.52/, /ANSI X9.19/, /HPC-SMC1/, chap. 11, 4.1, /eHC1/, chap. 10, 3.1.1) - RSA core algorithm according to the standard /PKCS1/ with key lengths of 1024, 1280, 1536, 1792 resp. 2048 bit modulus lengths (used for RSA encryption, decryption, signature generation and verification, see other TSFs related to RSA based mechanisms) - Random number generation by a deterministic RNG (incl. online-test of the HW-RNG for seeding the SW-RNG) - SHA-1 hash value calculation according to /ALGCAT/, chap. 2 resp. /FIPS 180-2/ - Negotiation of 3DES session keys - Derivation of individual 3DES keys according to the standard /ISO 10118-2/ (including a H2 hash value calculation and DES calculations) The resistance of the TSF against SPA, DPA, DFA and TA is part of the TSF F.SIDE_CHAN. The random number generation is in particular used for RSA and DES key generation and authentication mechanisms. The mechanism for the generation of session keys is directly connected with the TSFs F.IA_AKEY and F.IA_SKEY which realise internal and external authentication processes. Furthermore, the generation of random numbers of high quality, and depending on the authentication type, the SHA-1 hash value calculation of TSF F.CRYPTO are involved. The mechanism for the derivation of individual keys makes use of the SHA-1 hash value calculation and DES based calculations of the TSF F.CRYPTO. The TSF is directly supported by the TSFs of the underlying IC which supply cryptographic functionality.
F.RSA_KEYGEN	**RSA Key Pair Generation**
	The TSF generates RSA key pairs with key lengths of 1024, 1280, 1536, 1792 resp. 2048 bit modulus length for asymmetric cryptography which can be used later on e.g. for digital signatures or authentication purposes. The TSF enforces the key pair generation process and the related key material to meet the following requirements: - The RSA key pair generation process follows a well-designed key generation algorithm of sufficient quality; in particular, the requirements for RSA keys and their generation in /ALGCAT/, chap. 3.1 and 4 as well as in the corresponding European algorithm paper, chap. 4.5.2, 4.6, Annex C.2 and C.3 are taken into account. - Random numbers used in the key pair generation process for the generation of the primes are of high quality to ensure that the new key pair is unpredictable and

	unique with a high probability. - The generation of the random numbers necessary for the primes is performed by usage of a deterministic RNG running on the TOE. - Prime numbers produced in the key pair generation process are unique with a high probability and satisfy the requirements in /ALGCAT/, chap. 3.1 and 4. In particular, the so-called epsilon-condition is considered. - The primes are independently generated. - Sufficiently good primality tests with convincing limits are implemented to guarantee with a high probability for the property of the generated prime candidates to be prime. In particular, the actual version of the significance limit for primality tests is considered. - In the key pair generation process, for the public exponent given by the external world the corresponding private exponent is calculated and converted into its CRT parameters. - For each key length, the generated key pairs show a “good” distribution within the key range; in particular, the generated new key pair is unique with a high probability. - Only cryptographically strong key pairs with the intended key length are generated. In particular, for any generated key pair, the private key cannot be derived from the corresponding public key. - The key pair generation process includes a dedicated check if the generated private and public key match; only valid key pairs are issued. - During the key pair generation process, it is not possible to gain information about the chosen random numbers, about the calculated primes, about other secret values which will be used for the key pair to be generated or about the generated key pair and its parts itself. - During the key pair generation process, it is not possible to gain information about the design of the routines realising the key pair generation. - The key pair generation process includes a physical destruction of the old private key part before the new key pair is generated. The resistance of the TSF against SPA, DPA, DFA and TA is part of the TSF F.SIDE_CHAN. The TSF makes use of the TSF F.CRYPTO for random number generation and RSA signature generation and verification. The public part of the generated key pair can be exported with an authentication attribute which either can be a MAC (generation supported by the TSF F.CRYPTO) or a digital signature (generation supported by the TSF F.GEN_DIGSIG) over the public key data.
F.GEN_DIGSIG	**RSA Generation of Digital Signatures**
	The TSF provides a digital signature functionality based on asymmetric cryptography, particularly based on the RSA algorithm with key lengths of 1024, 1280, 1536, 1792 resp. 2048 bit modulus length. The TSF digital signature function will be used for several purposes with different formats for the digital signature input: - Explicit generation of digital signatures using the signature scheme with appendix according to the standard /PKCS1/, chap. 8.2.1 and with hash algorithm SHA-1, SHA-2 (224, 256, 384 resp. 512 bit) resp. RIPEMD160 (external hash value calcu-

	lation), see /HPC-SMC1/, chap. 11, /eHC1/, chap. 10 - Explicit generation of digital signatures using the signature scheme with appendix according to the standard /ISO 9796-2/ with random number based on the hash algorithm SHA-1, SHA-2 (224, 256, 384 resp. 512 bit) resp. RIPEMD160 (external hash value calculation), see /HPC-SMC1/, chap. 11, /eHC1/, chap. 10 - Implicit generation of digital signatures within authentication mechanisms for the creation of authentication tokens using the signature scheme with message recovery according to the standard /ISO 9796-2/ based on the hash algorithm SHA-1, see /HPC-SMC1/, chap. 11, Annex E.2, E.3, /eHC1/, chap. 10, Annex E.2, E.3 - Implicit generation of digital signatures within authentication mechanisms for the creation of authentication tokens using the signature scheme with message recovery according to the standard /PKCS1/, chap. 8.2.1 without hash and OID, but with an additional limitation of the length of the input message, see /HPC-SMC1/, chap. 11, Annex E.6, /eHC1/, chap. 10, Annex E.6 The TSF function for generation of a digital signature uses the private key which has been referenced before. The random numbers necessary for the padding of the data within the signature process are generated by using the TSF F.CRYPTO for random number generation. Furthermore, for the signature calculation itself, the TSF makes use of the TSF F.CRYPTO, and the computation of hash values is as well based on the TSF F.CRYPTO. Each private key used for the signature generation function is either generated on-card by usage of the TSF F.RSA_KEYGEN or is generated by the external world and loaded onto the card during the initialisation, personalisation or end-usage phase of the TOE. In the latter case, it is in the responsibility of the external world to guarantee for a sufficient cryptographic strength of the private key and to handle the private key outside the card in a sufficient secure manner. The resistance of the TSF against SPA, DPA, DFA and TA is part of the TSFs F.Log and F.SIDE_CHAN. For each private key - generated on-card or imported with the assumption that the external world meets the requirements on the key handling as defined before - the TSF digital signature function works in such a manner that the private key cannot be derived from the signature and the signature cannot be generated by other individuals not possessing that secret. Furthermore, the TSF digital signature function works in such a manner that no information about the private key can be disclosed during the generation of the digital signature.
F.VER_DIGSIG	**RSA Verification of Digital Signatures**
	The TSF provides a functionality to verify digital signatures based on asymmetric cryptography, particularly based on the RSA algorithm with key lengths of 1024, 1280, 1536, 1792 resp. 2048 bit modulus length. The TSF function to verify a digital signature will be used for several purposes with different formats for the digital signature input: - Implicit verification of digital signatures within authentication mechanisms for the verification of authentication tokens using the signature scheme with message recovery according to the standard /ISO 9796-2/ based on the hash algorithm SHA-1, see /HPC-SMC1/, chap. 11, Annex E.2, E.3, /eHC1/, chap. 10, Annex E.2, E.3 - Implicit verification of digital signatures within the verification and unwrapping of imported CV certificates using the signature scheme with message recovery according to the standard /ISO 9796-2/ based on the hash algorithm SHA-1, see /HPC-SMC1/, Annex B, /eHC1/, chap. 10, Annex B

	The TSF function to verify a digital signature uses the public key which has been referenced before. For the verification mechanism itself, the TSF makes directly use of the TSF F.CRYPTO, and the computation of hash values is as well based on the TSF F.CRYPTO. Each public key used for the function to verify a digital signature is either generated on-card by usage of the TSF F.RSA_KEYGEN or is generated by the external world and loaded onto the card during the initialisation, personalisation or end-usage phase of the TOE. In particular, loading via a CV certificate by a suitable preceding operation is possible.
F.RSA_ENC	**RSA Encryption**
	The TSF provides a functionality to encrypt data based on asymmetric cryptography, particularly based on the RSA algorithm with key lengths of 1024, 1280, 1536, 1792 resp. 2048 bit modulus length. The TSF encryption function will be used for several purposes with different formats for the encryption input: - Explicit encryption of a plain text using the “encryption scheme” with formatted plain message according to the standard /PKCS1/, chap. 7.2.1 and with hash algorithm SHA-1, see /HPC-SMC1/, chap. 11, 4.1, /eHC1/, chap. 10, 3.1.1 - Implicit encryption within authentication mechanisms for the generation of authentication tokens using the “encryption primitive” according to the standard /PKCS1/, chap. 5.1.1 The TSF encryption function uses the public key which has been referenced before. For the encryption mechanism itself, the TSF makes directly use of the TSF F.CRYPTO. Each public key used for the encryption function is either generated on-card by usage of the TSF F.RSA_KEYGEN or is generated by the external world and loaded onto the card during the initialisation, personalisation or end-usage phase of the TOE. In particular, loading via a CV certificate by a suitable preceding operation is possible.
F.RSA_DEC	**RSA Decryption**
	The TSF provides a functionality to decrypt data based on asymmetric cryptography, particularly based on the RSA algorithm with key lengths of 1024, 1280, 1536, 1792 resp. 2048 bit modulus length. The TSF decryption function will be used for several purposes with different formats for the data supplied within the cryptogram: - Explicit decryption of a cryptogram using the “decryption scheme” with formatted input according to the standard /PKCS1/, chap. 7.2.2 and with hash algorithm SHA-1, see /HPC-SMC1/, chap. 11, 4.1, /eHC1/, chap. 10, 3.1.1 - Implicit decryption within authentication mechanisms for the verification of authentication tokens using the “decryption primitive” according to the standard /PKCS1/, chap. 5.1.2 The TSF decryption function uses the private key which has been referenced before. For the decryption mechanism itself, the TSF makes directly use of the TSF F.CRYPTO. Each private key used for the decryption function is either generated on-card by usage of the TSF F.RSA_KEYGEN or is generated by the external world and loaded onto the card

	during the initialisation, personalisation or end-usage phase of the TOE. In the latter case, it is in the responsibility of the external world to guarantee for a sufficient cryptographic strength of the private key and to handle the private key outside the card in a sufficient secure manner. The resistance of the TSF against SPA, DPA, DFA and TA is part of the TSFs F.Log and F.SIDE_CHAN. For each private key - generated on-card or imported with the assumption that the external world meets the requirements on the key handling as defined before - the TSF decryption function works in such a manner that the private key cannot be derived from the cryptogram and the cryptogram cannot be deciphered by other individuals not possessing that secret. Furthermore, the TSF decryption function works in such a manner that no information about the private key may be disclosed during the decipherment of the cryptogram.

6.2 SOF Claim for TOE Security Functions

According to Common Criteria, /CC 2.3 Part1/ and /CC 2.3 Part3/, all TOE Security Functions (TSF) which are relevant for the assurance requirement AVA_SOF.1 are identified in this section.

For the TSFs explicitly defined for the underlying IC, information on the SOF claim can be found in /ST-ICPhilips/.

The TSFs related to the complete product using mechanisms which can be analysed for their permutational or probabilistic properties and which contribute to AVA_SOF.1 are the following:

TOE Security Function	SOF Claim	Description / Explanation
F.ACS_SFP	Not applicable	The TSF is not realised by permutational or probabilistic mechanisms.
F.IA_AKEY	SOF high	The TSF implements under usage of the TSFs F.CRYPTO, parts for RSA operations, hash value calculation and random number generation, and of the TSFs F.GEN_DIGSIG, F.VER_DIGSIG, F.ENC and F.DEC cryptographic mechanisms for authentication. The TSF is realised by permutational and probabilistic mechanisms.
F.IA_SKEY	SOF-high	The TSF implements under usage of the TSFs F.CRYPTO, parts for DES operations and random number generation, cryptographic mechanisms for authentication. The TSF is realised by permutational and probabilistic mechanisms.
F.IA_PWD	SOF high	The TSF includes a probabilistic password mechanism for the authentication of the user.

F.DATA_INT	Not applicable	In general, the mechanisms for generating and checking CRC-checksums can be analysed with permutational or probabilistic methods. But these mechanisms are not relevant for AVA_SOF.1 as the securing of data areas by CRC-checksums is only intended to secure against *accidental* data modification.
F.EX_CONF	Not applicable	The TSF includes cryptographic mechanisms using DES functionality from the TSF F.CRYPTO. Refer to the explanations for F.CRYPTO concerning the SOF claim resp. valuation of DES based encryption / decryption functions.
F.EX_INT	Not applicable	The TSF includes cryptographic mechanisms using DES functionality from the TSF F.CRYPTO. Refer to the explanations for F.CRYPTO concerning the SOF claim resp. valuation of DES based MAC generation / MAC verification functions.
F.RIP	Not applicable	The TSF is not realised by permutational or probabilistic mechanisms.
F.FAIL_PROT	Not applicable	The TSF is not realised by permutational or probabilistic mechanisms.
F.SIDE_CHAN	Not applicable	The TSF is not realised by permutational or probabilistic mechanisms.
F.SELFTEST	Not applicable	The TSF is not realised by permutational or probabilistic mechanisms, except for the functionality supported by the TSFs F.DATA_INT and F.CRYPTO (→ refer to the SOF claim for these TSFs).
F.CRYPTO	SOF high	The TSF includes cryptographic algorithms SHA-1, RSA with key lengths 1024, 1280, 1536, 1792 and 2048 bit modulus length as well as random number generation by usage of a deterministic RNG of quality class K4. These algorithms and key lengths defined for the TSF comply with the requirements in /ALGCAT/, chap. 2, 3.1, 4 for qualified electronic signatures and fulfill therefore the requirements for SOF high. The TSF part concerning DES functionality (used for encryption, decryption, MAC generation and MAC verification) are as well assigned to the SOF claim as permutational and probabilistic mechanisms are involved. The negotiation of session keys and the derivation of individual keys is not considered to part for the SOF analysis.
F.RSA_KEYGEN	SOF high	The TSF includes permutational and probabilistic mechanisms for the key generation process itself as well as for the integrated random number generation and key check. In particular, functionality from the TSF F.CRYPTO (random number generation, RSA signature generation and verification) is used by this TSF.
F.GEN_DIGSIG	SOF high	The TSF implements under usage of the TSF F.CRYPTO, parts for RSA operations and random number generation, cryptographic mechanisms for signature generation. The TSF is realised by permutational and probabilistic mecha-

		nisms, in particular the quality of the implemented security mechanisms against leakage can be analysed using permutational or probabilistic methods.
F.VER_DIGSIG	Not applicable	The implementation of the TSF uses only public keys and needs not to be considered with regard to high attack potential so that securing of the implementations against Simple Power Analysis (SPA), Differential Power Analysis (DPA), Differential Fault Analysis (DFA) and Timing Attacks (TA) is not necessary. Because of this fact, the TSF – although it can be analysed with permutational or probabilistic methods - is not relevant for AVA_SOF.1. Nevertheless, this TSF is secured by appropriate hardware security features.
F.RSA_ENC	Not applicable	The implementation of the TSF uses only public keys and needs not to be considered with regard to high attack potential so that securing of the implementations against Simple Power Analysis (SPA), Differential Power Analysis (DPA), Differential Fault Analysis (DFA) and Timing Attacks (TA) is not necessary. Because of this fact, the TSF – although it can be analysed with permutational or probabilistic methods - is not relevant for AVA_SOF.1. Nevertheless, this TSF is secured by appropriate hardware security features.
F.RSA_DEC	SOF high	The TSF implements under usage of the TSF F.CRYPTO, part for RSA operations, cryptographic mechanisms for decryption. The TSF is realised by permutational and probabilistic mechanisms, in particular the quality of the implemented security mechanisms against leakage can be analysed using permutational or probabilistic methods.

For each of the TOE Security Functions given in the preceding list an explicit claim of “SOF-high” is made.

The TOE´s cryptographic algorithms themselves can also be analysed with permutational or probabilistic methods but this is not in the scope of CC evaluations.

6.3 Assurance Measures

Appropriate assurance measures will be employed by the developer of the TOE to satisfy the security assurance requirements defined in chap. 5.1.3. For the evaluation of the TOE, the developer will provide appropriate documents describing these measures and containing further information supporting the check of the conformance of these measures against the claimed assurance requirements.

For the Smartcard Embedded Software part of the TOE (TOE-ES), the following table gives a mapping between the assurance requirements and the documents containing the relevant information for the respective requirement. All these documents concerning the TOE-ES are provided by the developer of the TOE-ES. The table below contains only the directly related documents, references to further documentation can be taken from the mentioned documents.

Overview of Developer´s TOE-ES related Documents		
Assurance Class	**Family**	**Document containing the relevant information**
ACM **Configuration Management**	ACM_AUT	- Document Configuration Control System
	ACM_CAP	- Document Life-Cycle Model - Document Configuration Control System
	ACM_SCP	- Document Configuration Control System - Document Life-Cycle Model
ADO **Delivery and Operation**	ADO_DEL	- Document Life-Cycle Model
	ADO_IGS	- Document Installation, Generation and Start-Up Procedures
ADV **Development**	ADV_FSP	- Document Functional Specification
	ADV_HLD	- Document High-Level Design - Detailed development documents as system specifications, design specifications, etc.
	ADV_LLD	- Document Low-Level Design - Detailed development documents as system specifications, design specifications, etc.
	ADV_IMP	- Source Code - Detailed development documents as system specifications, design specifications, etc.
	ADV_RCR	- Document Functional Specification - Document High-Level Design - Document Low-Level Design
	ADV_SPM	- Document TOE Security Policy Model
AGD **Guidance Documents**	AGD_ADM, AGD_USR	- User Guidance for the Personaliser of the TOE - User Guidance for the User of the TOE´s MICARDO OS platform - User Guidance for the User of the TOE´s HPC and SIG Application
ALC **Life Cycle Support**	ALC_DVS	- Document Security of the Development Environment
	ALC_LCD	- Document Life-Cycle Model
	ALC_TAT	- Configuration List
ATE **Tests**	ATE_COV	- Document Test Documentation - Detailed test documentation as system test specifications, test protocols, etc.

	ATE_DPT	- Document Test Documentation - Detailed test documentation as system test specifications, test protocols, etc.
	ATE_FUN	- Document Test Documentation - Detailed test documentation as system test specifications, test protocols, etc.
	ATE_IND	- Samples of the TOE - Source Code
AVA Vulnerability Assessment	AVA_MSU	- Document Analysis of the Guidance Documents
	AVA_SOF	- Document TOE Security Function Evaluation
	AVA_VLA	- Document Vulnerability Analysis

As mentioned, the evaluation of the TOE will re-use evaluation results of the CC evaluation of the underlying IC "Philips SmartMX P5CC036V1D Secure Smart Card Controller" provided by Philips Semiconductors GmbH. Therefore, for the TOE-IC the following documents will be at least provided by the IC developer:

Overview of Developer´s TOE-IC related Documents	
Class	**Documents**
Security Target	Security Target of the IC evaluation, /ST-ICPhilips/
Evaluation Report	Evaluation Technical Report Lite (ETR Lite) of the IC evaluation, /ETRLite-ICPhilips/
Configuration List	Configuration List for composite evaluation with Sagem Orga GmbH, /ConfListPhilips/
User Guidances	User Guidance for the IC, /UG-ICPhilips/
	Data Sheet for the IC, /DS-ICPhilips/
	Instruction Set for the IC, /IS-ICPhilips/

7 PP Claims

The Security Target claims conformance to the Protection Profile /PP-HPC/. Furthermore, as outlined in chap. 1.3 the Security Target takes into account the contents of the Protection Profile /PP SSCD Type3/. More detailed information on the differences to the mentioned Protection Profiles can be found in the following chapters 7.1 (for /PP-HPC/) resp. 7.2 (for /PP SSCD Type3/).

7.1 TOE´s HPC Application

7.1.1 PP References

The Security Target for the TOE and its HPC Application is based on the Protection Profile /PP-HPC/.

No substantial differences to the Protection Profile /PP-HPC/ exist.

7.1.2 PP Changes and Supplements

All assets, assumptions, threats, security policies, security objectives, security requirements and security functional requirements for the TOE and its environment as defined in the Protection Profile /PP-HPC/ are taken over without any change.

7.2 TOE´s SIG Application

7.2.1 PP References

The Security Target for the TOE and its SIG Application is based on the Protection Profile /PP SSCD Type3/ for SSCDs of Type 3, i.e. for devices with oncard - generation of the SCD/SVD key pair, secure storage and usage of the SCD and secure creation of electronic signatures using the dedicated SCD key.

Only the following substantial differences to the Protection Profile /PP SSCD Type3/ exist:

- Communication between the TOE and the external SCA:

  The establishment of a trusted channel resp. trusted path for the communication between the TOE and the SCA as required within /PP SSCD Type3/ is now specified as optional. In the case that a trusted channel resp. trusted path is not used the cardholder resp. signatory is responsible for establishing a trusted environment for the communication between the TOE and the SCA.

For the impact of these extensions on assets, assumptions, threats, security policies, security objectives, security requirements and security functional requirements for the TOE and its environment defined resp. not-defined in /PP SSCD Type3/ refer to the following section.

7.2.2 PP Changes and Supplements

All assets, assumptions, threats, security policies, security objectives, security requirements and security functional requirements for the TOE and its environment as defined in the Protection Profile /PP SSCD Type3/ for SSCDs of Type 3 are taken over without any change, except the following changes and supplements:

PP Changes and Supplements		
Name	**Reference in this ST**	**Description**
SIG Application / Personalisation Data	Chap. 3.1.3	New asset for the TOE´s personalisation phase
A.SIG_PERS	Chap. 3.2.3	New assumption for the TOE´s personalisation phase
T.SIG_PERS_Aut	Chap. 3.3.3	New threat for the TOE´s personalisation phase
T.SIG_PERS_Data	Chap. 3.3.3	New threat for the TOE´s personalisation phase
OT.DTBS_Integrity_TOE	Chap. 4.1.3	Changed objective due to extension of PP regards trusted channel/path
OT.SIG_PERS	Chap. 4.1.3	New security objective for the TOE´s personalisation phase
OE.HI_VAD	Chap. 4.2.3	Changed objective due to extension of PP regards trusted channel/path
OE.Trusted_Environment	Chap. 4.2.3	New objective due to extension of PP regards trusted channel/path
OE.SIG_PERS	Chap. 4.2.3	New security objective for the TOE´s personalisation phase
FDP_ACC.1/SIG Personalisation SFP	Chap. 5.1.1.3	New SFR for the TOE´s personalisation phase
FDP_ACF.1/Signature-Creation SFP	Chap. 5.1.1.3	New Application Note due to extension of PP regards trusted channel/path
FDP_ACF.1/SIG Personalisation SFP	Chap. 5.1.1.3	New SFR for the TOE´s personalisation phase
FDP_ITC.1/DTBS	Chap. 5.1.1.3	Changed Application Note due to extension of PP regards trusted channel/path
FDP_UIT.1/TOE DTBS	Chap. 5.1.1.3	New Application Note due to extension of PP regards trusted channel/path
FMT_MSA.1/SIG Personalisation	Chap. 5.1.1.3	New SFR for the TOE´s personalisation phase

FTP_ITC.1/DTBS Import	Chap. 5.1.1.3	New Application Note due to extension of PP regards trusted channel/path
FTP_TRP.1/TOE	Chap. 5.1.1.3	New Application Note
FPT_AMT.1	Chap. 5.1.1.3	New Application Note
FPT_FLS.1	Chap. 5.1.1.3	New Refinement
FPT_TST.1	Chap. 5.1.1.3	New Application Note and Refinements
FMT_SMF.1	Chap. 5.1.1.3	New SFR due to /AIS 32/
FTP_ITC.1/SIG Personalisation	Chap. 5.1.1.3	New SFR for the TOE´s personalisation phase
R.Trusted_Environment	Chap. 5.2.2	New requirement due to extension of PP regards trusted channel/path
R.SIG_PERS	Chap. 5.2.2	New requirement for the TOE´s personalisation phase

8 Rationale

The following chapters cover the security objectives rationale, the security requirements rationale and the TOE summary specification rationale.

8.1 Security Objectives Rationale

According to the requirements of Common Criteria, /CC 2.3 Part1/ and /CC 2.3 Part3/, the security objectives rationale demonstrates that the stated security objectives are traceable to all of the aspects identified in the TOE security environment and are suitable to cover them. In detail, the security objectives rationale demonstrates that the stated security objectives for the TOE and its environment are suitable to counter the identified threats to security and to cover all of the identified Organisational Security Policies and assumptions. Vice versa, the security objective rationale shows that each security objective of the TOE and its environment at least counters one threat or is correlated to one Organisational Security Policy or assumption.

8.1.1 Threats - Security Objectives

8.1.1.1 General Threats on the TOE

The general threats on the TOE as defined in chap. 3.3.1 can be mapped to the general security objectives for the TOE and its environment which are specified in chap. 4.1.1 and 4.2.1.

The rationale for this mapping can be found in /ST-MIC30/, chap. 8.1.1.1 and 8.1.1.2.

8.1.1.2 Specific Threats on the TOE´s HPC Application

The specific threats on the TOE´s HPC Application as defined in chap. 3.3.2 can be mapped to the specific security objectives for the TOE´s HPC Application which are specified in chap. 4.1.2.

The rationale for this mapping is given in /PP-HPC/, chap. 7.1.

8.1.1.3 Specific Threats on the TOE´s SIG Application

The specific threats on the TOE´s SIG Application as defined in chap. 3.3.3 can be mapped to the specific security objectives for the TOE´s SIG Application and its environment which are specified in chap. 4.1.3 and 4.2.3.

The rationale for this mapping can be found in /PP SSCD Type3/, chap. 6.2.1 and 6.2.2.2 whereat the following supplements have to be taken into account:

T.DTBS_Forgery (Forgery of the DTBS-representation) addresses the threat arising from modifications of the DTBS-representation sent to the TOE for signing which than does not correspond to the DTBS-representation corresponding to the DTBS the signatory intends to sign. In the case a trusted channel by cryptographic means is established the TOE counters this threat by the means of OT.DTBS_Integrity_TOE by verifying the integrity of the DTBS-representation. The TOE IT environment addresses T.DTBS_Forgery by the means of OE.SCA_Data_Intend and OE.Trusted_Environment.

T.SigF_Misuse (Misuse of the signature-creation function of the TOE) addresses the threat of misuse of the TOE signature-creation function to create SDO by others than the signatory to create SDO for data the signatory has not decided to sign, as required by the Directive /PP SSCD Type3/, Annex III, paragraph 1, literal (c). This threat is addressed by the OT.Sigy_SigF (Signature generation function for the legitimate signatory only), OE.SCA_Data_Intend (Data intended to be signed), OT.DTBS_Integrity_TOE (Verification of the DTBS-representation integrity), OE.Trusted_Environment (Trusted Environment for SCA and TOE), and OE.HI_VAD (Protection of the VAD) as follows: OT.Sigy_SigF ensures that the TOE provides the signature-generation function for the legitimate signatory only. OE.SCA_Data_Intend ensures that the SCA sends the DTBS-representation only for data the signatory intends to sign. The combination of OT.DTBS_Integrity_TOE, OE.Trusted_Environment and OE.SCA_Data_Intend counters the misuse of the signature generation function by means of manipulation of the channel between the SCA and the TOE. If the SCA provides the human interface for the user authentication, OE.HI_VAD provides confidentiality and integrity of the VAD as needed by the authentication method employed.

T.Sig_Repud (Repudiation of electronic signatures) deals with the repudiation of signed data by the signatory, although the electronic signature is successfully verified with the SVD contained in his un-revoked certificate. This threat is in general addressed by OE.CGA_Qcert (Generation of qualified certificates), OT.SVD_Auth_TOE (TOE ensures authenticity of the SVD), OE.SVD_Auth_CGA (CGA proves the authenticity of the SVD), OT.SCD_SVD_Corresp (Correspondence between SVD and SCD), OT.SCD_Unique (Uniqueness of the signature-creation data), OT.SCD_Secrecy (Secrecy of the signature-creation data), OT.EMSEC_Design (Provide physical emanations security), OT.Tamper_ID (Tamper detection), OT.Tamper_Resistance (Tamper resistance), OT.Lifecycle_Security (Lifecycle security), OT.Sigy_SigF (Signature generation function for the legitimate signatory only), OT.Sig_Secure (Cryptographic security of the electronic signature), OE.SCA_Data_Intend (SCA sends representation of data intended to be signed), OE.Trusted_Environment (Trusted Environment for SCA and TOE) and OT.DTBS_Integrity_TOE (Verification of the DTBS-representation integrity).

OE.CGA_QCert ensures qualified certificates which allow to identify the signatory and thus to extract the SVD of the signatory. OE.CGA_QCert, OT.SVD_Auth_TOE and OE.SVD_Auth_CGA ensure the integrity of the SVD. OE.CGA_QCert and OT.SCD_SVD_Corresp ensure that the SVD in the certificate correspond to the SCD that is implemented by the SSCD of the signatory. OT.SCD_Unique provides that the signatory's SCD can practically occur just once. OT.Sig_Secure, OT.SCD_Transfer, OT.SCD_Secrecy, OT.Tamper_ID, OT.Tamper_Resistance, OT.EMSEC_Design, and OT.Lifecycle_Security ensure the confidentiality of the SCD implemented in the signatory's SSCD. OT.Sigy_SigF provides that only the signatory may use the TOE for signature generation. OT.Sig_Secure ensures by means of robust cryptographic techniques that valid electronic signatures may only be generated by employing the SCD corresponding to the SVD that is used for signature verification and only for the signed data. OE.SCA_Data_Intend, OE.Trusted_Environment and OT.DTBS_Integrity_TOE ensure that the TOE generates electronic signatures only for DTBS-representations which the signatory has decided to sign as DTBS.

T.SVD_Forgery (Forgery of the signature-verification data) deals with the forgery of the SVD exported by the TOE to the CGA for the generation of the certificate. T.SVD_Forgery is addressed by OT.SVD_Auth_TOE which ensures that the TOE sends the SVD in a verifiable form to the CGA, as well as by OE.SVD_Auth_CGA which provides verification of SVD authenticity by the CGA.

T.SIG_PERS_Aut (Authentication for personalisation process of SIG Application) covers the circumvention of the authentication of the external world prior to loading personalisation data into the TOE. T.SIG_PERS_Aut is addressed by OT.SIG_PERS which ensures that the personalisation process can be started only after a preceding successful authentication of the external world.

T.SIG_PERS_Data (Modification or disclosure of personalisation data of SIG Application) deals with the modification and disclosure of personalisation data imported during the personalisation process. T.SIG_PERS_Data is addressed by OT.SIG_PERS which ensures for the integrity, authenticity and confidentiality of the data import of the personalisation data.

8.1.2 Assumptions - Security Objectives

8.1.2.1 General Assumptions for the TOE

The general assumptions for the TOE as defined in chap. 3.2.1 can be mapped to the general security objectives for the TOE and its environment which are specified in chap. 4.1.1 and 4.2.1. The rationale for this mapping can be found in /ST-MIC30/, chap. 8.1.2.

8.1.2.2 Specific Assumptions for the TOE´s HPC Application

The specific assumptions for the TOE´s HPC Application as defined in chap. 3.2.2 can be mapped to the specific security objectives for the TOE´s HPC Application and its environment which are specified in chap. 4.2.2.

The rationale for this mapping can be found in /PP-HPC/, chap. 7.1.

8.1.2.3 Specific Assumptions for the TOE´s SIG Application

The specific assumptions for the TOE´s SIG Application as defined in chap. 3.2.3 can be mapped to the specific security objectives for the TOE´s SIG Application and its environment which are specified in chap. 4.2.3.

The rationale for this mapping can be found in /PP SSCD Type3/, chap. 6.2.1 and 6.2.2.3 whereat the following supplements have to be taken into account:

A.SIG_PERS (Security of the personalisation process of the SIG Application) covers the security of the TOE´s personalisation process and is directly adressed by OE.SIG_PERS.

8.1.3 Organisational Security Policies - Security Objectives

8.1.3.1 General Organisational Security Policies for the TOE

The general organisational security policies for the TOE as defined in chap. 3.4.1 can be mapped to the general security objectives for the TOE and its environment which are specified in chap. 4.1.1 and 4.2.1. The rationale for this mapping can be found in /ST-MIC30/, chap. 8.1.3.

8.1.3.2 Specific Organisational Security Policies for the TOE´s HPC Application

The specific organisational security policies for the TOE´s HPC Application as defined in chap. 3.4.2 can be mapped to the specific security objectives for the TOE´s HPC Application and its environment which are specified in chap. 4.1.2 and 4.2.2.

The rationale for this mapping can be found in /PP-HPC/, chap. 7.1. The additional organisational security policy OSP.Limit_Usage is directly addressed by the additional security objective OT.Limited_Key_Usage.

8.1.3.3 Specific Organisational Security Policies for the TOE´s SIG Application

The specific organisational security policies for the TOE´s SIG Application as defined in chap. 3.4.3 can be mapped to the specific security objectives for the TOE´s SIG Application and its environment which are specified in chap. 4.1.3 and 4.2.3.

The rationale for this mapping can be found in /PP SSCD Type3/, chap. 6.2.1 and 6.2.2.1.

8.2 Security Requirements Rationale

According to the requirements of Common Criteria, /CC 2.3 Part1/ and /CC 2.3 Part3/, the security requirements rationale demonstrates that the set of security requirements of the TOE is suitable to meet and is traceable to the security objectives for the TOE and its environment. In detail, the following will be demonstrated:

- the combination of the individual functional and assurance requirements components for the TOE and its IT environment together meet the stated security objectives
- the set of security requirements together form a mutually supportive and internally consistent whole
- the choice of security requirements is justified, whereby any of the following conditions is specifically justified:
  - choice of additional requirements not contained in Parts 2 or 3
  - choice of additional assurance requirements not included in EAL 4
  - non-satisfaction of dependencies
- the selected strength of function level for the ST is consistent with the security objectives for the TOE

8.2.1 Security Functional Requirements Rationale

The following section demonstrates that the set and combination of the defined security functional requirements (SFRs) and security assurance requirements (SARs) for the TOE is suitable to satisfy the identified security objectives for the TOE and its environment. Furthermore, this section shows that each of these SARs and SFRs contributes to at least one of the security objectives for the TOE and its environment.

8.2.1.1 General Security Objectives for the TOE – Security Functional Requirements

The general security objectives for the TOE as defined in chap. 4.1.1 are related to the SARs and general SFRs for the TOE specified in chap. 5.1.3 and 5.1.1.1. The mapping of the general security objectives for the TOE to the relevant SARs and SFRs incl. rationale is performed in /ST-MIC30/, chap. 8.2.1.1 and 8.2.1.2.

8.2.1.2 Specific Security Objectives for the TOE´s HPC Application – Security Functional Requirements

The specific security objectives for the TOE´s HPC Application as defined in chap. 4.1.2 are related to the SARs and specific SFRs for the TOE´s HPC Application specified in chap. 5.1.3 and 5.1.1.2. The mapping of the specific security objectives for the TOE´s HPC Application to the relevant SARs and SFRs incl. rationale is performed in /PP-HPC/, chap. 7.2.1

and 7.2.2 whereat some supplements have to be taken into account (refer to the following explanations).

TOE Security Requirements Sufficiency:

The rationale in /PP-HPC/, chap. 7.2.1 and 7.2.2 is still valid under consideration of the following supplements:

The security objective **OT.Limited_Key_Usage** “Limitation of the C2C-Authentication Key” is implemented by the following SFRs:

(i) the SFR FMT_SMR.1 defines the card management system as known role of the TOE,

(ii) the SFR FMT_SMF.1 defines unblocking of the PrK.HPC.AUT as security management function,

(iii) the SFR FMT_MTD.1 limits the management of the key usage counter related to the key PrK.HPC.AUT to the card management system,

(iv) the SFR FIA_AFL.1/C2C protects and limits the usage of the key PrK.HPC.AUT.

TOE Environment Security Requirements Sufficiency:

Not applicable.

8.2.1.3 Specific Security Objectives for the TOE´s SIG Application – Security Functional Requirements

The specific security objectives for the TOE´s SIG Application as defined in chap. 4.1.3 are related to the SARs and specific SFRs for the TOE´s SIG Application which are specified in chap. 5.1.3 and 5.1.1.3. The mapping of the specific security objectives for the TOE´s SIG Application to the relevant SARs and SFRs incl. rationale is performed in /PP SSCD Type3/, chap. 6.3.1 and 6.3.2 whereat some supplements have to be taken into account (refer to the following explanations).

TOE Security Requirements Sufficiency:

The rationale in /PP SSCD Type3/, chap. 6.3.2.1 is still valid under consideration of the following supplements:

OT.Sigy_SigF (Signature generation function for the legitimate signatory only) is provided by FIA_UAU.1 and FIA_UID.1 that ensure that no signature generation function can be invoked before the signatory is identified and authenticated.

The security functions specified by FDP_ACC.1/Personalisation SFP, FDP_ACC.1/Signature-Creation SFP, FDP_ACF.1/Personalisation SFP, FDP_ACF.1/Signature-Creation SFP, FMT_MTD.1 and FMT_SMR.1 ensure that the signature process is restricted to the signatory.

The security functions specified by FIA_ATD.1, FMT_MOF.1, FMT_MSA.2, and FMT_MSA.3 and FMT_SMF.1 ensure that the access to the signature generation functions remain under the sole control of the signatory, as well as FMT_MSA.1/Signatory provides that the control of corresponding security attributes is under signatory's control.

The security functions specified by FDP_SDI.2 and FPT_TRP.1/TOE ensure the integrity of stored data both during communication and while stored.

The security functions specified by FDP_RIP.1 and FIA_AFL.1 provide protection against a number of attacks, such as cryptographic extraction of residual information, or brute force attacks against authentication.

The assurance measures specified by AVA_MSU.3 by requesting analysis of misuse of the TOE implementation, AVA_SOF.1 by requesting high strength level for security functions, and AVA_VLA.4 by requesting that the TOE resists attacks with a high attack potential assure that the security functions are efficient.

OT.SIG_PERS (Security of the personalisation process for the SIG Application) guarantees for a secure personalisation process and is provided by the security functions specified by FDP_ACC.1/SIG Personalisation SFP, FDP_ACF.1/SIG Personalisation SFP, FIA_UID.1, FIA_UAU.1 and FTP_ITC.1/SIG Personalisation which ensure that only authorised users can load the personalisation data and that the personalisation process is secured for integrity, authenticity and confidentiality. The security function specified by FMT_MSA.1/SIG Personalisation provides the secure handling of the security attributes related to the personalisation process.

TOE Environment Security Requirements Sufficiency:

The rationale in /PP SSCD Type3/, chap. 6.3.2.2 is still valid under consideration of the following supplements:

OE.HI_VAD (Protection of the VAD) covers confidentiality and integrity of the VAD which is provided by the trusted path FTP_TRP.1/SCA or the Trusted Environment R.Trusted_Environment.

OE.Trusted_Environment (Trusted Environment for SCA and TOE) is provided by R.Trusted_Environment which serves in the case that a trusted channel resp. trusted path between the TOE and the SCA by cryptographic means is not established that the environment for the TOE usage is secured with the target to keep confidentiality and integrity of the VAD and integrity of the DTBS within the data transfer to the TOE.

OE.SIG_PERS (Security of the personalisation process for the SIG Application) is directly provided by R.SIG_PERS which serves for a secure personalisation process.

8.2.2 Security Functional Requirements Dependencies

The following section demonstrates that all dependencies between the identified security functional requirements included in this ST are satisfied.

8.2.2.1 General SFRs of the TOE

The dependencies under the general SFRs of the TOE as defined in chap. 5.1.1.1 are considered in /ST-MIC30/, chap. 8.2.2.1 and 8.2.2.2.

8.2.2.2 Specific SFRs of the TOE´s HPC Application

The dependencies under the specific SFRs of the TOE´s HPC Application as defined in chap. 5.1.1.2 are considered in /PP-HPC/, chap. 7.2.3 and 10. In particular, a justification for non-satisfied dependencies is given.

8.2.2.3 Specific SFRs of the TOE´s SIG Application

The dependencies under the specific SFRs of the TOE´s SIG Application as defined in chap. 5.1.1.3 are considered in /PP SSCD Type3/, chap. 6.4.1. In particular, a justification for non-satisfied dependencies is given in /PP SSCD Type3/, chap. 6.4.2.

The dependencies under the specific SFRs concerning the IT-environment of the TOE related to the TOE´s SIG Application as defined in chap. 5.2.1 are considered in /PP SSCD Type3/, chap. 6.4.1. In particular, a justification for non-satisfied dependencies is given in /PP SSCD Type3/, chap. 6.4.2.

8.2.3 Strength of Function Level Rationale

Due to the requirements for smartcard products intended to be used for high security applications within the German Health Care System the level for the strength of the TOE´s security functional requirements is claimed as SOF-high. The TOE is considered as a product with critical security mechanisms which only have to be defeated by attackers possessing a high level of expertise, opportunity and resources, and whereby successful attack is judged beyond normal practicality. Refer as well to the explanations in /PP-HPC/, chap. 7.2.4 and /PP SSCD Type3/, chap. 6.7.

8.2.4 Security Assurance Requirements Rationale

The assurance requirements of this ST defined in chap. 5.1.3 are summarized in the following table:

Assurance Requirements	Name	Type
EAL4	Methodically Designed, Tested and Reviewed	Assurance Level / Class
ADV_IMP.2	Implementation of the TSF	Higher hierarchical component
ATE_DPT.2	Testing: Low-Level Design	Higher hierarchical component
AVA_MSU.3	Analysis and Testing for Insecure	Higher hierarchical component

	States	
AVA_VLA.4	Highly Resistant	Higher hierarchical component

8.2.4.1 Evaluation Assurance Level Rationale

Due to the requirements for smartcard products intended to be used for high security applications within the German Health Care System the assurance level for the TOE is chosen as EAL4 augmented by ADV_IMP.2, ATE_DPT.2, AVA_MSU.3 and AVA_VLA.4. Hereby, all assurance components will be used as defined in /CC 2.3 Part3/ and /CEM 2.3 Part2/.

The evaluation assurance level of EAL4 augmented is selected for the TOE since this level provides an adequate and meaningful level of assurance for the TOE, with regard to the security of the development process of the TOE as well as with regard to the TOE´s security and resistance against attacks with high attack potential in its operational use. The chosen assurance level permits the developer to gain maximum assurance from positive security engineering based on good commercial practices and represents a sufficiently high practical level of assurance expected for the security product. Furthermore, to guarantee for a sufficiently secure product, the evaluators should have access especially to the low level design and source code, whereby the lowest assurance level for such access is given with the assurance class EAL4.

EAL4 allows a developer to attain a reasonably high assurance level without the need for highly specialized processes and practices. It is considered to be the highest level that could be applied to an existing product line without undue expense and complexity. As such, EAL4 is appropriate for commercial products that can be applied to moderate to high security functions.

The assurance level EAL4 augmented requires knowledge of the Common Criteria evaluation scheme and process, but does not make use of specialist techniques on the part of the developer.

A more detailed rationale for the chosen augmentations of the evaluation assurance class EAL4 is provided in the following chap. 8.2.4.2.

8.2.4.2 Assurance Augmentations Rationale

The following section gives reason for the choice of the assurance components augmenting the evaluation assurance class EAL4. Refer as well to /PP-HPC/, chap. 7.2.4 and /PP SSCD Type3/, chap. 6.8.

Apriori, the assurance components ADV_IMP.2, ATE_DPT.2, AVA_MSU.3 and AVA_VLA.4 are chosen with respect to the common understanding of security requirements for high security smartcards intended to be used in the framework of the German Health Care System.

In detail, the following deliberations are of interest:

ADV_IMP.2 Implementation of the TSF

The implementation representation is used to express the notion of the least abstract representation of the TSF, specifically the one that is used to create the TSF itself without further design refinement.

The assurance component ADV_IMP.2 is a higher hierarchical component to EAL4, which only requires ADV_IMP.1 „Subset of the implementation of the TSF".

The augmentation by ADV_IMP.2 is chosen for the following reason: It is important for the TOE and its assurance that the evaluator evaluates the implementation representation of the *entire* TSF to determine that the SFRs as defined in the ST are addressed by the representation of the TSF and that the implementation representation is an accurate and complete instantiation of the TOE´s SFRs. This provides a direct correspondence between the TOE´s SFRs and the implementation representation, in addition to the pairwise correspondences required by the ADV_RCR family. The augmentation by ADV_IMP.2 is chosen according to the requirements in the Protection Profiles /PP-HPC/ and /PP SSCD Type3/.

ATE_DPT.2 Testing: Low-Level Design

Testing of the TSFs and their internal structure is done with the objective to counter the risk of missing an error or malicious code in the development of the TOE. Testing that exercises specific internal interfaces can provide assurance not only that the TSF exhibits the desired external security behaviour, but also that this behaviour stems from correctly operating internal mechanisms.

The assurance component ATE_DPT.2 is a higher hierarchical component to EAL4, which only requires ATE_DPT.1 „Testing: high-level design".

It is important for the TOE and its assurance that testing of the TSFs is not only done on basis of the high-level description of the internal workings of the TSF (level of the subsystems) in order to demonstrate the absence of any flaws and to provide assurance that the TSF subsystems have been correctly realised. Moreover, the testing of the TSFs shall cover tests on the modules of the TSFs providing a low-level description of the internal workings of the TSF with the goal to demonstrate the absence of any flaws and to provide assurance that the TSF modules have been correctly realised. The depth analysis shall demonstrate that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates in accordance with its high-level design and *low-level design*.

AVA_MSU.3 Analysis and Testing for Insecure States

Misuse investigates whether the TOE can be configured or used in a manner that is insecure but that an administrator or user of the TOE would reasonably believe to be secure.

The assurance component AVA_MSU.3 is a higher hierarchical component to EAL4, which only requires AVA_MSU.2 „Validation of analysis".

The augmentation by AVA_MSU.3 is chosen according to the requirements in the Protection Profiles /PP-HPC/ and /PP SSCD Type3/. Due to the nature of the TOE´s intended application, the TOE may be issued to users and may not be directly under the control of trained and dedicated administrators. As a result, it is imperative that misleading, unreasonable and conflicting guidance is absent from the guidance documentation, and that secure procedures for all modes of operation have been addressed. Insecure states should be easy to detect. In

AVA_MSU.3, an analysis of the guidance documentation by the developer is required to provide additional assurance that the objective has been met, and this analysis is validated and confirmed through testing by the evaluator.

AVA_VLA.4 Highly Resistant

According to the definition of the TOE, it must be shown to be highly resistant to penetration attacks. This is due to the fact that the TOE can be placed in a hostile environment.

This assurance requirement is achieved by the assurance component AVA_VLA.4. Independent vulnerability analysis is based on highly detailed technical information. The attacker is assumed to be thoroughly familiar with the specific implementation of the TOE and is presumed to have a high level of technical sophistication.

The assurance component AVA_VLA.4 is a higher hierarchical component to EAL4, which only requires AVA_VLA.2 „Independent vulnerability analysis“.

The augmentation by AVA_VLA.4 is chosen according to the requirements in the Protection Profiles /PP-HPC/ and /PP SSCD Type3/. For AVA_VLA.4, a systematical vulnerability analysis is performed by the developer to ascertain the presence of security vulnerabilities, and to confirm that they cannot be exploited in the intended environment for the TOE. Hereby, the analysis shall provide a justification that the analysis completely addresses the TOE deliverables. The evaluator performs independent penetration testing, supported by the evaluator’s independent vulnerability analysis, to determine that the TOE is resistant to penetration attacks performed by attackers possessing a high attack potential.

8.2.5 Security Assurance Requirements Dependencies

The security assurance requirements specified by this ST are drawn from the assurance class EAL4 with its augmentation by the higher hierarchical components ADV_IMP.2, ATE_DPT.2, AVA_MSU.3 and AVA_VLA.4.

EAL4 is asserted to be a known set of assurance components for which all dependencies are satisfied. For the components of the augmentation the following deliberation shows that all further dependencies resulting from the augmentation are satisfied:

ADV_IMP.2 has dependencies with ADV_LLD.1 „Descriptive Low-Level design”, ADV_RCR.1 „Informal correspondence demonstration”, ALC_TAT.1 „Well defined development tools”. These components are included in EAL4, and so these dependencies are satisfied.

ATE_DPT.2 has dependencies with ADV_HLD.2 „Security enforcing high-level design“, ADV_LLD.1 „Descriptive low-level design“ and ATE_FUN.1 „Functional testing“. All these dependencies are satisfied by EAL4.

AVA_MSU.3 has dependencies with ADO_IGS.1 “Installation, generation, and start-up procedures”, ADV_FSP.1 “Informal functional specification”, AGD_ADM.1 “Administrator guidance” and AGD_USR.1 “User guidance”. All these dependencies are satisfied by EAL4.

AVA_VLA.4 has dependencies with ADV_FSP.1 „Informal functional specification", ADV_HLD.2 „Security enforcing high-level design", ADV_LLD.1 „Descriptive low level design", ADV_IMP.1 „Subset of the implementation of the TSF", AGD_ADM.1" Administrator Guidance" and AGD_USR.1 „User Guidance". All these dependencies are satisfied by EAL4.

8.2.6 Security Requirements – Mutual Support and Internal Consistency

The following part of the security requirements rationale shows that the set of security requirements for the TOE consisting of the security assurance requirements (SARs) and the security functional requirements (SFRs) together forms a mutually supportive and internally consistent whole.

The analysis of the TOE´s security requirements with regard to their mutual support and internal consistency demonstrates:

- The assurance class EAL4 is an established set of mutually supportive and internally consistent assurance requirements.
- The dependency analysis for the additional assurance components in chap. 8.2.5 shows that the assurance requirements are mutually supportive and internally consistent as all (additional) dependencies are satisfied and no inconsistency appears.
- The dependency analysis for the security functional requirements of the TOE in general (IC and MICARDO V3.0 Operating System platform) in chap. 8.2.2.1 as well as of the TOE´s HPC and SIG Application in chap. 8.2.2.2 and 8.2.2.3 shows that the basis for mutual support and internal consistency between all defined functional requirements is satisfied. All dependencies between the chosen functional components are analysed, and non-dissolved dependencies are appropriately explained.

  The mutual support and internal consistency of the functional requirements is shown for the TOE in general (IC and MICARDO V3.0 Operating System platform) in chap. 8.2.1.1 as well as for the TOE´s HPC and SIG Application in chap. 8.2.1.2 and 8.2.1.3 within the mapping of the security objectives to the SFRs.

  Concerning the SFRs of the TOE´s HPC Application and SIG Application, the SFRs have been chosen under consideration of the Protection Profiles /PP-HPC/ resp. /PP SSCD Type3/. Obviously, overlapping SFRs defined for the TOE in general (see chap. 5.1.1.1) and for the TOE´s HPC Application and SIG Application (see chap. 5.1.1.2, 5.1.1.3) do not lead to any inconsistency or any weakness or contradict one another.
- All operations (assignment, selection, iteration and refinement) conducted on the CC functional components lead to a consistent and meaningful whole.

  First, all operations on the chosen SFRs are done with the target to reflect correctly and completely the security functionality provided by the TOE whereat the operations in this ST take the operations already done within the Protection Profiles /PP-HPC/ resp. /PP SSCD Type3/ into account. Furthermore, all assignment, selection, iteration and refinement operations are conducted in such a way that they do not contradict each other and build an internally consistent security system. In particular, the iterations of the functional components for cryptographic support, FCS_CKM and FCS_COP, are necessary to differentiate between the different cryptographic algorithms and mechanisms of the TOE. The iteration of the functional component FIA_AFL is necessary to differentiate between the different authentication mechanisms provided by the TOE.

- Inconsistency between functional and assurance requirements can only arise if there are functional-assurance dependencies which are not met, a possibility which has been shown not to arise in chap. 8.2.2. Furthermore, as discussed in chap. 8.2.4, the chosen assurance components are adequate for the functionality of the TOE what underlines that the assurance requirements and security functional requirements support each other and that there are no inconsistencies between these two groups of security requirements.

Refer as well to the explanations in /PP-HPC/, chap. 7.2.5 and /PP SSCD Type3/, chap. 6.8.

8.3 TOE Summary Specification Rationale

According to the requirements of Common Criteria, /CC 2.3 Part1/ and /CC 2.3 Part3/, the TOE summary specification rationale demonstrates that the TOE security functions (TSFs) and assurance measures are suitable to meet the TOE security requirements. In detail, the following will be demonstrated:

- the combination of the specified TOE´s IT security functions work together so as to satisfy the TOE security functional requirements
- the strength of the TOE function claims made are valid, or assertions that such claims are unnecessary are valid
- the claim that the stated assurance measures are compliant with the assurance requirements is justified

8.3.1 Security Functions Rationale

The following section demonstrates that the set and combination of the defined TOE security functions (TSFs) is suitable to satisfy the identified TOE security functional requirements (SFRs). Furthermore, this section shows that each of the TSFs is related to at least one security functional requirement.

8.3.1.1 General Security Functional Requirements for the TOE – TOE Security Functions

The mapping of the general SFRs for the TOE as defined in chap. 5.1.1.1 to the TSFs incl. the related rationale is part of /ST-MIC30/, chap. 8.3.1.1 and 8.3.1.2. Note that the TSF F.ACS as defined in /ST-MIC30/, chap. 6.1.2 is covered by the new TSF F.ACS_SFP of chap. 6.1.2.

8.3.1.2 Specific Security Functional Requirements for the TOE´s HPC Application – TOE Security Functions

The mapping of the specific SFRs for the TOE´s HPC Application as defined in chap. 5.1.1.2 to the TSFs as specified in chap. 6.1.2 is done in the following.

The table below gives an overview of which TSFs contribute to the realisation of the specific SFRs related to the TOE´s HPC Application.

Security Functional Requirements / TOE Security Functions	F.ACS_SFP	F.IA_AKEY	F.IA_SKEY	F.IA_PWD	F.DATA_INT	F.EX_CONF	F.EX_INT	F.RIP	F.FAIL_PROT	F.SIDE_CHAN	F.SELFTEST	F.CRYPTO	F.RSA_KEYGEN	F.GEN_DIGSIG	F.VER_DIGSIG	F.RSA_ENC	F.RSA_DEC	TSFs of IC
FCS_CKM.1 / ASYM		x								x		x						(x)
FCS_CKM.1 / SYM			x							x		x						(x)
FCS_CKM.4	(x)							x										
FCS_COP.1 / CSA	(x)									x		(x)		x				(x)
FCS_COP.1 / CCA_SIGN	(x)									x		(x)		x				(x)
FCS_COP.1 / RSA_DEC	(x)									x		(x)					x	(x)
FCS_COP.1 / CCA_VERIF	(x)									x		(x)			x			(x)
FCS_COP.1 / TDES	(x)									x		x						(x)
FCS_COP.1 / MAC	(x)									x		x						(x)
FCS_COP.1 / SHA										x		x						(x)
FCS_RND.1										x		x						(x)
FDP_ACC.2	x																	
FDP_ACF.1	x																	
FDP_RIP.1								x										
FDP_SDI.2 / Int-PersData					x													
FDP_SDI.2 / Int-TempData					x													
FDP_UCT.1						x						x						(x)
FDP_UIT.1							x					x						(x)
FIA_AFL.1 / HPC-PIN				x														
FIA_AFL.1 / C2C		x																
FIA_ATD.1	x																	
FIA_UAU.1	x																	
FIA_UAU.4		x	x									x						
FIA_UAU.6		x	x															
FIA_UID.1	x																	
FMT_LIM.1																		x
FMT_LIM.2																		x
FMT_MTD.1 / INI	x																	
FMT_MTD.1 / RAD_WR	x																	
FMT_MTD.1 / RAD_MOD	x																	
FMT_MTD.1 / PIN	x																	
FMT_MTD.1 / RAD_CH	x																	
FMT_MTD.1 / C2C	x																	
FMT_SMF.1	x																	
FMT_SMR.1	x																	
FPT_EMSEC.1										x								(x)
FPT_FLS.1									x									
FPT_PHP.3										x								x
FPT_RVM.1											x							
FPT_SEP.1	x																	

FPT_TST.1											x							
FTP_ITC.1		x	x			x	x											

Note:

X directly contributing TSF

(X) supporting TSF

The detailed description and analysis of the TOE Security Functions in chap. 6.1 demonstrate how the defined functions work together and support each other. Furthermore, this description shows that no inconsistencies exist. The deliberations above support this result.

In the following, for each SFR related to the TOE´s HPC Application it will be explained why and how the TSFs listed in the preceding tables meet the respective SFR.

FCS_CKM.1 / ASYM, FCS_CKM.1 / SYM

The generation of session keys used for securing the following data exchange is part of the TSFs F.IA_AKEY, F.IA_SKEY and F.CRYPTO and is carried out according to the requirements of the SFR FCS_CKM.1 / ASYM and SFR FCS_CKM.1 / SYM. The security of the key generation process is given by the TSF F.SIDE_CHAN and further TSFs of the underlying IC.

FCS_CKM.4

The TSF F.RIP fulfills the SFR FCS_CKM.4 as it implements the memory preparation upon the deallocation of resources whereby it ensures that any previous information content is no longer available. This concerns in particular the erasing of all volatile and non-volatile memories used for processing cryptographic keys or key related material.

FCS_COP.1 / CSA, FCS_COP.1 / CCA_SIGN

The TSF F.GEN_DIGSIG with support of the TSF F.CRYPTO supplies the functionality of creating electronic signatures and is carried out according to the requirements defined in the SFRs FCS_COP.1 / CSA and FCS_COP.1 / CCA_SIGN. The security of the signature-creation process is given by the TSF F.SIDE_CHAN and further TSFs of the underlying IC. The access to the relevant keys is regulated by the TSF F.ACS_SFP which implements the SFP HPC Access Control defined in chap. 5.1.1.2 with its dedicated access conditions for keys.

FCS_COP.1 / RSA_DEC

The TSF F.RSA_DEC with support of the TSF F.CRYPTO supplies the functionality of RSA decryption and is carried out according to the requirements defined in the SFR FCS_COP.1 / RSA_DEC. The security of the decryption process is given by the TSF F.SIDE_CHAN and further TSFs of the underlying IC. The access to the relevant keys is regulated by the TSF F.ACS_SFP which implements the SFP HPC Access Control defined in chap. 5.1.1.2 with its dedicated access conditions for keys.

FCS_COP.1 / CCA_VERIF

The TSF F.VER_DIGSIG with support of the TSF F.CRYPTO supplies the functionality of verifying electronic signatures and is carried out according to the requirements defined in the SFR FCS_COP.1 / CCA_VERIF. The access to the relevant keys is regulated by the TSF F.ACS_SFP which implements the SFP HPC Access Control defined in chap. 5.1.1.2 with its dedicated access conditions for keys.

FCS_COP.1 / TDES, FCS_COP.1 / MAC

The TSF F.CRYPTO covers the crypto functionality as required by the SFRs FCS_COP.1 / TDES and FCS_COP.1 / MAC. The security of the crypto functions is given by the TSF F.SIDE_CHAN and further TSFs of the underlying IC. The access to the relevant keys is regulated by the TSF F.ACS_SFP which implements the SFP HPC Access Control defined in chap. 5.1.1.2 with its dedicated access conditions for keys.

FCS_COP.1 / SHA, FCS_RND.1

The TSF F.CRYPTO covers the crypto functionality as required by the SFRs FCS_COP.1 / SHA and FCS_RND.1. The security of the crypto functions is given by the TSF F.SIDE_CHAN and further TSFs of the underlying IC.

FDP_ACC.2, FDP_ACF.1

The TSF F.ACS_SFP contributes directly to the SFRs FDP_ACC.2 and FDP_ACF.1 as it implements the SFP HPC Access Control defined in chap. 5.1.1.2.

FDP_RIP.1

The TSF F.RIP contributes directly to the SFR FDP_RIP.1 as it implements the memory preparation upon the deallocation of the respective resource whereby it ensures that any previous information content is no longer available. This concerns all volatile and non-volatile memories used for processing security relevant material.

FDP_SDI.2 / Int-PersData, FDP_SDI.2 / Int-TempData

The TSF F.DATA_INT contributes directly to the SFRs FDP_SDI.2 / Int-PersData and FDP_SDI.2 / Int-TempData as it realizes the monitoring of stored data for integrity errors. This concerns especially user data values as well as user data objects.

FDP_UCT.1

The TSF F.EX_CONF serves for a confidential communication channel by cryptographic means as required in the SFR FDP_UCT.1. The TSF F.EX_CONF is supported by the TSF F.CRYPTO and further TSFs of the underlying IC.

FDP_UIT.1

The TSF F.EX_INT serves for an integrity secured communication channel by cryptographic means as required in the SFR FDP_UIT.1. The TSF F.EX_INT is supported by the TSF F.CRYPTO and further TSFs of the underlying IC.

FIA_AFL.1 / HPC-PIN

The TSF F.IA_PWD realises the password based authentication mechanism of the TOE and is particularly responsible for the handling of authentication failures as required in the SFR FIA_AFL.1 / HPC-PIN.

FIA_AFL.1 / C2C

The TSF F.IA_AKEY realises the key based authentication mechanism of the TOE (on the base of asymmetric cryptography) and is particularly responsible for the handling of authentication failures as required in the SFR FIA_AFL.1 / C2C.

FIA_ATD.1

The maintaining of the security attributes as required by the SFR FIA_ATD.1 is realised by the TSF F.ACS_SFP as it implements the SFP HPC Access Control defined in chap. 5.1.1.2 which concerns especially the required security attributes.

FIA_UAU.1

The TSF F.ACS_SFP fulfills directly the SFR FIA_UAU.1 as it implements the SFP HPC Access Control defined in chap. 5.1.1.2 with its appropriate dedicated access regulations.

FIA_UAU.4

The TSFs F.IA_AKEY and F.IA_SKEY (supported by the TSF F.CRYPTO for random number generation) implement the key based authentication mechanisms of the TOE and handle particularly authentication data as required in the SFR FIA_UAU.4.

FIA_UAU.6

The TSFs F.IA_AKEY and F.IA_SKEY implement the key based authentication mechanisms of the TOE and handle in particular the necessity for re-authentication as required in the SFR FIA_UAU.6.

FIA_UID.1

The TSF F.ACS_SFP fulfills directly the SFR FIA_UID.1 as it implements the SFP HPC Access Control defined in chap. 5.1.1.2 with its appropriate dedicated access regulations.

FMT_LIM.1, FMT_LIM.2

The TSFs of the underlying IC, in particular F.COMP, serve for the realisation of the requirements specified in FMT_LIM.1 and FMT_LIM.2.

FMT_MTD.1 / INI, FMT_MTD.1 / RAD_WR, FMT_MTD.1 / RAD_MOD, FMT_MTD.1 / PIN, FMT_MTD.1 / RAD_CH, FMT_MTD.1 / C2C, FMT_SMF.1, FMT_SMR.1

Access restriction and its handling as required in the SFRs FMT_MTD.1 / INI, FMT_MTD.1 / RAD_WR, FMT_MTD.1 / RAD_MOD, FMT_MTD.1 / PIN, FMT_MTD.1 / RAD_CH, FMT_MTD.1 / C2C, FMT_SMF.1 and FMT_SMR.1 is regulated by the TSF F.ACS_SFP which implements the SFP HPC Access Control defined in chap. 5.1.1.2.

FPT_EMSEC.1

The TSF F.SIDE_CHAN with support of further TSFs of the underlying IC supplies effective hardware and software based mechanisms against side channel attacks satisfying the requirements of the SFR FPT_EMSEC.1.

FPT_FLS.1

The TSF F.FAIL_PROT realises effective hardware and software based features to preserve a secure operation state of the TOE in case of induced hardware or software failures or tampering. It satisfies directly the requirements of the SFR FPT_FLS.1.

FPT_PHP.3

Resistance to physical attacks is given directly by the TSFs of the underlying IC and by the TSF F.SIDE_CHAN which realise effective hardware and software based mechanisms against side channel attacks.

FPT_RVM.1

The TSF F.SELFTEST fulfills the requirements of the SFR FPT_RVM.1.

FPT_SEP.1

The TSF F.ACS_SFP implements different SFPs defined for the TOE and its applications. These SFPs regulate the access to the different TOE memories and stored data. In particular, the defined access regulations match the requirements of the SFR FPT_SEP.1.

FPT_TST.1

The TSF F.SELFTEST fulfills directly the requirements of the SFR FPT_TST.1.

FTP_ITC.1

The TSFs F.IA_AKEY and F.IA_SKEY serve for the installation of a trusted channel as required in the SFR FTP_ITC.1. The secure communication itself is conducted by the TSFs F.EX_INT and F.EX_CONF, if required, according to the requirements in the SFR FTP_ITC.1.

8.3.1.3 Specific Security Functional Requirements for the TOE´s SIG Application – TOE Security Functions

The mapping of the specific SFRs for the TOE´s SIG Application as defined in chap. 5.1.1.3 to the TSFs as specified in chap. 6.1.2 is done in the following.

The table below gives an overview of which TSFs contribute to the realisation of the SFRs related to the TOE´s SIG Application.

Security Functional Requirements / TOE Security Functions	F.ACS_SFP	F.IA_AKEY	F.IA_SKEY	F.IA_PWD	F.DATA_INT	F.EX_CONF	F.EX_INT	F.RIP	F.FAIL_PROT	F.SIDE_CHAN	F.SELFTEST	F.CRYPTO	F.RSA_KEYGEN	F.GEN_DIGSIG	F.VER_DIGSIG	F.RSA_ENC	F.RSA_DEC	TSFs of IC
FCS_CKM.1	x									x		(x)	x					(x)
FCS_CKM.4								x					x					
FCS_COP.1/CORRESP										x		(x)		x				(x)
FCS_COP.1/SIGNING-PKCS1										x		(x)		x				(x)
FCS_COP.1/SIGNING-ISO9796-2										x		(x)		x				(x)
FDP_ACC.1/Initialisation SFP	x																	
FDP_ACC.1/SVD Transfer SFP	x																	
FDP_ACC.1/Personalisation SFP	x																	
FDP_ACC.1/Signature-Creation SFP	x																	
FDP_ACC.1/SIG Personalisation SFP	x																	
FDP_ACF.1/Initialisation SFP	x																	
FDP_ACF.1/SVD Transfer SFP	x																	
FDP_ACF.1/Personalisation SFP	x																	
FDP_ACF.1/Signature-Creation SFP	x																	
FDP_ACF.1/SIG Personalisation SFP	x																	
FDP_ETC.1/SVD Transfer	x																	

FDP_ITC.1/DTBS	**x**																	
FDP_RIP.1								**x**										
FDP_SDI.2/Persistent					**x**													
FDP_SDI.2/DTBS					**x**													
FDP_UIT.1/SVD Transfer							**x**					**(x)**						**(x)**
FDP_UIT.1/TOE DTBS							**x**					**(x)**						**(x)**
FIA_AFL.1				**x**														
FIA_ATD.1	**x**																	
FIA_UAU.1	**x**																	
FIA_UID.1	**x**																	
FMT_MOF.1	**x**			**x**														
FMT_MSA.1/Administrator	**x**																	
FMT_MSA.1/Signatory	**x**			**x**														
FMT_MSA.1/SIG Personalisation	**x**																	
FMT_MSA.2	**x**																	
FMT_MSA.3	**x**																	
FMT_MTD.1	**x**			**x**														
FMT_SMF.1	**x**																	
FMT_SMR.1	**x**																	
FPT_AMT.1											**x**							
FPT_EMSEC.1										**x**								**(x)**
FPT_FLS.1									**x**									
FPT_PHP.1																		**x**
FPT_PHP.3										**x**								**x**
FPT_TST.1											**x**							
FTP_ITC.1/SVD Transfer						**x**	**x**					**(x)**						**(x)**
FTP_ITC.1/DTBS Import				**x**		**x**	**x**					**(x)**						**(x)**
FTP_ITC.1/SIG Personalisation	**x**					**x**	**x**					**(x)**						**(x)**
FTP_TRP.1/TOE				**x**		**x**	**x**					**(x)**						**(x)**

Note:

X	directly contributing TSF
(X)	supporting TSF

The detailed description and analysis of the TOE Security Functions in chap. 6.1 demonstrate how the defined functions work together and support each other. Furthermore, this description shows that no inconsistencies exist. The deliberations above support this result.

In the following, for each SFR related to the TOE´s SIG Application it will be explained why and how the TSFs listed in the preceding tables meet the respective SFR.

The rationale here is presented in form of tables. The full rationale as given in the TOE´s Security Target is not intended to be published and hence not part of the ST-Lite.

8.3.2 Assurance Measures Rationale

The assurance measures of the developer as mentioned in chap. 6.3 are considered to be suitable and sufficient to meet the CC assurance level EAL4 augmented by ADV_IMP.2, ATE_DPT.2, AVA_MSU.3 and AVA_VLA.4 as claimed in chap. 5.1.3. Especially the deliverables listed in chap. 6.3 are seen to be suitable and sufficient to document the fulfillment of the assurance requirements in detail.

As the development and production process of the TOE is very complex and a great number of assurance measures are implemented by the developer, a detailed description of these measures beyond the information given in chap. 2.2 and 2.3 as well as a detailed mapping of the assurance measures to the assurance requirements is not in the scope of this ST.

8.3.3 TOE Security Functions – Mutual Support and Internal Consistency

The detailed description of the TOE Security Functions in chap. 6.1 demonstrates how the defined functions work together and support each other. Furthermore, this description shows that no inconsistencies exist. The deliberations in chap. 8.3.1 support this result.

8.3.4 Strength of Functions

The selected Strength of Functions level for the TOE´s security functions of SOF-high is consistent with the security objectives for the TOE, as the TOE is considered as a security product with critical security mechanisms which shall be resistant against attacks with high attack potential.

8.4 Extensions

For a definition and description of the SFRs FCS_RND.1 „Quality Metric for Random Numbers“, FPT_EMSEC.1 „TOE Emanation“, FMT_LIM.1 „Limited capabilities“ and FMT_LIM.2 „Limited availability“ refer to /ST-MIC30/, chap. 8.4.

8.5 PP Claims Rationale

According to chapter 1.3 and 7, this Security Target claims conformance to the Protection Profile “Health Professional Card (HPC) – Heilberufsausweis (HBA)” /PP-HPC/ registered and certified by Bundesamt für Sicherheit in der Informationstechnik (BSI).

In chapter 7.1 of this document, it is clearly outlined that there are no substantial changes to the PP. The operations done for the SFRs taken from the PP are also clearly indicated.

The evaluation assurance level claimed for this TOE (EAL 4 augmented with ADV_IMP.2, ATE_DPT.2, AVA_MSU.3 and AVA_VLA.4) is shown in chapter 5.1.2 to include respectively exceed the requirements claimed by the PP (EAL 4 augmented with AVA_MSU.3 and AVA_VLA.4).

Reference

I Bibliography

/CC 2.3 Part1/
Title: Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model
Identification: CCIMB-2005-08-001
Version: Version 2.3
Date: August 2005
Author: CC Project Sponsoring Organisations CSE, SCSSI, BSI, NLNCSA, CESG, NIST, NSA

/CC 2.3 Part2/
Title: Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements
Identification: CCIMB-2005-08-002
Version: Version 2.3
Date: August 2005
Author: CC Project Sponsoring Organisations CSE, SCSSI, BSI, NLNCSA, CESG, NIST, NSA

/CC 2.3 Part3/
Title: Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements
Identification: CCIMB-2005-08-003
Version: Version 2.3
Date: August 2005
Author: CC Project Sponsoring Organisations CSE, SCSSI, BSI, NLNCSA, CESG, NIST, NSA

/CEM 0.6 Part1/
Title: Common Methodology for Information Technology Security Evaluation, Part 1: Introduction and General Model
Identification: CEM99/045
Version: Draft 0.6
Date: Jan. 1997
Author: CC Project Sponsoring Organisations CSE, SCSSI, BSI, NLNCSA, CESG, NIST, NSA

/CEM 2.3 Part2/
Title: Common Methodology for Information Technology Security Evaluation, Part 2: Evaluation Methodology
Identification: CCIMB-2005-08-004
Version: Version 2.3
Date: August 2005
Author: CC Project Sponsoring Organisations CSE, SCSSI, BSI, NLNCSA, CESG, NIST, NSA

/AIS32/
Title: Übernahme international abgestimmter CC Interpretationen
Identification: AIS 32
Date: 02.07.2001
Publisher: Bundesamt für Sicherheit in der Informationstechnik

/PP9806/
Title: Protection Profile - Smartcard Integrated Circuit
Identification: Registered at the French Certification Body (DCSSI) under the number PP/9806
Version: Version 2.0
Date: Sept. 1998
Author: Motorola Semiconductors, Philips Semiconductors, Service Central de la Securite des Systemes d´Information, Siemens AG Semiconductors, ST Microelectronics, Texas-Instruments Semiconductors

/PP9911/
Title: Protection Profile - Smartcard Integrated Circuit with Embedded Software
Identification: Registered at the French Certification Body (DCSSI) under the number PP/9911
Version: Version 2.0
Date: June 1999
Author: Atmel Smart Card ICs, Bull-SC&T, De la Rue – Card Systems, Eurosmart, Gemplus, Giesecke & Devrient GmbH, Hitachi Europe Ltd, Infineon Technologies AG, Microelectronica Espana, Motorola SPS, NEC Electronics, Oberthur Smart Card, ODS, ORGA Kartensysteme GmbH, Philips Semiconductors Hamburg, Schlumberger Cards Devision, Service Central de la Securite des Systemes d´Information, ST Microelectronics

/BSI-PP-0002/
Title: Smartcard IC Platform Protection Profile
Identification: Registered and Certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-PP-0002
Version: Version 1.0
Date: July 2001
Author: Atmel Smart Card ICs, Hitachi Europe Ltd, Infineon Technologies AG, Philips Semiconductors

/CompPP9806-BSIPP0002/
Title: Assessment on the Substitution of an Evaluation based on PP/9806 by an Evaluation based on BSI-PP-0002-2001
Version: Version 1.1
Date: May 2002
Publisher: Bundesamt für Sicherheit in der Informationstechnik (BSI)

/DS-ICPhilips/
Title: Data Sheet: SmartMX – P5CC036 Secure Smart Card Controller
Version: Revision 3.0
Date: Sept. 21st 2004
Publisher: Philips Semiconductors GmbH

/IS-ICPhilips/
Title: Instruction Set SmartMX-Family, Secure Smart Card Controller, Objective Specification
Version: Revision 1.0
Date: May 9th 2003
Publisher: Philips Semiconductors GmbH

/UG-ICPhilips/
Title: Guidance, Delivery and Operation Manual: Evaluation of the Philips P5CC036V1D Secure Smart Card Controller
Version: Revision 1.0
Date: March 18th 2005
Publisher: Philips Semiconductors GmbH

/ST-ICPhilips/
Title: Security Target - Evaluation of the Philips P5CC036V1D Secure Smart Card Controller
Identification: BSI-DSZ-CC-0293
Version: Version 1.0
Date: March 18th 2005
Publisher: Philips Semiconductors GmbH

/ETRLite-ICPhilips/
Titel: BSI-DSZ-CC-0293: ETR-lite for composition according to AIS 36
Version: Version 1.0
Date: July 6th 2005
Publisher: T-Systems GEI GmbH

/ConfListPhilips/
Title: Customer specific Appendix of the Configuration List for composite evaluation with ORGA (P5CC036V1D)
Version: Version 1.0
Publisher: Philips Semiconductors GmbH

/ISO9796-2/
Title: Information Technology – Security Techniques – Digital Signature Schemes Giving Message Recovery – Part 2: Mechanisms Using a Hash Function
Identification: ISO/IEC 9796-2

Version: First Edition
Date: 1997
Publisher: ISO / IEC

/ISO9798-3/
Title: Information Technology – Security Techniques – Entity Authentication Mechanisms – Part 3: Entity Authentication Using a public key algorithm
Identification: ISO/IEC 9798-3
Version: Second Edition
Date: 1998
Publisher: ISO / IEC

/ISO 7816-4/
Title: Integrated circuit(s) cards with contacts. Part 4: Interindustry commands for interchange
Identification: ISO/IEC 7816-4
Version: First edition
Date: September 1.1995
Publisher: International Organization for Standardization/International Electrotechnical Commission

/ISO 7816-8/
Title: Integrated circuit(s) cards with contacts. Part 8: Interindustry commands for interchange
Identification: ISO/IEC FDIS 7816-8
Date: June 1998
Publisher: International Organization for Standardization/International Electrotechnical Commission

/ISO 7816-9/
Title: Integrated circuit(s) cards with contacts. Part 9: Enhanced interindustry commands
Identification: ISO/IEC 7816-9
Version: First Edition
Date: Sept. 2000
Publisher: International Organization for Standardization/International Electrotechnical Commission

/SHA-1/
Title: Secure Hash Standard (SHS)
Identification: FIPS Publication 180-2
Date: August 2002
Publisher: National Institute of Standards and Technology (NIST)

/FIPS 46-3/
Title: Data Encryption Standard (DES)
Identification: FIPS Publication 46-3

Date: October 1999
Publisher: National Institute of Standards and Technology (NIST)

/ANSI X9.52/
Title: Triple Data Encryption Algorithm Modes of Operation
Identification: ANSI X9.52
Date: 1998
Publisher: American National Standards Institute (ANSI)

/PKCS1/
Title: PKCS #1 v2.1: RSA Cryptography Standard
Date: June 2002
Publisher: RSA Laboratories

/ISO 11770-3/
Title: Information Technology – Security Techniques – Key Management – Part 3: Mechanisms Using Asymmetric Techniques
Identification: ISO/IEC 11770-3
Date: 1996
Publisher: ISO/IEC

/ISO 10118-2/
Title: Information Technology – Security Techniques – Hash Functions – Part 2: Hash Functions Using an n-Bit Block Cipher Algorithm
Identification: ISO/IEC 10118-2
Date: 1994
Publisher: ISO/IEC

/ANSI X9.19/
Title: Financial Institution Retail Message Authentication
Identification: ANSI X9.19
Date: 1996
Publisher: American National Standards Institute (ANSI)

/ANSI X9.63/
Title: Public Key Cryptography for the Financial Services Industry: Key Agreement and Key Transport Using Elliptic Curve Cryptography
Identification: ANSI X9.63
Date: 2001
Publisher: American National Standards Institute (ANSI)

/eHC1/
Title: Die Spezifikation der elektronischen Gesundheitskarte, Teil 1: Kommandos, Algorithmen und Funktionen der Betriebssystem-Plattform

Version: Version 1.1.0
Date: 07.02.2006
Publisher: gematik mbH

/eHC2/
Title: Die Spezifikation der elektronischen Gesundheitskarte, Teil 2: Anwendungen und anwendungsspezifische Strukturen
Version: Version 1.1.0
Date: 07.02.2006
Publisher: gematik mbH

/HPC-SMC1/
Title: German Health Professional Card and Security Module Card, Part 1: Commands, Algorithms and Functions of the COS Platform
Version: Version 2.1.0
Date: 21.02.2006
Publisher: BundesÄrzteKammer, Kassenärztliche Bundesvereinigung, BundesZahnÄrzteKammer, BundesPsychotherapeutenKammer, Kassenzahnärztliche Bundesvereinigung, Werbe- und Vertriebsgesellschaft Deutscher Apotheker mbH

/HPC-SMC2/
Title: German Health Professional Card and Security Module Card, Part 2: HPC Applications and Functions
Version: Version 2.1.0
Date: 21.02.2006
Publisher: BundesÄrzteKammer, Kassenärztliche Bundesvereinigung, BundesZahnÄrzteKammer, BundesPsychotherapeutenKammer, Kassenzahnärztliche Bundesvereinigung, Werbe- und Vertriebsgesellschaft Deutscher Apotheker mbH

/SigG01/
Title: Gesetz über Rahmenbedingungen für elektronische Signaturen und zur Änderung weiterer Vorschriften
Identification: Bundesgesetzblatt Nr. 22, S. 876
Date: 16.05.2001
Publisher: Dtsch. Bundestag

/SigV01/
Title: Verordnung zur elektronischen Signatur
Identification: Bundesgesetzblatt Nr. 509, S. 3074
Date: 16.11.2001
Publisher: Dtsch. Bundestag

/ECDir/

Title: Richtlinie 1999/93/EG des Europäischen Parlaments und des Rates vom 13. Dezember 1999 über gemeinschaftliche Rahmenbedingungen für elektronische Signaturen
Identification: Amtblatt der Europäischen Gemeinschaften, L13/12-L13/20
Date: 19.01.2001
Publisher: Europäisches Parlament und Rat der Europäischen Union

/ALGCAT/
Title: Geeignete Algorithmen zur Erfüllung der Anforderungen nach §17 Abs.1 bis 3 SigG vom 22. Mai 2001 in Verbindung mit Anlage 1 Anschnitt I Nr. 2 SigV vom 22. Nov. 2001
Identification: Bundesanzeiger Nr. 58, S. 1913-1915
Date: 23.03.2006
Publisher: Bundesnetzagentur

/PP-eHC/
Title: Protection Profile – electronic Health Card (eHC) – elektronische Gesundheitskarte (eGK)
Identification: BSI-PP-0020
Version: 1.10
Date: Feb. 16th 2006
Publisher: Bundesamt für Sicherheit in der Informationstechnik (BSI)

/PP-HPC/
Title: Protection Profile – Health Professional Card (HPC) – Heilberufsausweis (HBA)
Identification: BSI-PP-0018
Version: 1.1
Date: April 2nd 2007
Publisher: Bundesamt für Sicherheit in der Informationstechnik (BSI)

/PP-SMC/
Title: Protection Profile – Security Module Card (SMC)
Identification: BSI-PP-0019
Version: 1.0
Date: Feb. 1st 2006
Publisher: Bundesamt für Sicherheit in der Informationstechnik (BSI)

/PP SSCD Type3/
Title: Protection Profile – Secure Signature-Creation Device Type 3 “EAL 4+”
Identification: BSI-PP-0006-2002
Version: Version 1.05
Date: July 25th 2001
Publisher: CEN/ISSS – Information Society Standardization System, Workshop on Electronic Signatures

/PP SSCD Type2/

Title: Protection Profile – Secure Signature-Creation Device Type 2 "EAL 4+"
Identification: BSI-PP-0005-2002
Version: Version 1.04
Date: July 25th 2001
Publisher: CEN/ISSS – Information Society Standardization System, Workshop on Electronic Signatures

/ST-MIC30/
Title: Security Target – MICARDO V3.0 R1.0
Identification: 3MIC3EVAL.CST.0002
Version: V1.00
Author: Dr. S. Pingel
Publisher: Sagem ORGA GmbH

II Summary of abbreviations

A.x	Assumption
AC	Access Condition
AID	Application Identifier
ALW	Always
AM	Access Mode
AR	Access Rule
AS	Application Software
ATR	Answer To Reset
AUT	Key Based Authentication
BS	Basic Software
CC	Common Criteria
CGA	Certification Generation Application
CH	Card Holder
CHV	Cardholder Verification
CSP	Certification Service Provider
DES	Data Encryption Standard
DF	Dedicated File
DFA	Differential Fault Analysis
DPA	Differential Power Analysis
DTBS	Data to be signed
EAL	Evaluation Assurance Level
EF	Elementary File
EHC	Electronic Health Card
ES	Embedded Software
HPC	Health Professional Card
IC	Integrated Circuit
IFD	Interface Device
MAC	Message Authentication Code
MF	Master File
O.x	Security Objective
OS	Operating System
PAR	Partial Access Rule
P.x	Organisational Security Policy

PIN	Personal Identification Number
PP	Protection Profile
PUC	PIN Unblocking Code
PW	Password
PWD	Password Based Authentication
RAD	Reference Authentication Data
RSA	Rivest-Shamir-Adleman Algorithm
SAR	Security Assurance Requirement
SCA	Signature Creation Application
SCD	Signature Creation Data
SCS	Signature Creation System
SDO	Signed Data Object
SFP	Security Function Policy
SFR	Security Functional Requirement
SM	Secure Messaging
SMC	Security Module Card
SOF	Strength of Functions
SPA	Simple Power Analysis
SPM	TOE Security Policy Model
SSC	Send Sequence Counter
SSCD	Secure Signature Creation Device
ST	Security Target
SVD	Signature Verification Data
TA	Timing Analysis
T.x	Threat
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy
VAD	Verification Authentication Data

III Glossary

For explanation of technical terms refer to the following documents:

/PP9911/, Annex A

/BSI-PP-0002/, Chap. 8.7

Appendix

Mapping SigG / SigV – TOE Sicherheitsfunktionen

#	Anforderungen aus SigG / SigV	Referenz	Relevante TSFs des EVG
1	(1) Für die Speicherung von Signaturschlüsseln sowie für die Erzeugung qualifizierter elektronischer Signaturen sind sichere Signaturerstellungseinheiten einzusetzen, die Fälschungen der Signaturen und Verfälschungen signierter Daten zuverlässig erkennbar machen und gegen unberechtigte Nutzung der Signaturschlüssel schützen. Werden die Signaturschlüssel auf einer sicheren Signaturerstellungseinheit selbst erzeugt, so gilt Absatz 3 Nr. 1 entsprechend.	/SigG01/, §17 „Produkte für qualifizierte elektronische Signaturen“, (1)	Eine Nutzung des Signaturschlüssels der Signaturapplikation der sicheren Signaturerstellungseinheit „MICARDO V3.0 R1.0 HPC V1.0“ ist nur nach erfolgreicher PIN-basierter Authentisierung des Nutzers möglich (Identifikation durch Besitz und Wissen). Die Sicherung des Signaturschlüssels und seiner Nutzung ist Gegenstand von TSF F.ACS_SFP (Zugriffskontrolle) und F.IA_PWD (Prozesse der PIN-basierten Authentisierung). Pro PIN-Verifikation ist alternativ entweder nur eine Signaturerzeugung möglich oder aber beliebig viele Signaturen können erzeugt werden. Die Auswahl der Variante erfolgt im Rahmen der Personalisierung des Produktes. Die Generierung des Signaturschlüsselpaares der Signaturapplikation der sicheren Signaturerstellungseinheit „MICARDO V3.0 R1.0 HPC V1.0“ erfolgt ausschließlich on-card. Die Anforderungen an die Qualität des Generierungsprozesses werden in TSF F.RSA_KEYGEN, F.SIDE_CHAN, F.CRYPTO und F.RIP umgesetzt. Die Schlüsselgenerierung findet ausschließlich im Rahmen der Personalisierung des Produktes (unter den in der User Guidance für den Personalisierer angegebenen Auflagen) statt. Insbesondere ist aufgrund der gesetzten Zugriffsregeln keine erneute Schlüsselgenerierung im Wirkbetrieb des Produktes möglich (TSF F.ACS_SFP). Die Sicherheit des Prozesses der Signaturerzeugung, insbesondere bzgl. der Gewinnung von Informationen über den benutzten Signaturschlüssel, wird über TSF F.GEN_DIGSIG, F.CRYPTO, F.SIDE_CHAN und F.RIP sichergestellt. Insbesondere sorgen die genannten TSF dafür, dass Fälschungen von Signaturen und Verfälschungen signierter Daten erkennbar gemacht werden.
2	(3) Die technischen Komponenten für Zertifizierungsdienste müssen Vorkehrun-	/SigG01/, §17 „Produkte für qualifizierte	Siehe Erklärungen zu Tabellenzeile 1.

	gen enthalten, um 1. bei Erzeugung und Übertragung von Signaturschlüsseln die Einmaligkeit und Geheimhaltung der Signaturschlüssel zu gewährleisten und eine Speicherung außerhalb der sicheren Signaturerstellungseinheit auszuschließen, ...	elektronische Signaturen“, (3), Satz 1	
3	(1) Sichere Signaturerstellungseinheiten nach § 17 Abs. 1 Satz 1 des Signaturgesetzes müssen gewährleisten, dass der Signaturschlüssel erst nach Identifikation des Inhabers durch Besitz und Wissen oder [...] angewendet werden kann. Der Signaturschlüssel darf nicht preisgegeben werden. [...] Die zur Erzeugung und Übertragung von Signaturschlüsseln erforderlichen technischen Komponenten nach § 17 Abs. 1 Satz 2 oder Abs. 3 Nr. 1 des Signaturgesetzes müssen gewährleisten, dass aus einem Signaturprüfschlüssel oder einer Signatur nicht der Signaturschlüssel errechnet werden kann und die Signaturschlüssel nicht dupliziert werden können.	/SigV01/, §15 „Anforderungen an Produkte für qualifizierte elektronische Signaturen“, (1)	Eine Nutzung des Signaturschlüssels der Signaturapplikation der sicheren Signaturerstellungseinheit „MICARDO V3.0 R1.0 HPC V1.0“ ist ausschließlich nach erfolgreicher PIN-basierter Authentisierung des Nutzers möglich (Identifikation durch Besitz und Wissen). Die Nutzung biometrischer Merkmale zur Authentisierung des Nutzers ist nicht implementiert. Die Sicherung des Signaturschlüssels und seiner Nutzung ist Gegenstand von TSF F.ACS_SFP (Zugriffskontrolle) und F.IA_PWD (Prozesse der PIN-basierten Authentisierung). Ein direktes Auslesen des Signaturschlüssels über die regulären Betriebssystem-Kommandos ist aufgrund der gesetzten Zugriffsregeln ebenfalls nicht möglich (TSF F.ACS_SFP). Die Generierung des Signaturschlüsselpaares der Signaturapplikation der sicheren Signaturerstellungseinheit „MICARDO V3.0 R1.0 HPC V1.0“ erfolgt ausschließlich on-card. Die Anforderungen an die Qualität des Generierungsprozesses werden in TSF F.RSA_KEYGEN, F.SIDE_CHAN, F.CRYPTO und F.RIP umgesetzt. Die Schlüsselgenerierung findet ausschließlich im Rahmen der Personalisierung des Produktes (unter den in der User Guidance für den Personalisierer angegebenen Auflagen) statt. Insbesondere ist aufgrund der gesetzten Zugriffsregeln keine erneute Schlüsselgenerierung im Wirkbetrieb des Produktes möglich (TSF F.ACS_SFP). Die Sicherheit des Prozesses der Signaturerzeugung, insbesondere bzgl. der Gewinnung von Informationen über den benutzten Signaturschlüssel, wird über TSF F.GEN_DIGSIG, F.CRYPTO, F.SIDE_CHAN und F.RIP sichergestellt.
4	(4) Sicherheitstechnische Veränderungen an technischen Komponenten nach den Absätzen 1 bis 3 müssen für den Nutzer	/SigV01/, §15 „Anforderungen an Produkte für	Die sichere Signaturerstellungseinheit „MICARDO V3.0 R1.0 HPC V1.0“ beinhaltet geeignete Sicherungsmechanismen,

	erkennbar werden.	qualifizierte elektronische Signaturen“, (4)	die einen sicheren Betriebszustand des Produktes garantieren und dem Nutzer (direkt oder indirekt, je nach Fehlerfall) Information hierüber geben. Die Sicherungsmechanismen werden in TSF F.FAIL_PROT, F.SELFTEST und F.SIDE_CHAN realisiert.
5	Restriktionen zur PIN-/PUK-Funktionalität	---	Die Signaturapplikation der sicheren Signaturerstellungseinheit „MICARDO V3.0 R1.0 HPC V1.0“ sieht folgende Restriktionen für die dem Signaturschlüssel zugeordnete Signatur-PIN (PIN.QES) vor: - Initialwert für den Fehlbedienungszähler: 3 - Mindestlänge der PIN: 6 Ziffern - Nutzung des Transport-PIN Verfahrens (Länge der Transport-PIN: 5 Ziffern, Wechsel der Transport-PIN über das Kommando CHANGE REFERENCE DATA notwendig vor erster Nutzung des Signaturschlüssels, d.h. vor erster erfolgreicher PIN-Verifikation über das Kommando VERIFY) - Verwendung einer PUK (Resetting Code) zum Freischalten einer gesperrten Signatur-PIN Für die der Signatur-PIN zugeordnete PUK sieht die Signaturapplikation folgende Restriktionen vor: - Keine Verwendung eines Fehlbedienungszählers - Initialwert für den Bedienungszähler: 10 - Länge der PUK: 8 Ziffern - Jeder Zugriff auf die PUK dekrementiert den zugehörigen Bedienungszähler. - Variante für RESET RETRY COUNTER: ohne Wechsel der Signatur-PIN, kein Setzen des Sicherheitszustandes der Signatur-PIN
6	Restriktionen zur Nutzung der Display-Message	---	Die Signaturapplikation der sicheren Signaturerstellungseinheit „MICARDO V3.0 R1.0 HPC V1.0“ verwendet ein Datenfeld für die Display-Message. Eine Änderung der Display-Message erfordert aufgrund der gesetzten Zugriffsregeln die erfolgreiche PIN Verifikation mit der PIN PIN.CH der HPC Karte. Die PIN PIN.CH ist ein von der Signatur-PIN PIN.QES zur Sicherung des Signaturschlüssels verschiedenes

			Objekt.
7	(5) ... Bei der Prüfung und Bestätigung der Sicherheit von Produkten nach § 17 Abs. 1 und 3 Nr. 1 des Signaturgesetzes sind die Vorgaben des Abschnitts II der Anlage 1 zu dieser Verordnung zu beachten.	/SigV01/, §15 „Anforderungen an Produkte für qualifizierte elektronische Signaturen“, (5)	Siehe Erklärungen in den folgenden Tabellenzeilen 8 - 10.
8	Die Prüfung der Produkte für qualifizierte elektronische Signaturen nach Maßgabe des § 15 Abs. 7 und des § 17 Abs. 4 des Signaturgesetzes hat nach den „Gemeinsamen Kriterien für die Prüfung und Bewertung der Sicherheit von Informationstechnik“ (Common Criteria for Information Technology Security Evaluation, BAnz. 1999 S. 1945, – ISO/IEC 15408) oder nach den „Kriterien für die Bewertung der Sicherheit von Systemen der Informationstechnik“ (ITSEC – GMBl vom 8. August 1992, S. 545) in der jeweils geltenden Fassung zu erfolgen. Die Prüfung muss ... b) bei sicheren Signaturerstellungseinheiten nach § 2 Nr. 10 des Signaturgesetzes mindestens die Prüftiefe EAL 4 oder E 3 umfassen, ...	/SigV01/, Anlage 1, I, 1.1 „Anforderungen an Prüftiefen“	Die sichere Signaturerstellungseinheit „MICARDO V3.0 R1.0 HPC V1.0“ unterliegt einer Evaluierung und Zertifizierung nach dem Standard Common Criteria Version 2.3 mit dem Evaluierungslevel EAL 4+ (mit den Augmentierungen ADV_IMP.2, ATE_DPT.2, AVA_MSU.3 und AVA_VLA.4) und SOF Hoch.
9	Bei den Prüfstufen „EAL 4“ und bei „EAL 3“ gemäß Abschnitt I Nr. 1.1 Buchstabe a bis c i) und Buchstabe d ist ergänzend zu den bei dieser Prüfstufe vorgeschriebenen Maßnahmen gegen ein hohes Angriffspotenzial zu prüfen und eine vollständige Missbrauchsanalyse durchzuführen. ...	/SigV01/, Anlage 1, I, 1.2 „Anforderungen an Schwachstellenbewertung / Mechanismenstärke“	Die sichere Signaturerstellungseinheit „MICARDO V3.0 R1.0 HPC V1.0“ unterliegt einer Evaluierung und Zertifizierung nach dem Standard Common Criteria Version 2.3 mit dem Evaluierungslevel EAL 4+ (mit den Augmentierungen ADV_IMP.2, ATE_DPT.2, AVA_MSU.3 und AVA_VLA.4) und SOF Hoch.
10	Die Algorithmen und zugehörigen Parameter müssen nach Abschnitt I Nr. 1.2 dieser Anlage als geeignet beurteilt sein.	/SigV01/, Anlage 1, I, 1.3 „Anforderungen an Algorithmen“	Die sichere Signaturerstellungseinheit „MICARDO V3.0 R1.0 HPC V1.0“ berücksichtigt für die Signaturerzeugung, Hashwert-Berechnung, Zufallszahlengenerierung und Schlüsselgenerierung Algorithmen und Parameter, die dem aktuellen Algorithmenkatalog /ALGCAT/ entsprechen. Vergleiche hierzu die TSFs F.GEN_DIGSIG, F.RSA_KEYGEN und F.CRYPTO.