Security Target lite
Electronic Health Card 6.20
GEGKOS

GEMALTO, B.P. 100, 13881 GEMENOS CEDEX, FRANCE.

Tel: +33 (0)4.42.36.50.00 Fax: +33 (0)4.42.36.50.90

Printed in France.

Document Reference: ASE11R10559 V7.3

October 06, 2011

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. ST INTRODUCTION

1.1 ST REFERENCE

Title:	ASE - Security Target Electronic Health Card
Reference:	ASE01R10559 version : 7.3 date : 06/10/11
Origin:	GEMALTO

Table 1 : ST References

This Security Target describes:

- The Target Of Evaluation, the TOE components, the components in the TOE environment, the product type, the TOE environment and life cycle, the limits of the TOE.
- The Assets to be protected and the threats to be countered by the TOE itself during the usage of the TOE.
- The security objectives for the TOE and its environment
- The TOE security assurance requirements
- The security functions and the assurance measures

This ST has been built with the:
Common Criteria for Information Technology Security Evaluation Version 3.1, July 2009 which comprises [CCPART1], [CCPART2], and [CCPART3].

1.2 TOE REFERENCE

Product and TOE are completely defined by information located in the following table.

Product Name	GEGKOS
Product Version	A6
TOE name	Electronic Health Card
TOE Version	6.20
Micro Controller	Infineon SLE78CX800P

Table 2: TOE References

1.3 TOE OVERVIEW

1.3.1 TOE type

The TOE " **Electronic Health Card "** is the smart card IC with Embedded Software.
.

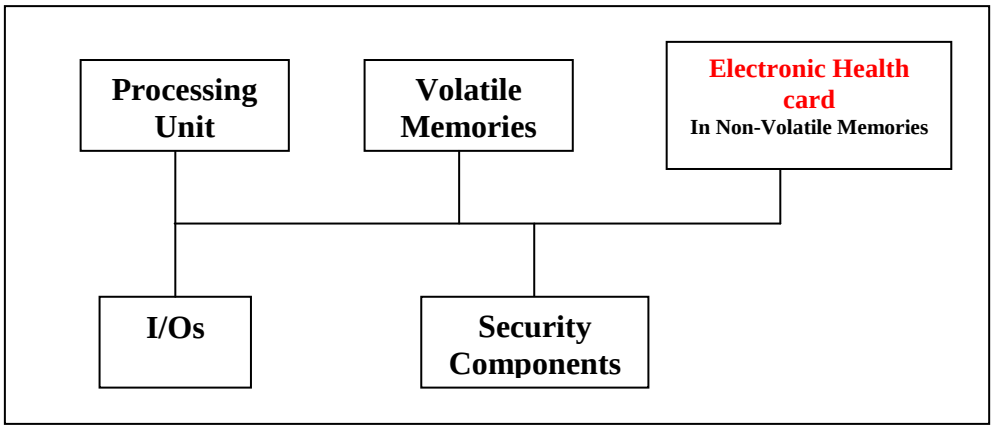


Figure 1 – Smart card IC with Embedded Software

The Smart Card Integrated circuit is the INFINEON SLE78CX800P micro-controller. The evaluation of the **Electronic Health Card** is built on the results of the evaluation of the SLE78CX800P.

1.3.2 Intended Use and Major Security Features

The Target of Evaluation (TOE) is a smart card, the electronic Health Card (eHC), which is conformant to the specification documents: "The specification of the German Electronic Health Card eHC" Part1,2 . [EHC spec part 1], [EHC spec part 2]
The size of the card is type ID-1 according to ISO 7810 (the usual credit-card-size).
The card is a card with contacts according to ISO 7816-1 to –3.

The TOE contributes to the Health application management by providing the following services:

- Mutual Authentication between the eHC and the Health Professional Card (HPC) or a Security Module Card (SMC)Mutual Authentication between the eHC and a security device (e. g. for online update of contract data in the card),
- Authentication of the card holder by use of one or two PINs (PIN.CH and PIN.home : Specific PINs for eHC functions)
- Secure storage of contractual and medical data, with respect to confidentiality, integrity and authenticity of these data
- Authentication of the card using private key and X.509 certificate
- Document content key decipherment using a private key
- Management of applications
- File content protection via access conditions driven by ES.

- Confidentiality of the PINs and the cryptographic keys.
- Integrity of PIN, cryptographic keys, and of file contents.

The services mentioned are implemented with following cryptography:

- 3TDES, that is Triple DES using 168 bit symmetric keys.
- RSA with key size of 2048 bit.
- hashing with SHA-256 The hash value can be transmitted directly to the card, computed completely by the TOE, or computed partly by the TOE.

To ensure the correct operation of the GeGKOS mechanisms the TOE implements following security features:

- A life cycle with secure production steps as specified in [GeGKOS_PERS]. This includes authentication mechanisms and secured communication protocols for administrators in productive phases.
- Storage of TOE data along with checksums to ensure integrity.
- Integrity and confidentiality of the embedded software (ES).
- TOE self protection by software design and utilization of the IC security features. For more details see § **7.11**.

With the mechanisms above the TOE protects the assets described in section § **3.1.1** by fighting the following risks:

- Cloning: Substitution of programmed microchip i.e personalized or non-personalized Smart Card.
- Confidential data disclosure: Disclosure of confidential data in programmed microchip, i.e. Application code, keys, PINs.
- Non-integrity: Use of non-valid data.
- Identity usurpation: Management (i.e. personalization,) by unauthorized administrators. Use of Application by unauthorized user, i.e. other than the legitimate one.
- Physical attacks : the physical tampering of the TOE user data, TSF data or by modification of security features
- Information leakage : as emanations, variations in power consumption, I/O characteristics, clock frequency or by changes in processing requirements
- Malfunction due to an environment stress
- Use of functions in wrong phase to manipulate TOE’s security functions or features or TSF data

1.4 TOE DESCRIPTION

1.4.1 TOE definition

The TOE comprises the following parts
TOE_IC, consisting of:

- the circuitry of the eHC's chip (the integrated circuit, IC) and
- the IC Dedicated Software with the parts IC Dedicated Test Software and IC Dedicated Support Software

TOE_ES,

- the operating system, branded GeGKOS ("Gemalto Elektronische GesundheitsKarte Operating System")

TOE_APP,

- the eHC applications (data structures and their content, not including card individual data like PIN and key values)

and
guidance documentation delivered together with the TOE.

1.4.2 Global Description

In essence the TOE consists of the electronic health applications that are instantiated on the Gemalto implementation of the GeGKOS operating system. This operation system in turn resides on the certified Infineon SLE78CX800P contact interface micro-controller.

Therefore the TOE is a composed one, containing the following components for this composite evaluation:

Component	Supplier
Embedded software and data structures for the eHC Applications (TOE_ES plus TOE_APP)	Gemalto
Micro-controller	Infineon

Table 3 - TOE components

1.4.3 Electronic Health Applications Description

With these applications the TOE forms an Electronic Health Card as described in the Protection Profile [PP eHC].

The TOE contributes to the electronic health application by providing the following mechanisms:

- Identity data or contractual data protection.
- "Verification Authentication Data" : check the PIN codes or a resetting code entered to activate certain functions of the TOE
- Store data as the "Reference Authentication Data" , initialisation data, personalisation data, logging data , emergency data
- MAC calculation and encryption with symmetric keys inside a trusted channel (TC)
- Management of the medical data (including the emergency data) through the voluntary application
- Authentication of the card holder by use of the PIN.CH or PIN.home
- Authentication of components (HPC/SMC) of health professional or Medical assistant (accredited)
- Authentication of the health insurance agency service provider
- Authentication of the self service terminal
- Authentication of the card with a client-server authentication private key
- Deciphering document content keys with a private keyConfidentiality of keys and PINs: client-server authentication private key, decipher private key, card authentication private key, PIN.CH, PIN.home

1.4.4 Operating System Description

The GEGKOS operating system (TOE_ES) meets the specification **[EHC spec part 1]**.The Applicative Data Structures, **Health application,** meet the specification **[EHC spec part 2]**.

These specifications are defined according [ISO C4], [ISO C4'], [ISO C8], [ISO C9], and [PKCS1] standards.

The OS provides the following functions:

- a file system according to [ISO C4],
- access control for the file system and the cryptographic services,
- secure messaging for external communication via a trusted channel (TC),
- selection and management of security environments;
- user authentication with passwords,
- component authentication with symmetric and asymmetric cryptographic keys,
- import of external public keys via CVC verification
- creation and verification of digital signatures,
- enciphering and deciphering with asymmetric cryptography.

The data structures of the ADFs determine the access to those functions and their execution modes by containing the appropriate access conditions and control information, e.g. key lengths or maximum PIN retry counters.

The TOE consists of the following software modules:

The APDU Manager

- For this TOE the APDU commands are defined in the specification **[eHC spec part 1]**

The Access Manager

- accesses the file system to find the relevant access rules for the command to be executed and the data to be accessed, This provides full control over the TOE assets like applicative data, PINs, and keys.
- checks if authentication and Secure Messaging has occurred as requested by the access conditions.

The Access Enabling Mechanisms

This module includes:

- Authentication by human users and external components,
- Secure Messaging.

The File System

The File System manages Data structured in DFs and EFs.
All persistent data of the electronic health applications (including PINs and keys) are stored in the file system.

The Cryptographic Computations

This is the package of cryptographic algorithms directly available at APDUs or used for the access Enabling Mechanisms.

A cryptographic library internally developed by GEMALTO supplies the basic cryptographic functionalities needed for these OS components, utilizing the chip's cryptographic co-processors:

- cryptographic algorithms based on 3-DES (key size 24 bytes = 3 parts of 56 bits),
- cryptographic algorithms based on RSA (key size 2048 bits),
- Hash algorithms (SHA-256)[1] ,

1.4.5 TOE security features

TOE implements following security features:

- All data in non-volatile memory (especially keys and PINs) are equipped with a checksum to detect integrity faults.
- The data structure of the card is a hierarchical file system. For a given eHC application, applicative data are not accessible from outside of the current application (DF).The file system is a built-in way to establish data separation between different applications.
- After start-up, the integrity of code patches is verified.
- Self protection by software design features as checking hardware registers, desynchronization, redundancy, usage of platform's protection and self test features like clock jitter or environmental sensors, sensitive data masked.
- In case of an application deletion the associated memory area is deleted.

1

Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung (20th may 2011)
Hash functions :
Suitable until end 2015 SHA-224 (SHA-1, RIPEMD-160)***
Suitable until end 2017 SHA-256, SHA-384, SHA-512
*** exclusively for the verification of qualified certificates.

1.4.6 Hardware Platform

The TOE contains software and hardware identified as Infineon SLE78CX800P and certified by BSI with the certificate reference [BSI-DSZ-CC-0606-2010] (Confimation of the reassessment - 17 May 2011). The IC is compliant with the [BSI-CC-PP-0035-2007]. The IC is certified at the level EAL5 augmented with ALC_DVS.2, AVA_VAN.5 components.
The Infineon SLE78CX800P provides algorithms to the embedded software as Triple Data Encryption Standard (3-DES), Rivest-Shamir-Adleman Cryptography (RSA) and Secure Hash Algorithm (SHA-256) but they are not used by the composite TOE.These algorithms are developed by Gemalto using hardware accelerators for cryptographic computation and are part of the embedded software,

This certified IC is described in the platform's Security Target [ST IC].
Besides state of the art attack resistance this IC provides a P2-TRNG (AIS31) SOF-high. The TOE_ES uses this TRNG for cryptographic computations.

1.4.7 TOE Boundaries

The following figures illustrate the TOE physical and logical boundaries.

The product is a smartcard including a plastic card and a module performing the interface between reader and the embedded chip. The Target of Evaluation (TOE) is the Smart Card Integrated Circuit with Embedded Software and data structures in operation and in accordance to its functional specifications. Other smart card product items (such as plastic, module, security printing…) are outside the scope of this evaluation.

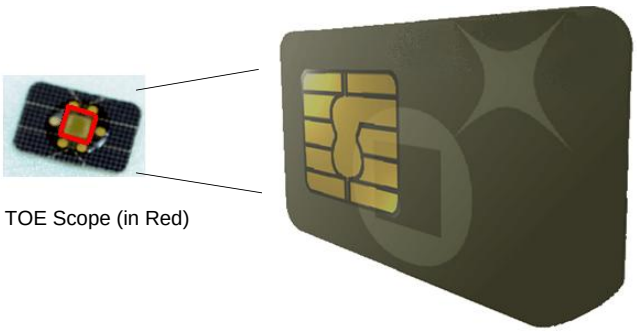


Figure 2 – TOE Physical Boundaries

Figure 3 describes how the Applications and the GEGKOS operating system are implemented on the IC. It describes the global architecture of the **Electronic Health Card .**

The physical scope of the TOE is the complete card framed by the grey line. The logical scope is highlighted in yellow. It is the chip with the embedded software and the data structures of the electronic health applications in EEPROM.

All the software modules are included inside the TOE (see the ***TOE enforcing element***). This software uses the hardware and its firmware to provide the TOE functionality. The hardware and its firmware is part of the TOE.

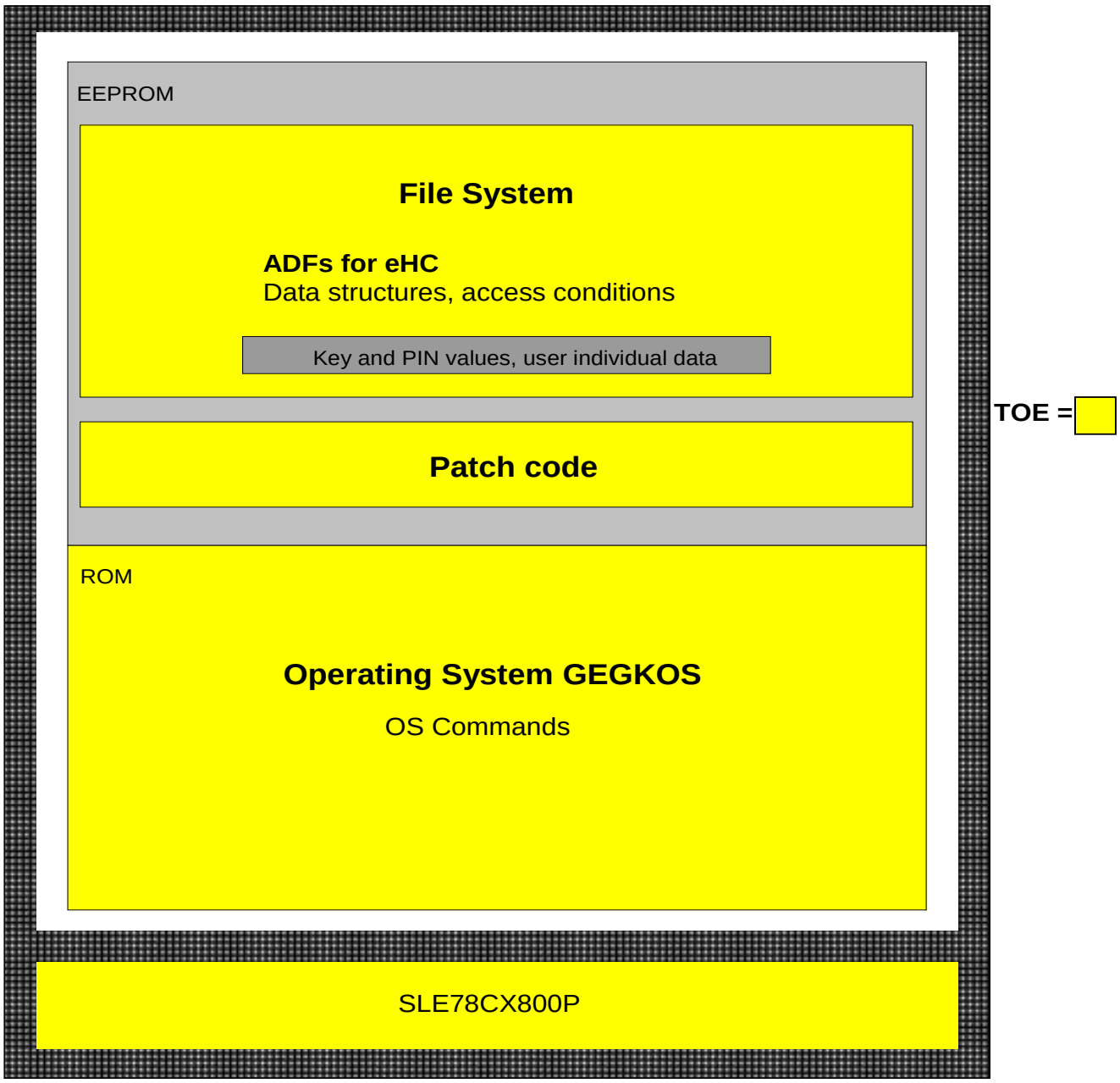


Figure 3 – TOE logical boundaries

The TOE is made of the chip, the embedded Software and the data structures in EEPROM, including the ADFs (Application DFs) for the applications under evaluation (described in **[EHC spec part 2]**).
By specification it is possible to create additional applications after card issuance, consequently there are parts in EEPROM outside the TOE scope (grey). Note that this mechanism is not able to influence the existing applications!
The ADFs cover all containers for the applicative data, including access conditions and OS dependent system data contents. Card individual data like PIN and key values are outside the scope of the TOE.

1.4.8 TOE Life Cycle and TOE Actors

A Smart Card's life cycle is decomposed in several phases.

Each life cycle phase is linked to certain TOE actors. This is shown in table Table 4 - TOE life cycle. Further details of the single phases follow in the subsections below

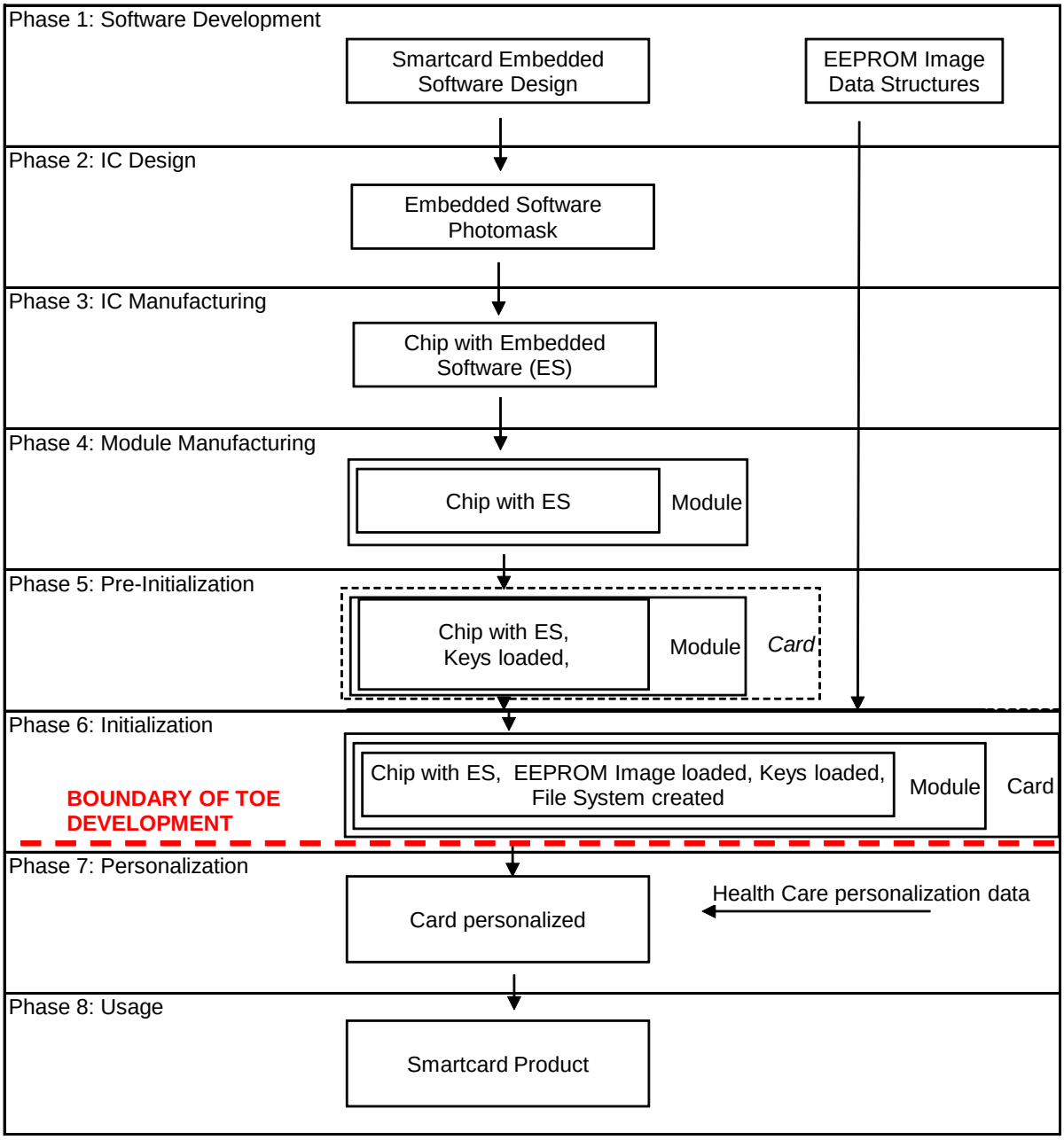


Figure 4 –Electronic Health Card Lifecycle

The following table presents the TOE actors, and logical phase associated with each step of the life cycle

Phase	TOE phase	Industrial deliverable	TOE actors
1	Software Development	ROM code and EEPROM image	Product developer
2	IC Design	Hardmask set	IC manufacturer
3	IC Manufacturing	Wafers with ICs	IC manufacturer
4	Module Manufacturing	Modules	Module manufacturer
5	Pre-initialization	Modules or Card pre-initialized	Card manufacturer
6	Initialization	Card initialized	Card manufacturer
7	Personalization	Card personalized	Personalizer
8	Usage	Smartcard	Card issuer : Health insurance, TOE user : Card issuer, End user, Terminal

Table 4 - TOE life cycle

1.4.9 TOE delivery

Phase 6 completely belongs to the TOE development, i.e. the TOE is delivered as an IC already embedded in the plastic card and containing all software and at least the data structures as defined in the specification [EHC spec part 1] and [EHC spec part 2].

The TOE will be delivered :

1. Documentation:
   - Administrator Guidance
   - User Guidance
   - Life Cycle Description for Electronic Health Card
2. HW-Part of TOE:
   - Chip modules embedded into smart cards with Infineon SLE78CX800P, (ROM mask of the TOE already implemented)
3. SW Part of the TOE:
   - EEPROM image with application DF Health Care (data structures).

1.4.10 TOE actors summary

The TOE actors as mentioned in the subsection of TOE Life Cycle are summarized in the following, categorized as Administrator or End user.

Administrators

The TOE administrators are listed below:

Administrators	Description
Product Developer	The Product developer designs the IC ES. For this product, the developer is **GEMALTO (phase 1)**.
IC Manufacturer	The IC manufacturer -or founder- designs, manufactures, and loads the ES in the IC. For this product, the IC manufacturer is **INFINEON (phase 2 & 3)**.
Module Manufacturer	The module manufacturer processes the ICs to modules. For this product, the module manufacturer is **GEMALTO (phase 4)**.
Card Manufacturer	The Card manufacturer is responsible: • For embedding the modules provided by the module manufacturer into Smart Cards (phase 5 or phase 6) • for pre-initialization of the Smart Cards (loading card serial number and secret keys for the initialization and personalization phases)(phase 5) • for initialization of smart cards (loading EEPROM image) (phase 6) For this product, the Card manufacturer is **GEMALTO (phase 5 & 6) or SYSTEMFORM MEDIACARD (for initialization phase 6).**.
Personalizer	The Personalizer personalizes the card by loading the Card issuer and End user data as well as Application secrets such as cryptographic keys and PIN. The personalization includes printing of the card holder specific visual readable data onto the physical smart card. For this product, the Personalizer is **GEMALTO or SYSTEMFORM MEDIACARD (phase 7).**
Card issuer	The Card issuer -short named « issuer » issues cards to its customers that are the « End users ». The card belongs to the Card issuer. Therefore, the Card Issuer is responsible for: • Personalization of the data • Distribution of the cards. • Maintenance of the cards (i.e. unblocking the PIN) • Invalidation of the cards. For this product, the Card Issuer are **Health insurance agencies (phase 8)**.

Table 5 –Administrators list

End users

The TOE end users are listed below:

Users	Description
End user	The End user (or cardholder) is a customer of the Card issuer. The card is personalized with the End user identification and secrets
Terminals	In the operational usage phase, the Electronic Health Card communicate through terminals : ⇒ Health professional terminals (read and write operations) ⇒ eKiosk self service terminals (read and write operations), ⇒ private PC (read only operations)

Table 6 –Users list

2. CONFORMANCE CLAIMS

2.1 CC CONFORMANCE CLAIMS

This security target claims to be conformant to the Common Criteria version 3.1, which comprises of:

- Common Criteria for Information Technology Security Evaluation (CC), V3.1, Part 1: Introduction and general model, Revision 3, July 2009 [CCPART1].
- Common Criteria for Information Technology Security Evaluation (CC), V3.1, Part 2: Security functional components, Revision 3, July 2009 [CCPART2].
- Common Criteria for Information Technology Security Evaluation (CC), V3.1, Part 3: Security assurance components, Revision 3, July 2009 [CCPART3].
- Common Methodology for Information Technology Security Evaluation [CEM], V3.1, Revision 3, July 2009,

as follows:

- Part 2 extended with
  - FPT_EMSEC TOE emanation
  - FCS_RND Quality metric for random numbers
  - FMT_LIM limited capabilities and availability
- Part 3 conformant

The evaluation is performed according [CEM] and supporting documents [AIS 36].

2.2 PP CLAIM

This ST claims strict conformance to [PP eHC] .

The TOE includes an Integrated Circuit certified with CC EAL5 augmented with ALC_DVS.2 and AVA_VAN.5.

2.3 PACKAGE CLAIMS

This ST is conformant to the EAL4 package as defined in [CCPART3].

The assurance level is EAL4 augmented with:

AVA_VAN.5 Advanced methodical vulnerability analysis

2.4 CONFORMANCE RATIONALE

This ST is claimed to be conformant to the above mentioned PP [PP eHC]. A detailed justification is given in the following by

- describing some single aspects which are main issues of PP conformance, and
- describing differences between the ST and the PP.

2.4.1 Main aspects

- The TOE description in section 1.3 is based on the TOE overview of [PP eHC, §1.2] and specific informations linked to the product have been added.
- All definitions of the security problem definition in [PP eHC, §3] have been included in the ST in the same wording.
- All definitions of the security objectives in [PP eHC, §4] have been included exactly in the same wording as the PP.
- The SFR defined in the extended components definition of [PP eHC, §5] has been included in the ST exactly in the same wording as the PP.
- All SFRs for the TOE from the [PP eHC, §5] have been included in the ST exactly in the same wording as the PP and filling all necessary selections or assignments.
- Text from introduction, TOE overview, TOE description has been taken from the PP and specific information linked to the product have been added.
- The security assurance requirements (SARs) are originally taken from SARs of CC 3.1 Part 3 according to the package conformance EAL 4 augmented with AVA_VAN.5.
- The structure of the ST is taken from the PP added by the section 7 (TOE summary specification) and section 8 (Statement of Compatibility concerning Composite Security Target).

2.4.2 Differences between ST and PP

The ST updates one SFR to those of the PP :
FDP_SDI is repeated with varying operation

- FDP_SDI.2/Persistent : integrity checked persistent stored data:
- FDP_SDI.2/Volatile : integrity checked volatile data

3. SECURITY PROBLEM DEFINITION

3.1 GENERAL

The Security Problem Definition (SPD) is the part of the ST , which describes :

- Assets, wich the TOE shall protect
- Subjects, who are users (human or system) of the TOE or who might be threats agents (i.e attack the security of the assets)
- Operational security policies, which describe overall security requirements defined by the organization in charge of the overall system including the TOE (in particular this may include legal regulations, standards and technical specifications)
- Threats against the assets, which shall be averted by the TOE together with its environment,
- Assumptions on security relevant properties and behavior of the TOE's environment.

3.1.1 Assets and objects

Personal and health insurance data (open) EF PD, EF.StatusVD, EF.VD	Identity data or contractual data, which can be read without authentication
Personal and health insurance data (protected) EF GVD	Identity data or contractual data, which can be read only with authentication
VAD (eHC)	"Verification Authentication Data": PIN codes or a resetting code entered by a card holder to activate certain functions of the TOE.
RAD (eHC) PIN.CH, PIN.home	"Reference Authentication Data": The PINs and corresponding resetting code values stored in the TOE and used for comparison with the VAD entered by the card holder.
Initialisation data	All data stored in the TOE during the initialisation process.
Personalisation data	All data stored in the TOE during personalisation process.
Logging data (EF Logging)	Data stored in the TOE in order to document the last fifty accesses to medical data by care providers.
Card Authentication Private Key PrK.eGK.AUT_CVC	The Card Authentication Private Key is a asymmetric cryptographic key used for the authentication of an eHC to a HPC, to a SMC or to a service provider.
Card Verifiable Authentication Certificate MF/EF.C…	These data include Card verifiable certificates of the Card Authentication Public Key as authentication reference data corresponding to the Card Authentication Private Key and used for the card-to-card authentication. They contain encoded access rights (Role ID) and are signed by a certificate provider on behalf of the card issuer. In addition these data contain a certificate for the CA used in the case of two-step certificate verification. These data are part of the user data provided for use by external entities as authentication reference data of the eHC.
Client-Server Authentication Private Keys PrK.CH.AUT, PrK.CH.AUTN.	The Client-Server Authentication Private Keys are asymmetric cryptographic keys used for the authentication of a client application acting on behalf of the card holder to a server.
Decipher Private Keys PrK.CH.ENC PrK.CH.ENCV	The Document Cipher Key Decipher Keys are asymmetric private keys used for document decryption on behalf of the card holder.
Display message EF.DM	A display message is used as a means for the card holder to check if a secure channel is established.
X.509 certificates EF.C.CH	The certificates for the keys used in the context of Service_Client_Server_Auth and Service_Data_Decryption. These certfificates are provided by the card to other entities, who want to verify the validity of the card's keys used for these services.
Public Key for CV Certificate Verification PUK.RCA.CS	Public keys of Certification Authorities used for verification of the card verifiable certificates.

Secret Keys for interaction with the "health insurance agency service provider" SK.VSD	Two symmetric keys for MAC-Calculation and encryption purposes during interaction with the "health insurance agency service provider (VSDD)"
Secret Keys for interaction with the "download service provider" SK.CMS	Two symmetric keys for MAC-Calculation and encryption purposes during interaction with the "download service provider" (also called card management system CMS)
Secret Keys for interaction with the "combined services provider" SK.VSDCMS	Two symmetric keys for MAC-Calculation and encryption purposes during interaction with the "combined services provider"
Permission data EF.Einwilligung	These data contain information about permissions given by the card holder to use specific applications in the card "freiwillige Anwendungen"
reference data (voluntary application) EF.Verweis	Data of a so called "freiwillige Anwendung" (these are application which may only be used if a patient has allowed this explicitly before the first use).
Emergency data EF.eNotfalldaten EFStatusNotfalldaten	Emergency data ("Notfalldaten") are a specific part of "medical data (voluntary application)".
Permission information EF.Verweise_Gesund-heitsdatendienste	References to signed permissions given by the insured person.
Evidence data EF.Prüfungsnachweis	Evidence data („Prüfungsnachweis") generated in the framework of an online-check.
Personal declaration EF.PersönlicheErklärun-gen, EF.StatusPersönliche-Erklärungen	Personal declarations given by the insured person and the status of these data.
User's charge EF.Zuzahlungscontainer, EF.StatusZuzahlungen, EF.Zuzahlungstickets	Vouchers and related validation data records of the insured person inclusive their status.
Test status EF.TTN	Information about the participation of the insured person in a test phase.

Table 7 – Data Objects list

3.1.2 Subjects

Card holder	The card holder of the TOE is the legitimate user of the card, who is authenticated by use of the PIN.CH or the PIN.home Note: The following terms are related to the card holder: The patient is the person who uses the eHC in order to receive e. g. treatment by a doctor. Normally the patient is identical to the card holder. However, the patient may be incapable of using the card himself (e. g. children) and the card holder may be a different person acting on behalf of the patient. The insured person ("Versicherter") is the person, who has the insurance relation to the health insurance company. Usually this person is again identical to the card holder, however the latter may be for example a child of the former. However, since the TOE cannot distinguish these roles, only the card holder is defined as a subject.
Health Professional	Person acting as health professionals providing medical care to a patient (e.g. physician, dentist, pharmacist, psychotherapist, but also other health professionals yet to be formally defined, like midwives). These health professionals hold a HPC with a Card Verifiable Certificate of the Card Authentication Key with Role ID 'CHA.2', 'CHA.3', 'CHA.4', 'CHA.5' or 'CHA.7'.
Medical Assistant	Persons supporting a Health Professional. These health employees usually hold a HPC with a Card Verifiable Certificate of the Card Authentication Key with Role ID corresponding to that of the health professional whom they support ie 'CHA.2', 'CHA.3', 'CHA.4','CHA.5' or 'CHA.7'. The additional Role IDs 'CHA.6', 'CHA.8' ,'CHA.9'and 'CHA.10' are defined for specific purposes
Security Module Card (health care) (SMC)	This security module card is used in a health care environment in order to allow interaction with the eHC in situations, where employees without a personal card provide services. The SMC has a Card Verifiable Certificate of the Card Authentication Key with Role ID usually corresponding to that of the Health Professional, who is responsible for its operation I.e. 'CHA.2', 'CHA.3', 'CHA.4', 'CHA.5' or 'CHA.7'.However, a special type of SMC for hospitals may exist, which has Role ID CHA.2, but can be activated by HPCs with other Role IDs. The additional Role IDs 'CHA.6' , 'CHA.8', 'CHA.9' and 'CHA.10' are defined for specific purposes
Self Service Terminal	A self service terminal allows a card holder of an eHC to perform certain services. The self service terminal has an SMC with a Card Verifiable Certificate of the Card Authentication Key with Role ID 'CHA.1', which is distinct from the Role Ids of the preceding subjects.
Health insurance agency service provider	The "health insurance agency service provider" interacts with the TOE on behalf of the health insurance agency (VSD). The service provider uses a security module (e. g. in form of a SMC), which is authenticated by use of the key SK.VSD. .

TOE manufacturer (2) **(2) The TOE manufacturer is named Card manufacturer in the ST**	Person(s) responsible for development and production of the TOE. Note: According to the life cycle description the initialisation of the card is either done by the TOE manufacturer or by the personalisation service provider.
Personalisation service provider	person(s) responsible for personalisation of the card Methods to authenticate this role may be TOE specific and have to be defined in the Security target of a TOE. Note: This role is only responsible for the personalisation in phase 7 of the TOE's life cycle and has no access rights in phase 8.
Download service provider	Person(s) responsible for Downloading additional applications (consisting of file structures, their access rights and data) into the card in phase 8 of the TOE's lifecycle. (Card management system CMS) The service provider uses a security module (e. g. in form of a SMC), which is authenticated by use of the key SK.CMS. Note: There may be other more specific roles to produce data for the TOE like certificate service providers. However, since the card cannot distinguish such more specific roles technically according to an authentication mechanism in the card, such roles will not be defined as subjects.
combined services provider	name for the combination of the health insurance agency service provider and the download service provider (in case a decision is made to combine these services or at least to allow the use of a shared key for these services)
Other person	All persons who interact with the TOE without being authorised (as one of the preceding roles).

Table 8 – Subjects list

3.2 THREATS

The threats are those defined by the eHC PP.

T.Compromise_Internal_Data	**Compromise of confidential User or TSF data :** An attacker with high attack potential tries to compromise confidential user data or TSF data through the communication interface of the TOE by sending commands or by listening to the communication between a terminal and the TOE. This threat comprises several attack scenarios e.g. guessing of the user authentication data (PIN) or reconstruction of the private decipher key using the response code for chosen cipher texts (like Bleichenbacher attack for the SSL protocol implementation).
T.Forge_Internal_Data	**Forge of User or TSF data :** An attacker with high attack potential try to forge internal user data or TSF data This threat comprises several attack scenarios of smart card forgery. The attacker may try to alter the user data e.g. to add keys for decipherment of documents. The attacker may misuse the TSF management functions to change the user authentication data to a known value.
T.Misuse	**Misuse of TOE functions** : An attacker with high attack potential tries to use the TOE functions to gain access to the assets without knowledge of user authentication data or any implicit authorization This threat comprises several attack scenarios e.g. the attacker may try to circumvent the user authentication to use the DECIPHER command for document keys without authorization. The attacker may try alter the TSF data e.g. to extend the user rights after successful card-to-card authentication.
T.Intercept	**Interception of Communication** An attacker with high attack potential try to intercept the communication between the TOE and an SMC, HPC, Download service provider or Health insurance agency service provider in order to read, to forge, to delete or to add other data to the transmitted data classified as assets This threat comprises several attack scenarios. A health professional reads from and writes onto eHC patient's data like medication or medical data, which an attacker may read or forge during transmission. Attacker may try to read the document keys output by the TOE as DECIPHER command response. Attackers may try to manipulate card management processes.
T.Phys_Tamper	Physical Tampering An attacker with high attack potential may perform physical probing of the IC in order : • to disclose User Data, • to disclose/reconstruct the IC Embedded Software or • to disclose TSF data. An attacker may physically modify the IC in order to :

	• modify security features or functions of the IC, • modify security functions of the IC Embedded Software, • to modify User Data or • to modify TSF data. The physical tampering may be focused directly on the discloser or manipulation of TOE User Data (e.g. the document decipherment key) or TSF Data (e.g. authentication key of the smart card) or indirectly by preparation of the TOE to following attack methods by modification of security features (e.g. to enable information leakage through power analysis). Physical tampering requires direct interaction with the IC internals. Techniques commonly employed in IC failure analysis and IC reverse engineering efforts may be used. Before that hardware security mechanisms and layout characteristics need to be identified. Determination of software design including treatment of User Data and TSF Data may also be a pre-requisite. The modification may result in the deactivation of a security function. Changes of circuitry or data can be permanent or temporary.
T.Information_Leakage	**Information Leakage from TOE's chip** An attacker with high attack potential may exploit information which is leaked from the TOE during its usage in order to disclose confidential data (User Data or TSF data). The information leakage may be inherent in the normal operation or caused by the attacker. Leakage may occur through emanations, variations in power consumption, I/O characteristics, clock frequency, or by changes in processing time requirements. This leakage may be interpreted as a covert channel transmission but is more closely related to measurement of operating parameters, which may be derived either from measurements of the contact less interface (emanation) or direct measurements (by contact to the chip still available even for a contact less chip) and can then be related to the specific operation being performed. No direct contact with the IC internals is required here. Examples are the Differential Electromagnetic Analysis (DEMA) and the Differential Power Analysis (DPA)
T.Malfunction	**Malfunction due to Environmental Stress** An attacker with high attack potential may cause a malfunction of TSF or of the IC Embedded Software by applying environmental stress in order to : • deactivate or modify security features or functions of the TOE or • circumvent or deactivate or modify security functions of the IC Embedded Software. This may be achieved e.g. by operating the IC outside the normal operating conditions, exploiting errors in the IC Embedded Software or misuse of administration function. To exploit this an attacker needs information about the functional operation.
T.Abuse_Func	**Abuse of Functionality** An attacker with high attack potential may use functions of the TOE which shall not be used in TOE operational phase in order to :

	• disclose or manipulate User Data, • to manipulate (explore, bypass, deactivate or change) security features or functions of the TOE or • to disclose or manipulate TSF Data. This threat address attacks using the IC as production material for the smart card and using function for personalization in the operational state after delivery of the smart card.

Table 9 – Threats list

3.2.1 Assets coverage

The following table shows how the threats are appropriated to complete assets.

Threats / Assets	Card Authentication Private Key	Card Verifiable Authentication Certificate	Client-Server Authentication Private Key	Decipher Private Key	Display message	Emergency data	Initialisation data	Logging data	Permission data	Personal and health insurance data (open)	Personal and health insurance data (protected)	Personalisation data	Public Key for CV Certification Verification	Secret Keys for interaction with the "download service provider"	Secret Keys for interaction with the "health insurance agency service provider"	RAD (eHC)	VAD (eHC)	X.509 certificates	Permission information	Secret key for interaction with the "combined services provider"	Reference data	Evidence data	Personal declaration	User's charge	Test status
T.Abuse_Func							X					X													
T.Compromise_Internal_Data					X	X	X	X	X	X	X	X				X			X		X	X	X	X	X
T.Forge_Internal_Data					X	X	X	X	X	X	X	X				X			X		X	X	X	X	X
T.Information_Leakage	X		X	X	X								X	X	X	X				X					
T.Intercept					X	X	X	X	X	X	X	X							X		X	X	X	X	X
T.Malfunction																X	X								
T.Misuse					X	X	X	X	X	X	X	X					X		X		X	X	X	X	X
T.Phys_Tamper	X		X	X	X								X	X	X	X	X		X	X					

Table 10 – Threats / Assets correspondence analysis

3.3 ORGANISATIONAL SECURITY POLICIES

OSP.eHC_Spec	**Compliance to eHC specifications** The eHC shall be implemented according to the security relevant requirements of the specifications : [EHC spec part 1] [EHC spec part 2]
OSP.Additional_Applications	**Protection of additional Applications** ⇒ The TOE shall provide the possibility to authorised parties to load data for additional applications to the card. Loading of additional executable code shall not be possible ⇒ The TOE shall separate existing applications from additional applications. This means that data structures, access rights and data contents of such additional applications can not modify the security properties, in particular access control, for the existing applications. ⇒ By definig access rights to the files belonging to additional applications suitably it shall be possible to provide access control to such files using the mutual authentication services or the PIN authentication services. This OSP is designed to provide the functionality to add such applications in a secure way and to provide support for their future security needs.
OSP.Legal_Decisions	**Legal responsibility of authorised persons** The decision, which data are legally feasible for storage on the eHC has to be made by the persons authorised to deal with the data. The same holds for the decision, when data need to be deleted.
OSP.services	**Services provided by the card** The eHC shall provide the following services: • Service_Asym_Mut_Auth_w/o_SM • Service_Asym_Mut_Auth_with_SM • Service_Sym_Mut_Auth_with_SM • Service_User_Auth_PIN_ and Servive_User_Auth_PUC • Service_Privacy • Service_Client_Server_Auth • Service_Data_Decryption • Service_Card_Management and • Service_Logging Note: The eHC also provides electronic signature services
OSP.logging	**Logging of access to medical data** All access to medical data (except reading access by the Card holder himself) must be logged. Access to the log file must be protected.

Table 11 – OSPs list

Service_Asym_Mut_Auth_w/o_SM (5): Mutual Authentication using asymmetric techniques between the eHC and a Health Professional Card (HPC) or a Security Module Card (SMC) without establishment of a Secure Channel .

This service is meant for situations, where the eHC requires authentication by a HPC or SMC, but where the following data exchange is done without help of a security module.

(5) The Abbreviation SM here stands for Secure Messaging, which is the card security protocol realising a secure channel.

Service_Asym_Mut_Auth_with_SM: Mutual Authentication using asymmetric techniques between the eHC and a Security Module Card (SMC) or another security module with establishment of a Secure Channel.

This service is meant for situations, where the eHC requires authentication by a SMC or another security module, which provides similar functionality, and where the following data exchange is done with the help of this security module and can therefore be encrypted and/or secured by a MAC.

Service_Sym_Mut_Auth_with_SM: Mutual Authentication using symmetric techniques between the eHC and a security module with establishment of a Secure Channel .

This service is meant for situations, where the eHC communicates with a central security module, which shares symmetric keys with the card. This may be a security module of the health insurance organisation, when managing the patient contractual data, or a module of the Download service provider, which may add new applications to the eHC (or manage the existing ones).

Service_User_Auth_PIN: The card holder authenticates himself with one of his PINs, either PIN.CH or PIN.home.

This service is meant as a support service for some of the other services, which may require user authentication. In addition it provides privacy protection because certain data in the card (or secured by the card) can only be accessed after user authentication. In particular this applies to sensitive medical data.
Functions to change the PIN or to unblock the PIN, when it was blocked (because of successive false PIN entries) are supporting this service. For the latter the PIN unblocking code (PUC) is used, this authentication will be called **Service_User_Auth_PUC**.

Service_Privacy: The card holder may deactivate sensitive medical data in the eHC. In order to use this service he authenticates himself with a PIN..

This service allows the card holder to prevent health care providers from accessing data, which the card holder doesn't want them to know. Note, that that the name Service_Privacy doesn't mean that this is the only privacy related service. In fact all other services also support privacy.

Service_Client_Server_Auth: The eHC implements a PKI application, which in particular allows to use the TOE as an authentication token for an authentication of a client to a server (by means of an asymmetric method using X.509 certificates). The eHC contains two different keys

and corresponding certificates for this service. In order to use this service the card holder authenticates himself with a PIN.. One of the keys can also be used without authentication by the card holder but requires authentication by a HPC or SMC in this case.

This service may for example be useful if the card holder wants to access a server provided by the health insurance organisation, where confidential data of the card holder are managed. So it can also be seen as an additional privacy feature.

Note, that a potential authentication of the server to the client is not supported by the eHC.

Service_Data_Decryption: The eHC implements a PKI application, which in particular allows using the TOE as a data decryption token. Symmetric document encipherment keys, which are themselves encrypted with the cards public key can only be decrypted with the help of the card. There are two sets of asymmetric key pairs in the eHC to allow following two possibilities of authentication for this service:

- In order to use this service the cardholder authenticates himself with a PIN. One of the key can also be used without authentication by the cardholder, but requires authentication by a HPC or SMC in this case.

This service is meant for situations, where confidential data are stored on a server, but shall only be accessible with the cardholder's permission or with the authentication of a health professional. So it can also be seen as a privacy feature.

Service_Card_Management: The eHC allows creation of new applications and management of existing applications to the card management system. This is secured by the service Service_Sym_Mut_Auth_with_SM.

Service_Logging: The eHC provides a file, which allows to store information about the fifty last accesses to medical data in the card. The card itself doesn't control the content of these data, it is up to the authorised persons, who have write access to these data, to write them correctly.

3.4 ASSUMPTIONS

A.Users	**Adequate usage of TOE and IT-Systems in the environment**. The card holder of the TOE uses the TOE adequately. In particular he doesn't tell the PIN (or PINs) of the eHC to others and doesn't hand the card to unauthorised persons. Other actors use their data systems according to the overall system security requirements. The Card holder of the eHC needs to be informed clearly about secure usage of the product.
A.Perso	**Secure handling of data during personalisation and additional personalisation** All data structures and data on the card produced during personalisation or additional personalisation steps during the end-usage phase are correct according to the specifications and are handled correctly regarding integrity and confidentiality of these data. This includes in particular sufficient cryptographic quality of cryptographic keys (in accordance with the cryptographic algorithms specified for the eHC) and their confidential handling.The personalisation service provider controls all materials equipment and information, which he uses to personalize authentic smartcards, in order to prevent counterfeit of the TOE. The same requirements hold for all activities belonging to Initialisation phase, if they are executed after TOE delivery. This holds for example if the personalisation service provider also sends the initialisation data to the TOE or if the TOE delivered by the TOE manufacturer in form of smart card modules, which are the inserted into the plastic cards at a later stage. .

Table 12 – Assumptions list

4. SECURITY OBJECTIVES

4.1 GENERAL

This section identifies and defines the security objectives for the TOE and its environment. Security objectives reflect the stated intent and counter the identified threats, as well as comply with the identified organizational security policies and assumptions.

4.2 SECURITY OBJECTIVES FOR THE TOE

OT.Access_rights	**Access control policy for data in the TOE** In the End Usage Phase the TOE shall implement the access control policy SFP_access_rules (define in following chapter) **Implementation of the security policies OSP.eHC_Spec, OSP.Logging** **Coverage of the threats T.Compromise_Internal_Data, T.Forge_Internal_Data, T.Misuse and T.Intercept**
OT.AC_Pers	**Access control for personalisation** The TOE must ensure that the Personalisation data can be written by an authorized personalisation service provider. **Implementation of the security policy OSP.eHC_Spec** **Coverage of the threats T.Compromise_Internal_Data, T.Forge_Internal_Data, T.Misuse and T.Intercept**
OT.Additional_Applications	**Protection of additional Applications** The TOE shall provide the possibility to authorised parties to load data for additional applications to the card. Loading of additional executable code shall not be possible. The TOE shall separate existing applications from additional applications. This means that data structures, access rights and data contents of such additional applications can not modify the security properties, in particular access control, for the existing applications. By defining access rights to the files belonging to additional applications suitably it shall be possible to provide access control to such files using the mutual authentication services or the PIN authentication services. **Implementation of the security policies OSP.eHC_Spec, OSP.Additional_Applications**
OT.Services	**Services provided by the Card** The eHC shall provide the following services: • Service_Asym_Mut_Auth_w/o_SM • Service_Asym_Mut_Auth_with_SM • Service_Sym_Mut_Auth_with_SM • Service_User_Auth_PIN and Service_User_Auth_PUC

	• Service_Privacy • Service_Client_Server_Auth • Service_Data_Decryption • Service_Card_Management and • Service_Logging **Implementation of the security policies OSP.eHC_Spec, OSP.Services, OSP.Logging** **Coverage of the threats T.Compromise_Internal_Data, T.Forge_Internal_Data, T.Misuse and T.Intercept**

OT.Cryptography	**Implementation of cryptographic algorithms**
	The cryptographic algorithms required by the eHC specifications, are implemented according to their definition.
	These algorithms are:
	• RSA
	  ○ PKCS #1 V1.5
	  ○ ISO 9796-2 (modes DS1 and DS2)
	  ○ RSA OAEP
	• SHA-256
	• 3TDES.
	Implementation of the security policy OSP.eHC_Spec
	
	Coverage of the threats T.Compromise_Internal_Data, T.Forge_Internal_Data, T.Misuse and T.Intercept

OT.Prot_Inf_Leak	**Protection against Information Leakage** The TOE must provide protection against disclosure of confidential data (User Data or TSF data) stored and/or processed in the TOE's chip • by measurement and analysis of the shape and amplitude of signals or the time between events found • by measuring signals on the electromagnetic field, power consumption, clock, or I/O lines and • by forcing a malfunction of the TOE and/or • by a physical manipulation of the TOE **Coverage of the threat T.Information_Leakage**
OT.Prot_Phys_Tamper	**Protection against Physical Tampering** The TOE must provide protection the confidentiality and integrity of the User Data, the TSF Data, and the chip Embedded Software. This includes protection against attacks with high attack potential by means of • measuring through galvanic contacts which is direct physical probing on the chips surface except on pads being bonded (using standard tools for measuring voltage and current) or • measuring not using galvanic contacts but other types of physical interaction between charges (using tools used in solid-state physics research and IC failure analysis) • manipulation of the hardware and its security features, as well as

	• controlled manipulation of memory contents (User Data, TSF Data). with a prior • reverse-engineering to understand the design and its properties and functions. **Coverage of the threat T.Phys-Tamper**
OT.Prot_Malfunction	**Protection against Malfunctions** The TOE must ensure its correct operation. The TOE must prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent errors. The environmental conditions may include external energy (esp. electromagnetic) fields, voltage (on any contacts), clock frequency, or temperature. **Coverage of the threat T.Malfunction**
OT.Prot_Abuse_Func	**Protection against Abuse of Functionality** The TOE must prevent that functions of the TOE which may not be used after TOE Delivery can be abused in order • to disclose critical User Data, • to manipulate critical User Data of the Smartcard Embedded Software, • to manipulate Soft-coded Smartcard Embedded Software or • bypass, deactivate, change or explore security features or functions of the TOE. Details depend, for instance, on the capabilities of the Test Features provided by the IC Dedicated Test Software which are not specified here. **Coverage of the threat T.Abuse_Func**

Table 13 – TOE's objectives list

4.2.1 SFP access Rules for Electronic Health Application

The following subjects may interact with the TOE :
Card holder, Medical Assistant, Health professional, Security Module Card (health care), Self Service Terminal, Health insurance agency service provider, TOE manufacturer, Personalisation service provider, Download service provider, combined services provider, other person.

The following objects are covered by the policy :
Personal and health insurance data (open), Personal and health insurance data (protected), VAD (eHC), RAD (eHC), Logging data, Card Authentication Private Key, Card Verifiable Authentication Certificate, Client-Server Authentication Private Keys, Decipher Private Keys, Display message, X.509 certificates, Public Key for CV Certification Verification, SK.VSD, SK.CMS,SK.VSDCMS, permission data, reference data (voluntary application), emergency data, permission information, evidence data,personal declaration, user's change, test status.

The following authentication methods are covered by the policy:
The services : Service_Asym_Mut_Auth_w/o_SM, Service_Asym_Mut_Auth_with_SM, Service_Sym_Mut_Auth_with_SM, Service_User_Auth_PIN and Service_User_Auth_PUC

The following security attributes for subjects are maintained by the TOE:
For every authentication method the TOE maintains the status of successful authentication (successful PIN verification, successful mutual authentication). (These are security attributes for the connected subject, because the TOE derives the access rights from these attributes).

The following access methods are maintained by the TOE:
Access is allowed only using the defined command interface of the TOE. In other words: A subject sends a command APDU as defined in the eHC specification to the TOE and the TOE processes it. Requirements for encryption or MAC-protection (Using Secure Messaging) will be included in addition for access to some of the data.

The following types of access are used in the rules below:
"Read", "write", "delete", "deactivate" (this means making data invisible for other subjects, but without deleting them), "activate" (making deactivated data visible again), "use" (a command is called, which uses data internally, this is relevant for cryptographic keys).
As specific variants of the write access the following terms are used: "Modify" means to change existing data. "Append" means to add data at the end of existing data. "Create" means to create new data structures

The following access rules are defined for the TOE's objects
For all files and other security relevant data (PINs, keys) the TOE maintains the following access rules as defined in the eHC specification, [eHC spec part 2].

4.3 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT

OE.Users	**Adequate usage of TOE and IT-Systems in the environment.** The Card holder of the TOE needs to use the TOE adequately. In particular he mustn't tell the PIN (or PINs) of the eHC to others and mustn't hand the card to unauthorised persons. **Implementation of the assumption A.Users**
OE.legal_decisions	**Legal responsibility of authorised persons** The decision, which data are legally feasible for storage on the eHC has to be made by the persons authorised to deal with the data. The same holds for the decision, when data need to be deleted. These persons must use their IT systems according to the legal requirements. This objective holds for all subjects (or the persons controlling them, if the subjects themselves are technical devices), except the Card holder (who's behaviour is covered by other objectives) and the category "Other person", which includes attackers. **Implementation of the security policies OSP.Legal_Decisions, OSP.Logging** **Coverage of the threats T.Compromise_Internal_Data, T.Forge_Internal_Data, T.Misuse and T.Intercept**
OE.data_protection	**Protection of sensitive data outside of the eHC** The persons responsible for the handling of sensitive data outside of the eHC (this includes medical data, PINs, cryptographic keys and sensitive personal data) use adequate protection for confidentiality and integrity of these data. **Coverage of the threats T.Compromise_Internal_Data, T.Forge_Internal_Data, T.Misuse and T.Intercept**
OE.Perso	**Secure handling of data during personalisation and additional personalisation** All data structures and data on the card produced during personalisation or additional personalisation steps during the end-usage phase must be correct according to the specifications and must be handled correctly regarding integrity and confidentiality of these data. This includes in particular sufficient cryptographic quality of cryptographic keys (in accordance with the cryptographic algorithms specified for the eHC) and their confidential handling. The personalisation service provider must control all materials, equipment and information needed to personalize authentic smart cards in order to prevent counterfeit of the TOE. The same requirements hold for all activities belonging to Phase 6

	"Initialisation", if they are executed after TOE delivery. This holds for example if the personalisation service provider also sends the initialisation data to the TOE or if the TOE delivered by the TOE manufacturer in form of smart card modules, which are then inserted into the plastic cards at a later stage. **Implementation of the security policy OSP.Additional_Applications** **Implementation of the assumption A.Perso**

Table 14 – Environment's objectives list for the Electronic Health Application

4.4 SECURITY OBJECTIVES RATIONALE

4.4.1 Security Objectives Coverage

	OT.AC_Pers	OT.Access_Rights	OT.Additional_Applications	OT.Cryptography	OT.Services	OT.Prot_Abuse-Func	OT.Prot_Inf_Leak	OT.Prot_Phys-Tamper	OT.Prot_Malfunction	OE.Data_Protection	OE.Legal_Decisions	OE.Perso	OE.Users
A.Perso												X	
A.Users													X
OSP.Additional_Applications			X									X	
OSP.eHC_Spec	X	X	X	X	X							X	
OSP.Legal_Decisions											X		
OSP.Logging		X			X						X		
OSP.Services					X								
T.Abuse_Func						X							
T.Compromise_Internal_Data	X	X		X	X					X	X		
T.Forge_Internal_Data	X	X		X	X					X	X		
T.Information_Leakage							X						
T.Intercept	X	X		X	X					X	X		
T.Malfunction									X				
T.Misuse	X	X		X	X					X	X		
T.Phys Tamper								X					

Table 15 – Security objectives / Threats-Assumptions-Policies correspondence analysis

4.4.2 Security Objectives Sufficiency

The following text describes for every OSP, Threat and Assumption, how they are covered by Security Objectives.

The organizational security policy **OSP.eHC_Spec** "Compliance to eHC specifications" is implemented by the following TOE security objectives:

- OT.Services requires that the TOE provides the security services, which are realised by the commands defined in the specification.
- OT.Cryptography requires that the cryptographic algorithms as defined in the specification are implemented.
- OT.Access_Rights requires that the access rights are defined according to the policy SFP_access_rules. These rules are chosen according to the access rights defined in the [eHC spec], part 2, annex B.
- OT.Additional_Applications requires rules for the loading of additional applications, which is also compatible to the definitions in the specifications.
- The objectives for the TOE environment OE.Perso "Secure personalisation" (together with OT.AC_Pers "Access control for personalisation" protecting the personalisation functions of the TOE) ensure that the Personalisation service provider will provide a genuine TOE initialized and personalized according to the specification to the Card holder.

OSP.Additional_Applications is fully covered by OT.Additional_Applications, which is essentially identical to OSP.Additional_Applications. In addition it is supported by OE.Perso because this security objective requires adequate organisational security, when loading additional applications during the operational phase.

OSP.Legal_Decisions is fully covered by OE.Legal_Decisions, which is essentially identical to OSP.Legal_Decisions.

OSP.Services is fully covered by OT.Services, which is essentially identical to OSP.Services.

OSP.Logging is realised in cooperation between the TOE and its operational environment:

- According to OT.Services the TOE provides the service "Service_Logging". This service authorized users to write logging data into the card.
- According to OE.Legal_Decision39all authorized users are responsible for the correctness of the logging data, they write into the card. This compensates for the fact that the card cannot control the content of this file.
- According to OT.Access_Rights, access to the log file is protected.

The threats **T.Compromise_Internal_Data, T.Forge_Internal_Data, T.Misuse and T.Intercept** are all countered by the following combination of objectives:

- OT.Access_Rights (supported by OT.Services, OT.Cryptography) implies that data in the TOE can only be read, written or modified according to the access rules as defined in the access control

policy SFP_access_rules, which was defined in OT.Access_Rights. The support by OT.Services is needed since several rules of SFP_access_rules restrict the access to certain subjects (card holder, health professional, etc.) the authenticity of which is made sure by services required by OT.Services (e.g. Service_User_Auth_PIN, Service_Sym_Mut_Auth_with_SM, Service_Asym_Mut_Auth_with_SM,). The support by OT.Cryptography is needed since several services required by OT.Services rely on cryptographic mechanisms required by OT.Cryptography (e.g a symmetric encryption algorithm is needed for Service_Sym_Mut_Auth_with_SM, an asymmetric algorithm for Service_Asym_Mut_Auth_with_SM).

- OT.AC_Pers protects the personalisation functions of the TOE against unauthorised use.

- OE.Legal_Decisions and OE.Data_Protection imply that authorised persons, who are allowed to read, write or modify data in the card, use these rights only in an environment, where unauthorised access to these data is prevented by the environment.

  An example for this is as follows: The service Service_Asym_Mut_Auth_w/o_SM allows health professionals to access Electronic prescriptions in the card. This is allowed only in a closed environment, where attackers cannot access the data transmitted between eHC and the health professionals IT equipment. For the case of transmission over insecure lines the service Service_Asym_Mut_Auth_with_SM is provided and the objectives for the environment imply that health professionals use these services adequately.

The threat **T.Phys-Tamper** “Physical Tampering” is adverted directly by the security objective OT.Prot_Phys-Tamper “Protection against physical tampering”.

The threat **T.Information_Leakage** “Information Leakage from smart card chip” is adverted directly by the security objective OT.Prot_Inf_Leak “Protection against information leakage” addressing the protection against disclosure of confidential data (User Data or TSF data) stored and/or processed in the TOE by attacks including but not limited to use of side channels, fault injection or physical manipulation.

The threat **T.Malfunction** “Malfunction due to Environmental Stress” is adverted directly by the security objective OT.Prot_Malfunction “Protection against Malfunctions”.

The threat **T.Abuse_Func** “Abuse of Functionality” is adverted directly by the security objective OT.Prot_Abuse-Func “Protection against abuse of functionality” preventing the use of TOE functions which are intended for the testing, the initialisation and the personalisation of the TOE and which must not be accessible after TOE delivery.

The security objective for the environment **OE.Users** “Adequate usage of TOE and IT-Systems” implements directly the assumption **A.Users** “Adequate usage of TOE and IT-Systems”.

The security objective for the environment OE.Perso “Secure personalisation” implements the assumption **A.Perso** “Personalisation of the Smart Card”.

5. EXTENDED COMPONENT DEFINITION

5.1 FCS_RND GENERATION OF RANDOM NUMBERS

To define the IT security functional requirements of the TOE an additional family (FCS_RND) of the Class FCS (cryptographic support) is defined here. This family describes the functional requirements for random number generation used for cryptographic purposes.

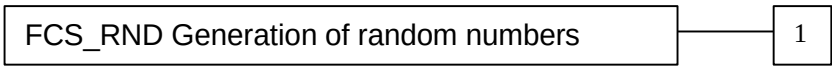
The family "Generation of random numbers (FCS_RND)" is specified as follows.

FCS_RND Generation of random numbers

Family behaviour

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

Component levelling:



FCS_RND.1	Generation of random numbers requires that random numbers meet a defined quality metric.
Management:	FCS_RND.1 There are no management activities foreseen.
Audit:	FCS_RND.1 There are no actions defined to be auditable.
FCS_RND.1	Quality metric for random numbers
Hierarchical to:	No other components.
FCS_RND.1.1	The TSF shall provide a mechanism to generate random numbers that meet [assignment: *a defined quality metric*].
Dependencies:	No dependencies.

5.2 FMT_LIM LIMITED CAPABILITIES AND AVAILABILITY

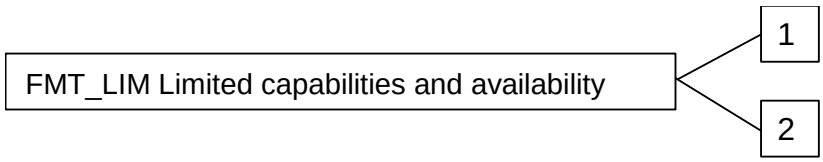
The family "Limited capabilities and availability (FMT_LIM)" is specified as follows.

FMT_LIM Limited capabilities and availability

Family behaviour

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

Component levelling:



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

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

Management: FMT_LIM.1, FMT_LIM.2

There are no management activities foreseen.

Audit: FMT_LIM.1, FMT_LIM.2

There are no actions defined to be auditable.

To define the IT security functional requirements of the TOE an additional family (FMT_LIM) of the Class FMT (Security Management) is defined here. This family describes the functional requirements for the Test Features of the TOE. The new functional requirements were defined in the class FMT because this class addresses the management of functions of the TSF. The examples of the technical mechanism used in the TOE show that no other class is appropriate to address the specific issues of preventing the abuse of functions by limiting the capabilities of the functions and by limiting their availability.

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

FMT_LIM.1 Limited capabilities

Hierarchical to: No other components.

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 [assignment: *Limited capability and availability policy*].

Dependencies: FMT_LIM.2 Limited availability.

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

FMT_LIM.2 Limited availability

Hierarchical to: No other components.

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*].

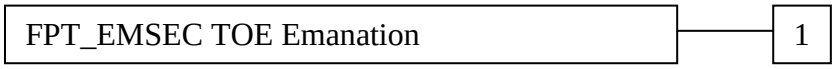
Dependencies: FMT_LIM.1 Limited capabilities.

5.3 FPT_EMSEC TOE EMANATION

The family "TOE Emanation (FPT_EMSEC)" is specified as follows.

Family behaviour

This family defines requirements to mitigate intelligible emanations.



Component levelling:

FPT_EMSEC.1 TOE emanation has two constituents:

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

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

Management: FPT_EMSEC.1

There are no management activities foreseen.

Audit: FPT_EMSEC.1

There are no actions defined to be auditable.

FPT_EMSEC.1 TOE Emanation

Hierarchical to: No other components.

FPT_EMSEC.1.1	The TOE shall not emit [assignment: *types of emissions*] in excess of [assignment: *specified limits*] enabling access to [assignment: *list of types of TSF data*] and [assignment: *list of types of user data*].
FPT_EMSEC.1.2	The TSF shall ensure [assignment: *type of users*] are unable to use the following interface [assignment: *type of connection*] to gain access to [assignment: *list of types of TSF data*] and [assignment: *list of types of user data*].

Dependencies: No other components.

6. SECURITY REQUIREMENTS

6.1 GENERAL

This chapter gives the security functional requirements and the security assurance requirements for the TOE.
Section 5 describes the extended component FPT_EMSEC.1, FMT_LIM.1, FMT_LIM.2 and FCS_RND.1. Section 6.2 provides the security functional requirements. Operations for assignment, selection and refinement have been made.
The TOE security assurance requirements statement is given in section 6.3.

6.2 TOE SECURITY FUNCTIONAL REQUIREMENTS

6.2.1 TOE security functional requirements list

The CC allows several operations. Each of these operations is used in this document :

- The *refinement* operation is used to add detail to a requirement. Refinement of security requirements is denoted by the word **refinement**.
- The *assignment* operation is used to assign a specific value . Assignment is denoted by using **bold**.
- The *iteration* operation is used when a component is repeated with varying operations. Iteration is denoted by showing a slash **"/",** and the iteration indicator after the component identifier.
- The *selection* operation is used to select one or more options. Selections are denoted as **underlined bold text.**

Identification	Description
FCS	**Cryptographic support**
FCS_CKM.1	**Cryptographic key generation**
FCS_CKM.4	**Cryptographic key destruction**
FCS_COP.1	**Cryptographic operation**
FCS_RND.1(9)	**Random Number Generation**
FDP	**User data protection**
FDP_ACC.2	**Complete Access Control**
FDP_ACF.1	**Security attribute based access control**
FDP_RIP.1	**Subset residual information protection**
FDP_SDI.2	**Stored Data integrity**
FDP_UCT.1	**Basic data exchange confidentiality**
FDP_UIT.1	**Data exchange integrity**
FIA	**Identification and authentication**
FIA_AFL.1	**Authentication failure handling**
FIA_ATD.1	**User attribute definition**
FIA_UAU.1	**Timing of authentication**
FIA_UAU.4	**Single-use authentication mechanisms**
FIA_UID.1	**Timing of identification**
FMT	**Security management**
FMT_LIM.1(9)	**Limited capabilities**
FMT_LIM.2(9)	**Limited availability**
FMT_MTD.1	**Management of TSF data**
FMT_SMF.1	**Specification of Management Functions**
FMT_SMR.1	**Security roles**
FPT	**Protection of the TSF**
FPT_EMSEC.1 (9)	**TOE Emanation**
FPT_FLS.1	**Failure with preservation of secure state**
FPT_PHP.3	**Resistance to physical attack**
FPT_TST.1	**TSF testing**
FTP	**Trusted path/channels**
FTP_ITC.1	**Import of user data without security attributes**

Table 16 – TOE security functional requirements list

(9) This requirement is an extension to [CCPART2].

6.2.1.1 FCS – Cryptographic support

6.2.1.1.1 FCS_CKM.1

FCS_CKM.1.1 /SM	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **card-to-card authentication with secure messaging** and specified cryptographic key sizes **168 bit** that meet the following : **[EHC spec part 1 §7.2]** **Application note :** The Key Generation is done during a mutual authentication with trusted channel establishment.The Authentication Protocol produces agreed parameters to generate the encryption key and the message authentication keys for secure messaging. The algorithm uses random numbers generated by the TSF as required by FCS_RND.1.

6.2.1.1.2 FCS_CKM.4

FCS_CKM.4.1	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method : • **Volatile keys are destroyed by overwriting RAM area with 00** • **Permanently stored keys (in EEPROM) are overwritten by their new values if updated** that meets the following: **None** Application note : The TOE shall destroy the encryption session key and the message authentication session keys for secure messaging after reset or termination of secure messaging session or reaching fail secure state according to FPT_FLS.1.

6.2.1.1.3 FCS_COP.1

FCS_COP.1.1/ HASH	The TSF shall perform **hashing** in accordance with a specified cryptographic algorithm **SHA 256** and cryptographic key sizes **none** that meet the following**: [EHC spec part 1 §7.1]**
FCS_COP.1.1/ CCA_SIGN	The TSF shall perform **digital signature-creation** in accordance with a specified cryptographic algorithm **RSA ISO 9796-2 (DS1)** and cryptographic key size of **2048 bits** that meet the following: **[EHC spec part 1 §7.6.3.1]**
FCS_COP.1.1/ CCA_VERIF	The TSF shall perform **digital signature-verification** in accordance with a specified cryptographic algorithm **RSA ISO 9796-2 (DS1)** and cryptographic key size of **2048 bits** that meet the following: **[EHC spec part 1 §7.6.4.1]**
FCS_COP.1.1/ CSA	The TSF shall perform **digital signature-creation** in accordance with a specified cryptographic algorithms **RSA ISO 9796-2 (DS2), RSA PKCS#1-v1_5, or RSA PKCS#1-PSS** and cryptographic key sizes **2048 bits** that meet the following: **[EHC spec part 1 §7.6.3.1]**
FCS_COP.1.1/ ASYM_DEC	The TSF shall perform **decryption** in accordance with a specified cryptographic algorithm **RSA PKCS#1 V1.5 and RSA OAEP** and cryptographic key **2048 bits length** that meet the following: **[EHC spec part 1 §7.8]**
FCS_COP.1.1/ SYM	The TSF shall perform **encryption and decryption** in accordance with a specified cryptographic algorithm **3TDES in CBC mode** and cryptographic key size of **168 bits** that meet the following: **[EHC spec part 1 §7.7]**
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 (CBC in authentication protocols and CFB in session MACs)**and cryptographic key size of **168 bits** that meet the following: **[EHC spec part 1 §7.6.1]**

6.2.1.1.4 FCS_RND.1

FCS_RND.1.1	The TSF shall provide a mechanism to generate random numbers that meet **TRNG of class P2 ([AIS31]) with strength of mechanism set to high.** Application note : This SFR requires the TOE to generate random numbers used for : * the authentication protocols as required by FIA_UAU.4, and * the key agreement FCS_CKM.1/SM for secure messaging. The quality metric shall be chosen to ensure the strength of function high.

6.2.1.2 FDP – User data protection

6.2.1.2.1 FDP_ACC.2

FDP_ACC.2.1	The TSF shall enforce the **SFP access Rules** on **all subjects and objects defined by SFP access Rules** 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 controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

6.2.1.2.2 FDP_ACF.1

FDP_ACF.1.1	The TSF shall enforce the **SFP access Rules** to objects based on the following: **all subjects and objects together with their respective security attributes as defined in SFP access Rules**
FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **rules for all access methods and the access rules defined in SFP access Rules.**
FDP_ACF.1.3	The TSF shall explicitly authorize 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 rule: **rules for all access methods and the access rules defined in SFP access Rules**

6.2.1.2.3 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: **PINs, secret and private cryptographic keys, data in all files, which are not freely accessible.**

6.2.1.2.4 FDP_SDI.2

The following data persistently stored by TOE have the user attribute "**integrity checked persistent stored data**":

1. ***All user data***
2. ***cryptographic keys (persistent ones)***
3. ***PINs (persistent),***
4. ***user data in files on the card (persistent),***
5. ***file management information (like access rules for files), and***

the card life cycle status (persistent),

The following volatile data used by TOE have the user attribute "**integrity checked volatile data**":

1. ***cryptographic keys (volatile keys as session keys and external public keys)***
2. ***security relevant status variables of the card (e. g. authentication status for the PIN or for mutual authenticate) (volatile)***

security states: always volatile in RAM, secured with checksum

FDP_SDI.2.1/Persistent	The TSF shall monitor user data stored in containers controlled by the TSF for **integrity errors** on all objects, based on the following attributes: **integrity checked persistent stored data :** ➢ **PIN (RAD),** ➢ **Crypto keys : Private RSA keys, symmetric authentication keys (SK.VSD/CMS), public key for certificate verification (CVC),** ➢ **User data that must be integrity checked according to [EHC spec part 2] (some can be updated with respect to access condition, some need not be integrity checked),** ➢ **File management access rules for files (keys and pins - cannot be updated),** ➢ **Card Life Cycle Status.** Note : that all those data reside in files, and therefore automatically have a checksum, keys and pin reference values additionally masked
FDP_SDI.2.2/Persistent	Upon detection of a data integrity error, the TSF shall: **1. Prohibit the use of the altered data** **2. Inform the connected entity about integrity error.**

FDP_SDI.2.1/Volatile	The TSF shall monitor user data stored in containers controlled by the TSF for **integrity errors** on all objects, based on the following attributes: **integrity checked volatile data :** ➢ **Crypto keys : session keys, public keys entered via certificate verification,** ➢ **Security states** ➢ **Input data for electronic signature.**
FDP_SDI.2.2/Volatile	Upon detection of a data integrity error, the TSF shall: **3. Prohibit the use of the altered data** **4. Inform the connected entity about integrity error.**

6.2.1.2.5 FDP_UCT.1

FDP_UCT.1.1	The TSF shall enforce the **SFP_access_rules** to be able to **transmit and receive** objects in a manner protected from unauthorized disclosure. **Application note**: The TOE supports secure messaging with symmetric encryption (cf. SFR FCS_COP.1/SYM) after card-to-card authentication with secure messaging

6.2.1.2.6 FDP_UIT.1

FDP_UIT.1.1	The TSF shall enforce the **SFP_access_rules** 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.

Application note: The TOE supports secure messaging with MAC (cf. FCS_COP.1/MAC) after card-to-card authentication with secure messaging.

6.2.1.3 FIA – Identification and Authentication

6.2.1.3.1 FIA_AFL.1

FIA_AFL.1.1/ PIN	The TSF shall detect when **3** unsuccessful authentication (PIN.CH and PIN.home) attempts occur related to **consecutive failed human user authentication for the health care application .**
FIA_AFL.1.2/ PIN	When the defined number of unsuccessful authentication attempts has been **met**, the TSF shall **block the PIN (PIN.CH and PIN.home) for authentication until successful unblock with resetting code.**
FIA_AFL.1.1/ PUC	The TSF shall detect when **10** ~~unsuccessful~~ [2] attempts occur related to **usage of the eHC-PIN unblocking code.**
FIA_AFL.1.2/ PUC	When the defined number of ~~unsuccessful~~[3] authentication attempts has been **met** , the TSF shall **block the PIN unblocking code.**

6.2.1.3.2 FIA_ATD.1

FIA_ATD.1.1	The TSF shall maintain the following list of security attributes belonging to individual users: **identity and role**. **Application note :** Applies to (i) the human user authentication, i.e. the card holder, whose identity is given in the Personal and health insurance data (open), and to (ii) the card-to-card authentication where the identity (i.e. the ICCSN.ICC) and the role (i.e. Role ID) are encoded in the CV certificate.

6.2.1.3.3 FIA_UID.1

FIA_UID.1.1	The TSF shall allow **(1) reading the ATR** **(2) reading the Card Verifiable Authentication Certificate,** **(3) reading the Certificate Service Provider Certificate** **(4) reading EF_GDO (containing ICCSN)** **(5) reading EF_DIR (listing all applications)** **(6) Selecting Applications (Select(AID)** **(7) Changing SE with Manage Security Environment (Restore)** 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.

[2] And [4] : Refinement : not only unsuccessful but all attempts shall be counted here – obviously this refinement is valid, because the original requirement is still fulfilled

6.2.1.3.4 FIA_UAU.1

FIA_UAU.1.1	The TSF shall allow : **(1) reading the ATR** **(2) reading the Card Verifiable Authentication Certificate,** **(3) reading the Certificate Service Provider self-signed Certificate,** **(4) Identification by providing the users eHC-PIN** **(5) identification by providing the users certificate** **(6) identification of "health insurance agency service provider (VSD)", "download service provider (CMS)", or "combined service provider" by selection of the corresponding key set SK.VSD, SK.CMS, or SK.VSDCMS.** 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.

6.2.1.3.5 FIA_UAU.4

FIA_UAU.4.1	The TSF shall prevent reuse of authentication data related to **Card-to-Card Authentication Mechanism** **Application note :** The Card-to-Card Authentication Mechanism is based on asymmetric cryptographic primitives as required by FCS_COP.1/CCA_SIGN and FCS_COP.1/CCA_VERIF or on symmetric cryptography using FCS_COP.1/SYM and uses the freshness generated by the TOE random data (see FCS_RND.1) as challenge to prevent reuse of a response generated in a successful authentication attempt.

6.2.1.4 FMT – Security Management

6.2.1.4.1 FMT_LIM.1

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.**

6.2.1.4.2 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**.

6.2.1.4.3 FMT_MTD.1

FMT_MTD.1.1/ ini	The TSF shall restrict the ability to **write** the **Initialisation data** to **the TOE manufacturer .**
FMT_MTD.1.1/ pers	The TSF shall restrict the ability to **write** the **Personalisation data** to **the Personalisation service provider** . Application note : the management of applications during the end usage phase is not a task for the "Personalisation Service Provider" but for the "Download Service Provider".
FMT_MTD.1.1/ CMS	The TSF shall restrict the ability to **write** the **1. File structures for additional Applications,** **2. Cryptographic Keys for additional applications** **3. PINs and other user authentication reference data for additional applications and** 4. **Access Rights for additional applications** to **the Download service provider.**
FMT_MTD.1.1/ PIN	The TSF shall restrict the ability to **modify and unblock** the **PIN** to **the Card Holder** . Application note : The cardholder modifies his or her PIN as special case of the User Authentication Reference Data by means of : * the command CHANGE REFERENCE DATA and providing the old and the new PIN or * the command RESET RETRY COUNTER and providing the PUC and the new PIN. He or she unblocks the PIN by means of :

	* the command RESET RETRY COUNTER and providing the PUC and the new PIN or * the command RESET RETRY COUNTER and providing the PUC (without a new PIN).
FMT_MTD.1.1/ KEY_MOD	The TSF shall restrict the ability to **modify** the **Public Key for CV Certification Verification** to **none** .

6.2.1.4.4 FMT_SMF.1

FMT_SMF.1.1	The TSF shall be capable of performing the following security management functions: **1. Initialisation** **2. Personalisation** **3. the "Service_Card_Management"** 4. **Modification of the PIN**

6.2.1.4.5 FMT_SMR.1

FMT_SMR.1.1	The TSF shall maintain the roles **Health Professional, Medical Assistant, Security Module Card (Health care), Self service terminal, health insurance agency service provider, combined services provider, Card holder, Download service provider, Personalisation service provider, TOE manufacturer**
FMT_SMR.1.2	The TSF shall be able to associate users with roles.

6.2.1.5 FPT – Protection of the TSF

6.2.1.5.1 FPT_EMSEC.1

FPT_EMSEC.1.1	The TOE shall not emit **electromagnetic radiation** in excess of **Unintelligible emission** enabling access to **1. PIN and PUC and** **2. Card Authentication Private Keys,** **3. Client-Sever Authentication Private Key** **4. Document Cipher Key Decipher Key** **5. 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 **1. PIN and PUC and** **2. Card Authentication Private Key,** **3. Client-Server Authentication Private Key** **4. Document Cipher Key Decipher Key** 5. **secure messaging keys .**

6.2.1.5.2 FPT_FLS.1

FPT_FLS.1.1	The TSF shall preserve a secure state when the following types of failures occur: **1. exposure to operating conditions where therefore a malfunction could occur,** 2. **self-test according to FPT_TST.1 .**

6.2.1.5.3 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 SFRs are always enforced.
Refinement :	
Devices/Elements	Physical tampering scenarios
Hardware random generator	Inappropriate random numbers
Software random generator	Modification of the secret data of the deterministic RNG
Active Shield	Physical access to or modification of internal circuits
Clock	Frequency out of allowed range
Power supply	Voltage out of allowed range
Temperature sensor	Ambient temperature out of allowed range
Light sensor	Electromagnetic irradiation
Probing sensor	Physical access to or modification of internal circuits
Glitch sensor	Short time variations in power supply

6.2.1.5.4 FPT_TST.1

FPT_TST.1.1	The TSF shall run a suite of self tests **at the conditions**: **1. Integrity verification of TSF data stored in EEPROM whenever read internally or externally.** **2. Integrity verification of TSF code patches at startup (only if personalization phase was completed)** **3. Keys and Security status stored in RAM, test of integrity whenever accessed.** **4. Test on proper operation of the underlying hardware (hardware sensors always active, sensor self test before each APDU processing, tests by software before critical operations)** **5. Testing validity flag of hardware random number generator after each retrieval.** **6. Test if Code patches are existing, done at specific points of the ROM code (hard coded)** to demonstrate the correct operation of **the TSF.**
FPT_TST.1.2	The TSF shall provide authorized users with the capability to verify the integrity of **TSF data.**
FPT_TST.1.3	The TSF shall provide authorized users with the capability to verify the integrity of stored **TSF code patches**.

6.2.1.6 FTP – Trusted path/channels

6.2.1.6.1 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 **another trusted IT product** to initiate communication via the trusted channel
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for **all functions requiring a trusted channel as defined by SFP_access_rules**.

6.3 TOE Security Assurance Requirements

The TOE security assurance requirements define the assurance requirements for the TOE using only assurance components drawn from [CCPART3].

Assurance Class	Assurance components
ADV: Development	ADV_ARC.1 Architectural Design with domain separation and non-bypassability
	ADV_FSP.4 Complete functional specification
	ADV_IMP.1 Implementation representation of the TSF
	ADV_TDS.3 Basic modular design
AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
ALC: Life-cycle support	ALC_CMC.4 Production support, acceptance procedures and automation
	ALC_CMS.4 Problem tracking CM coverage
	ALC_DEL.1 Delivery procedures
	ALC_DVS.1 Identification of security measures
	ALC_LCD.1 Developer defined life-cycle model
	ALC_TAT.1 Well-defined development tools
ASE: Security Target evaluation	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Derived security requirements
	ASE_SPD.1 Security problem definition
	ASE_TSS.1 TOE summary specification
ATE: Tests	ATE_COV.2 Analysis of coverage
	ATE_DPT.1 Testing: basic design
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing – sample
AVA: Vulnerability assessment	AVA_VAN.5 Advanced methodical vulnerability analysis

Table 17 – Assurance Requirements: EAL4 augmented with AVA_VAN.5

6.4 RATIONALE

6.4.1 Security Requirements Rationale

6.4.1.1 Security Requirement Coverage

	OT.AC_Pers	OT.Access_Rights	OT.Additional_Applications	OT.Cryptography	OT.Services	OT.Prot_Abuse-Func	OT.Prot_Inf_Leak	OT.Prot_Phys-Tamper	OT.Prot_Malfunction
FCS_CKM.1/SM				**X**	**X**				
FCS_CKM.4				**X**	**X**				
FCS_COP.1/HASH				**X**	**X**				
FCS_COP.1/CCA_SIGN				**X**	**X**				
FCS_COP.1/CCA_VERIF				**X**	**X**				
FCS_COP.1/CSA				**X**	**X**				
FCS_COP.1/ASYM_DEC				**X**	**X**				
FCS_COP.1/SYM				**X**	**X**				
FCS_COP.1/MAC				**X**	**X**				
FCS_RND.1				**X**	**X**				
FDP_ACC.2		**X**			**X**				
FDP_ACF.1		**X**			**X**				
FDP_RIP.1		**X**	**X**						
FDP_SDI.2/Persistent		**X**							
FDP_SDI.2/Volatile		**X**							
FDP_UCT.1		**X**			**X**				
FDP_UIT.1		**X**			**X**				
FIA_AFL.1/PIN		**X**			**X**				
FIA_AFL.1/PUC		**X**			**X**				
FIA_ATD.1		**X**			**X**				
FIA_UID.1	**X**	**X**			**X**				
FIA_UAU.1	**X**	**X**			**X**				
FIA_UAU.4					**X**				
FMT_LIM.1		**X**	**X**			**X**			
FMT_LIM.2		**X**	**X**			**X**			
FMT_MTD.1/Ini	**X**	**X**	**X**		**X**				
FMT_MTD.1/Pers	**X**	**X**	**X**		**X**				
FMT_MTD.1/CMS		**X**	**X**		**X**				
FMT_MTD.1/PIN		**X**	**X**		**X**				
FMT_MTD.1/KEY_MOD		**X**	**X**		**X**				
FMT_SMF.1	**X**	**X**	**X**		**X**				
FMT_SMR.1	**X**	**X**	**X**		**X**				
FPT_EMSEC.1							**X**		
FPT_FLS.1							**X**		**X**
FPT_PHP.3							**X**	**X**	**X**

	OT.AC_Pers	OT.Access_Rights	OT.Additional_Applications	OT.Cryptography	OT.Services	OT.Prot_Abuse-Func	OT.Prot_Inf_Leak	OT.Prot_Phys-Tamper	OT.Prot_Malfunction
FPT_TST.1							**X**		**X**
FTP_ITC.1		**X**			**X**				

Table 18 – Functional Requirement to TOE security objective mapping

6.4.1.2 TOE Security Requirements Sufficiency

The security objective **OT.AC_Pers** "Access control for personalisation" is implemented by following SFRs:

- the SFR FMT_SMR.1 defines the Personaliser as known role of the TOE and the SFR FMT_SMF.1 defines personalisation as security management function,
- the SFR FIA_UID.1 and FIA_UAU.1 require identification and authentication as necessary precondition for the personalisation (i.e. this TSF mediated function is not allowed before the user is identified and successfully authenticated),
- the SFR FMT_MTD.1/Pers limit right to write Personalisation data to the Personalisation service provider and
- the SFR FMT_MTD.1/INI limiting the right to write any data before personalisation to the TOE manufacturer, which in particular implies that the Personaliser role shall be created by the TOE manufacturer.

The security objective **OT.Access_Rights** is the central security requirement for the TOE. Therefore it is supported by many of the SFRs. It is mainly implemented by

- the SFRs FDP_ACC.2 and FDP_ACF.1, which require to implement the access rules defined in the security policy SFP_access_rules as defined in OT.Access_Rights,

and supported by :

- SFRs FIA_AFL.1/PIN, FIA_AFL.1/PUC, FIA_ATD.1, FMT_SMF.1, FMT_SMR.1, FMT_MTD.1/PIN, which all support the security of the Card holders eHC-PIN and PUC.
- SFRs FIA_UID.1 and FIA_UAU.1, which support timing of Identification and authentication,
- SFRs FDP_RIP.1, FDP_SDI.2/Persistent and FDP_SDI.2/Volatile (as well as all the more low-level oriented SFRs, which are not repeated here) prevent unwanted knowledge of secret data or unauthorised modification of the assets.
- the SFRs FDP_UCT.1, FDP_UIT.1 and FTP_ITC.1 provide the trusted channel for the protection of the confidentiality and integrity of transmitted data, which is required by some of the rules in SFP_access_rules.
- the SFRs FMT_MTD.1/Ini, FMT_MTD.1/Pers, FMT_MTD.1/CMS, FMT_MTD.1/KEY_MOD restrict the management of applications to authorised subjects and FMT_LIM.1 and FMT_LIM.2 prevent unauthorised use of management functions. Together they prevent the attempt to use management commands in order to bypass the access control policy.

The security objective **OT.Additional_Applications** covers the rules for the download of additional applications into the TOE. Therefore it is mainly supported by

- FMT_MTD.1/CMS, which restricts download of additional applications to the Download service provider (as also required by SFP_access_rules).

- The other SFRs on management functions FMT_SMF.1, FMT_SMR.1/, FMT_LIM.1, FMT_LIM.2, FMT_MTD.1/Ini, FMT_MTD.1/Pers, FMT_MTD.1/PIN, FMT_MTD.1/KEY_MOD support this, because they restrict other management functions to authorised subjects

- A more "low level" support is given by FDP_RIP.1, which require the deletion of secret data before any memory area is re-used. (All hardware-oriented SFRs, which are not repeated here, also support non-bypassability.)

The security objective **OT.Services** addresses the implementation and the access control of the TOE security services. The security services are implemented by the following SFR:

- the TOE security service **Service_Asym_Mut_Auth_w/o_SM** is implemented by the SFR FCS_COP.1/CCA_SIGN, FCS_COP.1/CCA_VERIF, FCS_COP.1/HASH, FCS_RND.1 and FIA_UAU.4.

- the TOE security service **Service_Asym_Mut_Auth_with_SM** is implemented by the SFR FCS_CKM.1/SM, FCS_CKM.4, FCS_COP.1/CCA_SIGN, FCS_COP.1/CCA_VERIF, FCS_COP.1/HASH, FCS_RND.1, FCS_COP.1/SYM, FCS_COP.1/MAC and FIA_UAU.4. The trusted channel established by this service is described by SFRs FDP_UCT.1, FDP_UIT.1 and FTP_ITC.1.

- the TOE security service **Service_Sym_Mut_Auth_with_SM** is implemented by the SFR FCS_CKM.1/SM, FCS_CKM.4, FCS_RND.1, FCS_COP.1/SYM, FCS_COP.1/MAC and FIA_UAU.4. The trusted channel established by this service is described by SFRs FDP_UCT.1, FDP_UIT.1 and FTP_ITC.1.

- the TOE security services **Service_User_Auth_PIN** and **Service_User_Auth_PUC** are implemented by the SFRs FIA_AFL.1/PIN, FIA_AFL.1/PUC, FIA_ATD.1, FMT_SMF.1, FMT_SMR.1, FMT_MTD.1/PIN, which all support the security of the Card holders eHC-PIN and PUC. Also it is supported by FDP_ACC.2 and FDP_ACF.1, because these SRFs require implementation of SFP_access_rules, which involves PIN authentication.

- the TOE security service **Service_Privacy** is implemented mainly by the SFRs FDP_ACC.2 and FDP_ACF.1, because the possibility for the Cardholder to deactivate sensitive medical data is defined as a rule in SFP_access_rules, which is mainly supported by these two SFRs (in fact all other SFRs supporting OT.Access_Rights, as listed for that objective, also support this services).

- the TOE security service **Service_Client_Server_Auth** is implemented by the SFR FCS_COP.1/CSA

- the TOE security service **Service_Data_Decryption** is implemented by the SFR FCS_COP.1/ASYM_DEC.

- the TOE security service **Service_Card_Management** is implemented by the SFRs already listed for the service **Service_Sym_Mut_Auth_with_SM**, because this service is used for authentication of the Download service provider and for the establishment of secure messaging for the trusted channel. Also the SFRs listed for the objective OT.Additional_Applications support this service.

- the TOE security service **Service_Logging** is implemented by access rules for the asset Logging data defined in SFP_access_rules, so it is realised mainly by the SFRs FDP_ACC.2 and FDP_ACF.1 (and in fact all other SFRs supporting OT.Access_Rights, as listed for that objective, also support this service).

The human user authentication and the access control for all of these security services is implemented mainly by the SFRs FDP_ACC.1 and FDP_ACF.1, because the policy SFP_access_control includes rules for the use of the services. (This is described in SFP_access_control in the form of rules for the use of the keys, which are relevant for the services.)

The TOE security objective **OT.Cryptography** is implemented by the SFRs of the FCS class. They include symmetric algorithms as used for secure messaging, hash functions, asymmetric algorithms and random number generation.

The security objective **OT.Prot_Inf_Leak** “Protection against information leakage” is implemented by the following SFR:

- The SFR FPT_EMSEC.1 protects user data and TSF data against information leakage through side channels.

- The SFR FPT_TST.1 detects errors and the SFR FPT_FLS.1 preserves a secure state in case of detected error which may cause information leakage e.g. trough differential fault analysis.

- The SFR FPT_PHP.3 resists physical manipulation of the TOE hardware to enforce information leakage e.g. by deactivation of countermeasures or changing the operational characteristics of the hardware.

The security objective **OT.Prot_Phys-Tamper** “Protection against physical tampering” is implemented directly by the SFR FPT_PHP.3.

The security objective **OT.Prot_Malfunction** “Protection against Malfunctions” is implemented by the following SFR:

- The SFR FPT_TST.1 detects errors and the SFR FPT_FLS.1 prevents information leakage by preserving a secure state in case of detected errors or insecure operational conditions where reliability and secure operation has not been proven or tested.

- The SFR FPT_PHP.3 resists physical manipulation of the TOE hardware controlling the operational conditions e.g. sensors.

The security objective **OT.Prot_Abuse-Func** “Protection against abuse of functionality” is implemented by the following SFR:

- The SFR FMT_LIM.1 and FMT_LIM.2 prevent the misuse of TOE functions intended for the testing, the initialisation and the personalisation of the TOE in the operational phase of the TOE,

6.4.2 Dependency Rationale for Security Functional Requirements

The following table provides an overview how the dependencies of the security functional requirements are solved and a justification why some dependencies are not being satisfied.

SFR	Dependency	Support of the dependencies
FCS_CKM.1/SM	**[FCS_CKM.2 or FCS_COP.1], FCS_CKM.4**	**FCS_COP.1/sym and FCS_COP.1/MAC Included**
FCS_CKM.4	**[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1/SM],**	**- - Included**
FCS_COP.1/HASH	**[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1/SM], FCS_CKM.4**	**Justification 1**
FCS_COP.1/CCA_SIGN	**[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1], FCS_CKM.4**	**Justification 2**
FCS_COP.1/CCA_VERIF	**[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1/SM], FCS_CKM.4**	**Justification 2**
FCS_COP.1/CSA	**[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1/SM], FCS_CKM.4**	**Justification 2**
FCS_COP.1/ASYM_DEC	**[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1/SM], FCS_CKM.4**	**justification 2**
FCS_COP.1/SYM	**[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1/SM], FCS_CKM.4**	**- - Included Included**
FCS_COP.1/MAC	**[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1/SM], FCS_CKM.4**	**- - Included Included**
FCS_RND.1	**-**	**-**
FDP_ACC.2	**FDP_ACF.1**	**Included**
FDP_ACF.1	**FDP_ACC.1,**	**FDP_ACC.2**

SFR	Dependency	Support of the dependencies
	FMT_MSA.3	Not included (justification 3)
FDP_RIP.1	-	
FDP_SDI.2/persistent	-	-
FDP_SDI.2/volatile	-	-
FDP_UCT.1	[FTP_ITC.1, or FTP_TRP.1], [FDP_ACC.1, or FDP_IFC.1]	Included - FDP_ACC.2 -
FDP_UIT.1	[FTP_ITC.1, or FTP_TRP.1], [FDP_ACC.1, or FDP_IFC.1]	Included - FDP_ACC.2 -
FIA_AFL.1/PIN	FIA_UAU.1	Included
FIA_AFL.1/PUC	FIA_UAU.1	Included
FIA_ATD.1	-	-
FIA_UID.1	-	-
FIA_UAU.1	FIA_UID.1	Included
FIA_UAU.4	-	-
FMT_LIM.1	FMT_LIM.2	Included
FMT_LIM.2	FMT_LIM.1	Included
FMT_MTD.1/INI	FMT_SMF.1, FMT_SMR.1	Included included
FMT_MTD.1/PIN	FMT_SMF.1, FMT_SMR.1	Included included
FMT_MTD.1/Pers	FMT_SMF.1, FMT_SMR.1	Included included
FMT_MTD.1/CMS	FMT_SMF.1, FMT_SMR.1	Included included
FMT_MTD.1/KEY_MOD	FMT_SMF.1, FMT_SMR.1	Included included
FMT_SMF.1	-	-
FMT_SMR.1	FIA_UID.1	Included
FPT_EMSEC.1	-	-
FPT_FLS.1	-	-
FPT_PHP.3	-	-
FPT_TST.1	-	-
FTP_ITC.1	-	-

6.4.2.1.1 Justification of unsupported security functional requirements dependencies

Justification 1 : The cryptographic algorithm for hashing does not use any cryptographic key. Therefore none of the listed SFR are needed to be defined for this specific instantiation of FCS_COP.1.
Justification 2 : The SFR FCS_COP.1/CCA_SIGN, FCS_COP.1/CCA_VERIF, FCS_COP.1/CSA and FCS_COP.1/ASYM_DEC use keys which are loaded or generated during the personalisation and are not updated or deleted over the life time of the TOE. Therefore none of the listed SFR are needed to be defined for this specific instantiations of FCS_COP.1.
Justification 3 : FDP_ACC2, justification. The access control TSF according to FDP_ACF.1 uses security attributes which are defined during the personalisation and are fixed over the whole life time of the TOE. No management of these security attribute (i.e SFR FMT_MSA.3) is necessary here.

6.4.3 Rationale for EAL 4 Augmented

The EAL 4 was chosen to permit a developer to gain maximum assurance from the positive security engineering based on good commercial development practices which, though rigorous, do not require substantial specialist knowledge, skills, and other resources. EAL4 is the highest level at which it is likely to be economically feasible to retrofit to an existing product line. EAL4 is applicable in those circumstances where developers or users require a moderate to high level of independently assured security in conventional commodity TOEs and are prepared to incur additional security specific engineering costs.

The TOE shall be shown to be highly resistant to penetration attacks with high attack potential as described in threats. Therefore the component AVA_VAN.5 was chosen in order to meet the security objectives.

The component AVA_VAN.5 has the following dependencies:

ADV_ARC.1 Architectural Design with domain separation and non-bypassability
ADV_FSP.4 Complete functional specification
ADV_TDS.3 Basic modular design
ADV_IMP.1 Implementation representation of the TSF
AGD_OPE.1 Operational user guidance
AGD_PRE.1 Preparative procedures
ATE_DPT.1 Testing: basic design

All of these dependencies are met or exceeded in the EAL4 assurance package.

7. TOE SUMMARY SPECIFICATION

The following sections describe the general technical mechanisms implemented by the TOE to meet all the requirements of the SFRs. Those are denoted in parentheses at the paragraphs that are related to them and again are listed in the last section with references to where they appear in the description, as a kind of index for the whole chapter.

7.1 LIFE CYCLE STATE MACHINE

The ES incorporates a state machine to reflect the TOE life cycle phases. It ensures the secure evolution of the TOE from the IC manufacturing phase to the usage phase. Technically the life cycle state is an integrity-protected value stored in EEPROM, coding the life cycle states VIRGIN, MODULE, PERSO, and APPLICATIVE as specified in [GeGKOS_PERS]. The life cycle state machine operating on this state value has following properties:

(i) With the IC manufacturing process this life cycle state is unconditionally set to VIRGIN.

(ii) The life cycle evolves linearly in the sequence VIRGIN → MODULE → PERSO → APPLICATIVE (FMT_SMF.1) by successful execution of the production commands (see next section 7.2). The only way backwards is a switch from PERSO to MODULE by completely deleting the EEPROM content loaded so far (especially PIN and key values already personalized).

(iii) The main distinction in life cycles is the one between the productive phases (VIRGIN, MODULE, and PERSO) and the APPLICATIVE phase. Before APPLICATIVE phase only the production commands are available (FMT_MTD.1.1/ini, FMT_MTD.1.1/pers). The switch to APPLICATIVE phase is irreversible; after this transition the applicative APDU commands are executable, but no longer the production commands. Technically the separation between production and application commands is accomplished by two different APDU dispatch routines.

7.2 PRODUCTION COMMANDS

The production of the TOE is accomplished via a dedicated set of production commands. Together with the Life Cycle State Machine they tie up the production flow as specified in [GeGKOS_PERS].
Each production command is implemented with a hard coded check for the necessary authentication state and the exact production phase(s) where it can be executed. Successful execution will process the life cycle state in a determined way.

(i) In VIRGIN state a command is available to invoke various hardware tests predefined in the ROM code (e.g. testing CPU execution, memory cells,) to detect defective chips even before beginning the production process. In this life cycle state no applicative TSF data are loaded yet and read access to ROM code is not possible (FMT_LIM.1). The test command cannot be executed after leaving the VIRGIN state, which is an irreversible life cycle switch (FMT_LIM.2).

(ii) The loading of initialization data (D.IMAGE) can only be executed in MODULE state and only after authentication with a dedicated, chip individual key only known to the TOE manufacturer (FMT_MTD.1.1/ini, FMT_SMF.1.1, FMT_SMR.1.1, FMT_SMR.1.2). After that authentication the initialization flow is as follows:

- Loading a key for image verification. This key is encrypted and integrity protected with the chip individual authentication key.
- Loading the keys for the personalization phase.

- Blockwise loading of image data and optionally also filter code (“patches”).
- Authenticating the loaded image by sending a MAC computed with the key for image verification. Only if this last step is executed successfully, proving that the initialization data are authentic and integer, the life cycle state is advanced to PERSO.

(iii) In the PERSO state the personalization service provider authenticates himself using the personalization keys that were loaded by the TOE manufacturer in the initialization phase (FMT_MTD.1.1/pers, FMT_SMF.1.1, FMT_SMR.1.1, FMT_SMR.1.2). The image contains information which personalization data have to be loaded and which of them have to be sent encrypted. Upon completion of personalization the life cycle state is irreversibly switched to APPLICATIVE state and all the productive keys are deleted.
The personalization process follows the [EMV-CPS] scheme.

7.3 INITIAL SETTINGS

During initialization phase an EEPROM image (D.IMAGE) is loaded onto the card. This image contains following preset data relevant for TOE scope:

(i) The access conditions for all card objects are set in compliance with [eHC spec part 2]. Those imply that there are no restrictions to select applications or to perform authentications by PINs or keys (FIA_UID.1, FIA_UAU.1).

(ii) Especially the access conditions for the command LOAD APPLICATION (the only way to build up additional applications, FMT_SMF.1.1) require authentication and secure messaging with keys only known to the Download Service Provider (FMT_MTD.1.1/CMS).

(iii) Initial and maximum retry counter of the card holder PINs are set to value 3 (FIA_AFL.1.1/ PIN), their minimum length is set to 6 digits.

(iv) Maximum usage counters of the card holder’s unblocking codes are set to value 10 (FIA_AFL.1.1/ PUC), their minimum length is set to 8 digits.

(v) The access conditions for updating the card holder’s pins require entering the old PIN value (FMT_MTD.1.1/PIN) by using the command CHANGE REFERENCE DATA (FMT_SMF.1.1).

(vi) The access conditions for unblocking the card holder’s pins require entering the card holder’s PUC (FMT_MTD.1.1/PIN) by using the command RESET RETRY COUNTER (FMT_SMF.1.1).

(vii) The access conditions for the public key for CV verification forbid any update of the key (FMT_MTD.1.1/KEY_MOD).

7.4 RANDOM NUMBERS

For the cryptographic computations and authentication protocols described in the following sections the TOE has to generate random numbers that meet a defined quality metric. This is achieved by utilizing the AIS31 TRNG of the hardware platform fulfilling class P2 with strength of mechanism “high” (FCS_RND.1).

7.5 CRYPTOGRAPHIC COMPUTATIONS

The ES contains a cryptographic library to implement the cryptographic procedures made available via the respective APDU commands. The basic RSA and 3-DES operation are performed by the respective hardware co-processor. The following functionalities can be executed for this TOE:

(i) Different signature schemes based on RSA with a preset key length of 2048 bit for creating and verifying signatures:
- "ISO9796-2" scheme in the two modes DS1 and DS2,
- "PKCS#1v1_5" scheme, and
- "PKCS#1-PSS" scheme,

where exclusively the SHA-256 algorithm is used in internal hash computations (FCS_COP.1/HASH).

These fundamental RSA schemes are used in following functionalities:

(ii) Signature generation with all three schemes (FCS_COP.1/CCA_SIGN and FCS_COP.1/CSA).

(iii) Verification of CV certificates according ISO 9796-2 to import transient public keys used in a subsequent (asymmetric) component authentication (FCS_COP.1/CCA_VERIF).

(iv) Asymmetric authentication according ISO9796-2 (DS1) (FCS_COP.1/CCA_SIGN and FCS_COP.1/CCA_VERIF).

(v) Client/Server authentication according ISO9796-2 (DS2) , PKCS#1-v1_5, or PKCS#1-PSS (FCS_COP.1/CSA).

Further cryptographic operations are:

(vi) Data deciphering with PKCS#1v1_5 padding and RSA OAEP (FCS_COP.1/ASYM_DEC).

(vii) Three-Key-TripleDES with a key length of 168 bit (3TDES) in following modes (FCS_COP.1/SYM and FCS_COP.1/MAC):
- 3TDES in CBC mode for message encryption the symmetric authentication protocol,
- 3TDES in CBC mode for message encryption in trusted channels (FDP_UCT.1),
- RetailMAC in CBC mode in the symmetric authentication protocol, and
- RetailMAC in CFB mode for message integrity in trusted channels (FDP_UIT.1).

The last three are executed with message padding according [ISO-C4] 5.6.3.1 ("ISO-Padding").

7.6 CARD HOLDER AUTHENTICATION

(i) The Card Holder authenticates himself by correctly presenting PIN.CH or PIN.home via the ISO APDU command VERIFY. These PINs have a preset retry counter of 3 (FIA_AFL.1.1/PIN) and a minimum length of 6 digits. On correct PIN presentation an associated security state is established, which represents the card holder's identity (FIA_ATD.1.1, FMT_SMR.1.2) and is referenced by the access conditions relevant for the card holder.

(ii) After successive wrong PIN presentation exceeding the retry counter the PIN is blocked so that no more PIN authentication can be achieved, even by presenting the correct PIN value (FIA_AFL.1.2/PIN).

(iii) For each of PIN.CH and PIN.home there is an associated unblocking code with a minimum length of 8 digits. Each one can be used 10 times to unblock the associated PIN in case it got blocked (FIA_AFL.1.1/PUC). After the 10^{th} usage, regardless whether unsuccessful or not, the unblocking code itself gets irreversibly blocked and can no more be used then (FIA_AFL.1.2/PUC).

(iv) Directly after card production a PIN might be in transport state, depending on the personalization data. In this state it is not possible to establish the security state for that PIN. The card holder first has to replace the transport PIN by his preferred PIN, which must have at least the minimum PIN length preset. Only after this replacement the security state for this PIN can be set. It is not possible

to switch the PIN back to transport state. The ISO command CHANGE REFERENCE DATA is used to replace the PIN. It is also used to select a new PIN value by presenting the old value, what restricts that operation to the card holder (FMT_MTD.1.1/PIN).

7.7 ASYMMETRIC AUTHENTICATION

Asymmetric authentication is used by the components of the health professionals to prove their authenticity to the card and optionally to secure the subsequent communication. The authentication protocol is as follows:

(i) If the public key of the external component's CA is not available inside the TOE, the corresponding certificate (containing that key) is entered (via APDU command PSO VERIFY CERTIFICATE). On successful certificate check (via FCS_COP.1/CCA_VERIF) with the root key the public key of the external component's CA is temporally stored in the TOE.

(ii) The certificate of the external component's public key is entered. On successful certificate check (via FCS_COP.1/CCA_VERIF) with the CA key the public key of the external component is temporally stored in the TOE. By the name of the entered key (CHA) the external component is identified (FIA_UAU.1.1, FMT_SMR.1).

(iii) With the command sequence INTERNAL AUTHENTICATE, GET CHALLENGE, EXTERNAL AUTHENTICATE (using FCS_RND.1, FCS_COP.1/CCA_SIGN, and FCS_COP.1/CCA_VERIF) a mutual asymmetric, one time challenge-response authentication is performed (FIA_UAU.4.1).

A successful authentication has following effects:

(iv) The authentication state for the entered external public key is set, representing the corresponding role (FIA_ATD.1.1, FMT_SMR.1.2). That authentication state is evaluated when checking the access to the card data.

(v) If indicated by the algorithm selected for the authentication protocol, volatile session keys are negotiated from the random numbers exchanged (FCS_CKM.1.1/SM) to establish a trusted channel for securing the subsequent communication via Secure Messaging.

7.8 SYMMETRIC ADMINISTRATOR AUTHENTICATION

In usage phase the administrator can authenticate himself by a symmetric one-time challenge-response protocol with the command sequence GET CHALLENGE and MUTUAL AUTHENTICATE.

(i) Before executing that protocol the download service provider (CMS) identifies himself by selecting the corresponding key via key-ID (FIA_UAU.1.1). With the selected key the symmetric authentication protocol is performed (utilizing FCS_RND.1, FCS_COP.1/SYM, and FCS_COP.1/MAC). The involved challenge prevents the reuse of a successful authentication attempt (FIA_UAU.4.1).

A successful authentication has following effects:

(ii) The authentication state for the selected key is set, identifying and representing the corresponding role (FMT_SMR.1.2). That authentication state is evaluated when checking the access to the card data.

(iii) Volatile session keys are negotiated from the random numbers exchanged (FCS_CKM.1.1/SM) to establish a trusted channel for securing the subsequent communication via Secure Messaging.

7.9 ACCESS MANAGEMENT

As this product is a smart card complying with ISO 7816 the external world can only communicate with it via APDU commands. No direct access to the resources of the smart card, which in essence are file contents, PINs, and keys, is possible.

(i) In productive phases the access check is hard wired within the production commands (FDP_ACC.2, FMT_SMR.1.1) and determined by the life cycle state, see sections 7.1 and 7.2.

(ii) In the usage phase all resources are equipped with access rules to mediate the access to them (FDP_ACC.2). The access rules are preset with the EEPROM image loaded, in compliance with [eHC spec part 2] (FDP_ACF.1), and are evaluated on each APDU command before the intended functionality is invoked. Access rules consist of a Boolean combination of single "access conditions". Those access conditions can specify:

- the presence of component or administrator authentications executed before, represented by so called security states (FMT_SMR.1.1) referring to the key each authentication had been performed with,
- the presence of a Card Holder authentication executed before (with PIN.CH or PIN.home), represented by so called security state (FMT_SMR.1.1) referring to the PIN the authentication had been performed with,
- presence of secure messaging with volatile session keys established with a preceding component or administrative authentication (FDP_UCT.1, FDP_UIT.1, FTP_ITC.1.3),
- and any Boolean combination of those.

7.10 SECURE MESSAGING

This component provides the functionality to ensure protection of the data exchanged via APDUs by authenticity, integrity and confidentiality, (trusted channel) using 3TDES cryptography.

(i) The authenticity and integrity is ensured by adding a Message Authentication Code (MAC) to the data (FDP_UIT.1).

(ii) The confidentiality is achieved by encrypting the exchanged data (FDP_UCT.1).

(iii) The Secure Messaging uses the volatile session keys that were negotiated in the preceding authentication protocols executed by administrator (symmetric) or health professional's component (asymmetric) (FTP_ITC.1.1).

(iv) Once the session keys are established to form a trusted channel with the authenticated external IT product, any command APDU may be sent by the external IT product with Secure Messaging using those session keys (FTP_ITC.1.2).

(v) The need to use Secure Messaging is governed by the access conditions set for the resource to be accessed. MAC and/or encryption must be present in command or response APDUs if listed in the access conditions (FTP_ITC.1.3), but may still be present if not listed.

7.11 TSF PROTECTION

The ES is designed to protect the TOE against fraudulent attacks. Supported by the security features of the platform the following general mechanisms are in place:

(i) On each reset the TOE is set to a secure state before the normal operation of the TSF starts, even after an unexpected abortion of TSF execution or TOE halt in response to attack detection (FPT_FLS.1.1). This includes the deletion of any session keys and security states established by authentication from users or components.

(ii) If during TSF execution an unexpected error occurs, the secure state of the TSF will be preserved by halting their execution. Such a halt state can only be left by a reset (FPT_FLS.1.1), what will set the TOE to a secure state again (see above).

(iii) Before the execution of the first APDU after start-up, the integrity of code patches is verified (FPT_TST.1.1).

(iv) During execution of the TSF at specific points it is checked if a relevant code patch is existing and in such a case it is executed (FPT_TST.1.1).

(v) The ES utilizes the hardware platform's protection and self check features like clock jitter or environmental sensors. Detection of faults leads to a TOE halt (FPT_PHP.3).

(vi) When retrieving random bytes from the hardware platform's TRNG the corresponding validity flag is evaluated (FPT_TST.1.1). A detected fault would result in a TOE halt.

(vii) Before critical operations the ES executes a routine to check hardware registers and undisturbed hardware operation (FPT_TST.1.1). Detected faults would result in a TOE halt. Also some desynchronization by software via random delay loops is done regularly.

(viii) All data in non-volatile memory are equipped with a checksum to detect integrity faults. While this feature can be deactivated for applicative data files at file creation time (Loading of D.IMAGE or card management by download service provider), it can not be deactivated for sensitive objects. While integrity errors of file contents to read would result in a warning on a read attempt, for sensitive objects like PIN and key values, life cycle state, access conditions, and patch code, the TOE execution would be halted (FDP_SDI.2/Persistent, FPT_TST.1.1, FPT_TST.1.2).

(ix) Security relevant data temporarily stored in RAM are also secured by a checksum: security states, session keys, external public keys, and transient RAM copies of non-volatile keys. In the case of an integrity error the TOE execution would be halted, muting the card (FDP_SDI.2/Volatile, FPT_TST.1.1, FPT_TST.1.2).

(x) Session keys, RAM copies of private or secret keys, and volatile PIN data are explicitly erased as soon as they are no longer needed (FDP_RIP.1.1, for keys also FCS_CKM.4.1).

(xi) Sensitive data, especially keys and PIN values (D.RAD), are stored in a protected form: the data are masked so even in case an attacker succeeds in retrieving a memory dump those data are not available in plain (FPT_PHP.3).

(xii) Sensitive operations like the RSA and 3TDES computations or PIN verification are programmed in a way that processing timing, electromagnetic radiation, or power consumption of the chip cannot be used to discover any PIN or secret/private key (FPT_EMSEC.1).

(xiii) All sensitive code flows are secured by redundant branch checks, secure variable values, and execution tracing to permanently protect the TOE against physical tampering (FPT_TST.1.1).

(xiv) The ISO file system handling of GeGKOS provides a natural way to separate the data structures between applications (domain separation): An application is represented by a dedicated application DF and its child EFs. For the given eHC applications the applicative data are not accessible from outside the current application DF. Furthermore, the file system is completely separated form TSF internal data like counter measure configuration.

(xv) In case that a file (DF or EF) is explicitly deleted, the associated memory area is cleared directly at deletion time, making the previous information content unavailable (FDP_RIP.1.1). This automatically covers PINs and keys: The ES (GeGKOS) stores them in regular EFs.

7.12 COVERAGE OF SFRS

Requirement	Covering Location in Summary Specifiction
FCS_CKM.1/SM	7.7(v) and 7.8(iii): negotiation of session keys in authentication protocols.
FCS_CKM.4	7.11(x): keys in RAM are deleted as soon as possible.
FCS_COP.1/HASH	7.5(i)
FCS_COP.1/CCA_SIGN	7.5(ii),(iv), indirectly also in 7.7(iii)
FCS_COP.1/CCA_VERIF	7.5(iii),(iv) indirectly also in 7.7(i)-(iii)
FCS_COP.1/CSA	7.5(v), 7.5(ii)
FCS_COP.1/ASYM_DEC	7.5(vi)
FCS_COP.1/SYM	7.5(vii), indirectly also in7.8(i)
FCS_COP.1/MAC	7.5(vii), indirectly also in 7.8(i)
FCS_RND.1	7.4, indirectly also in 7.7(iii) and 7.8(i)
FDP_ACC.2	7.9(i): hard wired access check for productive commands. 7.9(ii): unconditional access checking for usage phase commands.
FDP_ACF.1	7.9(ii): access conditions loaded with EEPROM image.
FDP_RIP.1	7.11(xv): explicit erasing of deallocated EEPROM memory. 7.11(x): explicit erasing of sensitive RAM as soon as no longer needed.
FDP_SDI.2/persistent	7.11(viii): Checksum on persistent data for integrity check.
FDP_SDI.2/volatile	7.11(ix): Checksum on transient data for integrity check.
FDP_UCT.1	7.5(vii): 3TDES encryption in trusted channel, in connection with access condition check 7.9(ii) for Secure Messaging 7.10(ii).
FDP_UIT.1	7.5(vii): RetailMAC in trusted channel, in connection with access condition check 7.9(ii) for Secure Messaging 7.10(i).
FIA_AFL.1/PIN	7.3(iii) and 7.6(i),(ii): PIN with preset retry counter.
FIA_AFL.1/PUC	7.3(iv) and 7.6(iii): unblocking code for blocked PIN.
FIA_ATD.1	7.6(i): security state of PINs associated with card holder.
FIA_UID.1	See FIA_UAU.1.
FIA_UAU.1	7.7(ii) roles in usage phase identified by CHA in entered certificate. 7.8(i) identification for CMS operations in usage phase via key selection. Otherwise (entering card holder PIN or authentication in production phases) identification and authentication is one step. The image sets access conditions for the TSF-mediated actions 7.3(i); there are no restrictions to execute the authentication by PIN or keys.
FIA_UAU.4	7.7(iii) and 7.8(i): card's challenge is part of the authentication protocols.
FMT_LIM.1	7.2(i): test features can not access TSF data.
FMT_LIM.2	7.2(i): test features can only executed in very first production step.

Requirement	Covering Location in Summary Specifiction
FMT_MTD.1/INI	7.1(iii): separation from roles in usage phase also by TOE life cycle. 7.2(ii): only EEPROM images authenticated by the TOE manufacturer can be loaded
FMT_MTD.1/PIN	7.3(v)(vi) and 7.6(iv) : Replacement of PINs only by entering old PIN value, enforced by access conditions.
FMT_MTD.1/Pers	7.1(iii): separation from roles in usage phase also by TOE life cycle 7.2(iii): personalization is secured by life cycle and dedicated keys.
FMT_MTD.1/CMS	7.3(ii): application loading is secured by access conditions in EEPROM image.
FMT_MTD.1/KEY_MOD	7.3(vii): the access conditions set do not allow write access to CV verification key.
FMT_SMF.1	7.1(ii): initialization, personalization, and usage phase separated by TOE life cycle 7.2: initialization and personalization can be performed with the production commands. 7.3(ii)(v)(vi): card management and PIN modification are controlled by access conditions.
FMT_SMR.1	7.7(ii) roles in usage phase identified by CHA in entered certificate. 7.2: roles in card production are hard coded in the production commands. 7.6(i), 7.7(iv), 7.8(ii), 7.9: roles in usage phase are distinguished by authentication states and associated access condition check.
FPT_EMSEC.1	7.11(xii) unintelligible electromagnetic radiation. 7.11 in general to detect attacks in hardware and software.
FPT_FLS.1	7.11(i)(ii) Secure state after reset and unexpected errors.
FPT_PHP.3	7.11(v)(xi)
FPT_TST.1	7.11 in general.
FTP_ITC.1	7.9(ii), 7.10: Secure Messaging in connection with access condition check.

8. COMPOSITION TASKS

8.1 SFR PART

The following table lists the SFRs that are declared in the SLE78CX800P security target [ST IC], and separates them in relevant platform-SFRs (RP_SFR) and irrelevant platform-SFRs (IP_SFR).
The table also provides the link between the relevant platform-SFRs and the composite product SFRs.

Platform SFR	Platform SFR Content	Platform SFR additional Information	RP_ SFR	IP_ SFR	Composite product SFRs
FRU_FLT.2	**Limited fault tolerance**: The TSF shall ensure the operation of all the TOE's capabilities when the following failures occur: exposure to operating conditions which are not detected according to the requirement Failure with preservation of secure state (FPT_FLS.1).	SF_PMA	X		FPT_FLS.1 FPT_PHP.3
FPT_FLS.1	**Failure with preservation of secure state**: The TSF shall preserve a secure state when the following types of failures occur: exposure to operating conditions which may not be tolerated according to the requirement Limited fault tolerance (FRU_FLT.2) and where therefore a malfunction could occur.	SF_PS, SF_PMA, SF_PLA, SF_CS	X		FPT_FLS.1 FPT_PHP.3
FMT_LIM.1	The TSF shall be designed and implemented in a manner that limits their capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced: Limited capability and availability Policy.	SF_DPM	X		No direct link to any composite-product SFR - used "transparently"
FMT_LIM.2	The TSF shall be designed and implemented in a manner that limits their availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced: Limited capability and availability Policy.	SF_DPM	X		No direct link to any composite-product SFR.-used "transparently"
FAU_SAS.1	The TSF shall provide the test process before TOE Delivery with the capability to store the Pre-Initialization Data and / or Initialization and / or supplements of the Security IC Embedded Software in the not changeable configuration page area and non-volatile memory.	SF_DPM	X		No direct link to any composite-product SFR.-used "transparently"
FPT_PHP.3	The TSF shall resist physical manipulation and physical probing, to the TSF by responding automatically such that the SFRs are always enforced.	SF_DPM, SF_PS, SF_PMA, SF_PLA, SF_CS	X		FPT_FLS.1 FPT_PHP.3
FDP_ITT.1	The TSF shall enforce the Data Processing Policy to prevent the disclosure of user data when it is transmitted between physically-separated parts of the TOE.	SF_DPM, SF_PS, SF_PMA, SF_PLA, SF_CS	X		FPT_FLS.1 FPT_PHP.3
FPT_ITT.1	The TSF shall protect TSF data from disclosure when it	SF_DPM,	X		FPT_FLS.1

Platform SFR	Platform SFR Content	Platform SFR additional Information	RP_ SFR	IP_ SFR	Composite product SFRs
	is transmitted between separate parts of the TOE. The different memories, the CPU and other functional units of the TOE (e.g.a cryptographic co-processor) are seen as separated parts of the TOE.	SF_PS, SF_PMA, SF_CS			FPT_PHP.3
FDP_IFC.1	The TSF shall enforce the Data Processing Policy on all confidential data when they are processed or transferred by the TSF or by the Security IC Embedded Software.	SF_PS, SF_PMA, SF_PLA	X		FPT_FLS.1 FPT_PHP.3
FCS_RNG.1	The TSF shall provide a physical random number generator that implements total failure test of the random source.	SF_CS	X		FCS_RND.1
FPT_TST.2	The TSF shall run a suite of self tests at the request of the authorized user to demonstrate the correct operation of the alarm lines and/or following environmental sensor mechanisms	SF_PMA, SF_CS	X		FPT_FLS.1 FPT_PHP.3
FDP_ACC.1	The TSF shall enforce the Memory Access Control Policy on all subjects (software running at the defined and assigned privilege levels), all objects (data including code stored in memories) and all the operations defined in the Memory Access Control Policy, i.e. privilege levels.	SF_DPM, SF_PMA, SF_PLA	X		FPT_FLS.1 FPT_PHP.3
FDP_ACF.1	The TSF shall enforce the Memory Access Control Policy to objects based on the following: Subject: - software running at the IFX, OS1 and OS2 privilege levels required to securely operate the chip. This includes also privilege levels running interrupt routines. - software running at the privilege levels containing the application software Object: - data including code stored in memories Attributes: - the memory area where the access is performed to and/or - the operation to be performed. The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: evaluate the corresponding permission control information of the relevant memory range before, during or after the access so that accesses to be denied can not be utilized by the subject attempting to perform the operation.	SF_DPM, SF_PMA, SF_PLA	X		No direct link to any composite-product SFR.-used “transparently”
FMT_MSA.1	The TSF shall enforce the Memory Access Control Policy to restrict the ability to change default, modify or delete the security attributes permission control information to the software running on the privilege levels.	SF_DPM, SF_PMA, SF_PLA	X		No direct link to any composite-product SFR.-used “transparently”
FMT_MSA.3	The TSF shall enforce the Memory Access Control Policy to provide well defined default values for security attributes that are used to enforce the SFP. The TSF shall allow any subject, provided that the Memory AccessControl Policy is enforced and the necessary access is therefore allowed, to specify alternative initial values to override the default values when an object or information is created.	SF_DPM, SF_PMA, SF_PLA	X		No direct link to any composite-product SFR.-used “transparently”

Platform SFR	Platform SFR Content	Platform SFR additional Information	RP_ SFR	IP_ SFR	Composite product SFRs
FMT_SMF.1	The TSF shall be capable of performing the following security management functions: access the configuration registers of the MMU.	SF_DPM, SF_PMA, SF_PLA	X		No direct link to any composite-product SFR.-used “transparently”
FCS_COP.1/ DES	The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Triple Data Encryption Standard (3DES) in the Electronic Codebook Mode (ECB), in the Cipher Block Chaining Mode (CBC), in the Blinding Feedback Mode (BLD) and in the Cipher Feedback Mode (CFB)and with cryptographic key sizes of 2 x 56 bit or 3 x 56 bit, that meet the following standards: National Institute of Standards and Technology (NIST), Technology Administration, U.S. Department of Data Encryption Standard (DES),NIST Special Publication 800-67, Version 1.1	SF_CS	X		FCS_COP.1 ES does not use these functionalities but the ES uses the hardware accelerators for cryptographic computations.
FCS_COP.1/ AES	The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Advanced encryption Standard (AES) and cryptographic key sizes of 128 bit or 192 bit or 256 bit that meet the following standards: U.S. Department of Commerce, National Institute of Standards and Technology, Information Technology Laboratory (ITL),Advanced Encryption Standard (AES), FIPS PUB 197	SF_CS		X	ES does not use these functionalities.
FCS_COP.1/ ECDSA	The TSF shall perform signature generation and signature verification in accordance with a specified cryptographic algorithm ECDSA and cryptographic key sizes 192 - 521 bits that meet the following standard: Signature Generation: 1. According to section 7.3 in ANSI X9.62 – 2005 Not implemented is step d) and e) thereof. The output of step e) has to be provided as input to our function by the caller. Deviation of step c) and f): The jumps to step a) were substituted by a return of the function with an error code, the jumps are emulated by another call to our function. 2. According to sections 6.2 (6.2.2. + 6.2.3) in ISO/IEC 15946-2:2002 Not implemented is section 6.2.1: The output of 5.4.2 has to be provided by the caller as input to the function. Signature Verification: 1. According to section 7.4.1 in ANSI X9.62–2005 Not implemented is step b) and c) thereof. The output of step c) has to be provided as input to our function by the caller. Deviation of step d): Beside noted calculation, our algorithm adds a random multiple of BasepointerOrder n to the calculated values u1 and u2. 2. According to sections 6.4 (6.4.1. + 6.4.3 + 6.4.4) in ISO/IEC 15946-2:2002 Not implemented is section 6.4.2: The output of 5.4.2 has to be provided by the caller as input to the function.	FS_CS		X	ES does not use these functionalities.

Platform SFR	Platform SFR Content	Platform SFR additional Information	RP_ SFR	IP_ SFR	Composite product SFRs
FCS_COP.1/ ECDH	The TSF shall perform elliptic curve Diffie-Hellman key agreement in accordance with a specified cryptographic algorithm ECDH and cryptographic key sizes 192 - 521 bits that meet the following standard: 1. According to section 5.4.1 in ANSI X9.63 -2001 Unlike section 5.4.1.3 our, implementation not only returns thex-coordinate of the shared secret, but rather the x-coordinate and y-coordinate. 2. According to sections 8.4.2.1, 8.4.2.2, 8.4.2.3, and 8.4.2.4 in ISO/IEC 15946-3:2002: The function enables the operations described in the four sections.	FS_CS		X	ES does not use these functionalities.
FCS_COP.1/ SHA	The TSF shall perform hash-value calculation of user chosen data in accordance with a specified cryptographic algorithm SHA-2 and with cryptographic key sizes of none that meet the following standards: U.S. Department of Commerce / National Bureau of Standards Secure Hash Algorithm, FIPS PUB 180-3, 2008-October, section 6.2 SHA-256 and section 6.4 SHA-512.	FS_CS	X		FCS_COP.1 ES does not use these functionalities. But the ES uses the hardware accelerators for cryptographic computations.
FCS_COP.1/ RSA	The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Rivest-Shamir-Adleman (RSA) and cryptographic key sizes 1024 - 4096 bits that meet the following Standards Encryption: According to section 5.1.1 RSAEP in PKCS v2.1 RFC3447,without 5.1.1.1. Decryption (with or without CRT): According to section 5.1.2 RSADP in PKCS v2.1 RFC3447 for u = 2, i.e., without any (r_i, d_i, t_i), i >2, therefore without 5.1.2.2.b (ii)&(v), without 5.1.2.1. 5.1.2.2.a, only supported up to n < 22048 Signature Generation (with or without CRT):: According to section 5.2.1 RSASP1 in PKCS v2.1 RFC3447 for u = 2, i.e., without any (r_i, d_i, t_i), i >2, therefore without 5.2.1.2.b (ii)&(v), without 5.2.1.1. 5.2.1.2.a, only supported up to n < 22048 Signature Verification: According to section 5.2.2 RSAVP1 in PKCS v2.1 RFC3447, without 5.2.2.1.	FS_CS	X		FCS_COP.1 ES does not use these functionalities but the ES uses the hardware accelerators for cryptographic computations.
FCS_CKM.1/ RSA	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm rsagen1 (PKCS v2.1 RFC3447) and specified cryptographic key sizes of 1024 – 4096 bits that meet the following standard: According to section	FS_CS		X	The TOE does not use the manufacturer library.

Platform SFR	Platform SFR Content	Platform SFR additional Information	RP_ SFR	IP_ SFR	Composite product SFRs
	3.2(2) in PKCS v2.1 RFC3447, for u=2, i.e., without any (r_i, d_i, t_i), i > 2. For p x q < 22048 additionally according to section 3.2(1).				
FCS_CKM.1/ EC	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm Elliptic Curve EC specified in ANSI X9.62-2005 and ISO/IEC 15946-1:2002 and specified cryptographic key sizes 192 - 521 bits that meet the following standard ECDSA Key Generation: 1. According to the appendix A4.3 in ANSI X9.62-2005 the cofactor h is not supported. 2. According to section 6.1 (not 6.1.1) in ISO/IEC 15946-1:2002	FS_CS		X	The TOE does not use the manufacturer library.
FDP_SDI.1	The TSF shall monitor user data stored in containers controlled by the TSF for inconsistencies between stored data and corresponding EDC on all objects, based on the following attributes: EDC value for the RAM, ROM and EEPROM	SF_PMA	X		FPT_FLS.1 FPT_PHP.3
FDP_SDI.2	The TSF shall monitor user data stored in containers controlled by the TSF for data integrity and one- and/or more-bit-errors on all objects, based on the following attributes: corresponding EDC value for RAM, ROM and EEPROM and error correction ECC for the EEPROM. Upon detection of a data integrity error, the TSF shall correct 1 bit errors in the EEPROM automatically and inform the user about more bit errors.	SF_PMA	X		FPT_FLS.1 FPT_PHP.3

Table 19 – Composition – SFR part

SF_DPM : Device Phase Management
Transparent
SF_PS : Protection against snooping
Transparent
SF_PMA : Protection against Modifications attacks
The ES calls the UMSLC test e.g. before RSA crypto operations.
SF_PLA : Protection against logical attacks
Transparent
SF_CS : Cryptographic Support
The ES uses the hardware accelerators for cryptographic computations

8.2 THREATS PART

IC threat label	IC threat title	IC threat content	Link to the composite product
T.Phys-Manipulation	Physical Manipulation	An attacker may physically modify the Security IC in order to (i) modify User Data (ii) modify the Security IC Embedded Software (iii) modify or deactivate security services of the TOE, or (iv) modify security mechanisms of the TOE to enable attacks disclosing or manipulating the User Data or the Security IC Embedded	T.Phys_Tamper T.Information_Leakage
T.Phys-Probing	Physical Probing	An attacker may perform physical probing of the TOE in order (i) to disclose User Data (ii) to disclose/reconstruct the Security IC Embedded Software or (iii) to disclose other critical information about the operation of the TOE to enable attacks disclosing or manipulating the User Data or the Security IC Embedded	T.Phys_Tamper T.Information_Leakage
T.Malfunction	Malfunction due to Environmental Stress	An attacker may cause a malfunction of TSF or of the Security IC Embedded Software by applying environmental stress in order to (i) modify security services of the TOE or (ii) modify functions of the Security IC Embedded Software (iii) deactivate or affect security mechanisms of the TOE to enable attacks disclosing or manipulating the User Data or the Security IC Embedded Software. This may be achieved by operating the Security IC outside the normal operating conditions.	T.Malfunction
T.Leak-Inherent	Inherent Information Leakage	An attacker may exploit information which is leaked from the TOE during usage of the Security IC in order to disclose confidential User Data as part of the assets. No direct contact with the Security IC internals is required here. Leakage may occur through emanations, variations in power consumption, I/O characteristics, clock frequency, or by changes in processing time requirements.	T.Information_Leakage
T.Leak-Forced	Forced Information Leakage	An attacker may exploit information which is leaked from the TOE during usage of the Security IC in order to disclose confidential User Data as part of the assets even if the information leakage	T. Information_Leakage T.Phys_Tamper

IC threat label	IC threat title	IC threat content	Link to the composite product
		is not inherent but caused by the attacker.	
T.Mem-Access	Memory Access Violation	Part of the Smartcard Embedded Software may cause security violations by accidentally or deliberately accessing restricted data (which may include code) or privilege levels. Any restrictions are defined by the security policy of the specific application context and must be implemented by the Smartcard Embedded Software.	T.Phys_Tamper
T.Abuse-Func	Abuse of Functionality	An attacker may use functions of the TOE which may not be used after TOE Delivery in order to (i) disclose or manipulate User Data (ii) manipulate (explore, bypass, deactivate or change) security services of the TOE or (iii) manipulate (explore, bypass, deactivate or change) functions of the Security IC Embedded Software or (iv) enable an attack disclosing or manipulating the User Data or the Security IC Embedded Software.	T.Phys_Tamper
T.RND	Deficiency of Random Numbers	An attacker may predict or obtain information about random numbers generated by the TOE security service for instance because of a lack of entropy of the random numbers provided.	T.Forge_Internal_Data

Table 20 – Composition – Threats part

8.3 OSP PART

IC OSP label	IC OSP content	Link to the composite product
P.Process-TOE	Protection during TOE Development and Production: An accurate identification must be established for the TOE. This requires that each instantiation of the TOE carries this unique identification.	No contradiction with the present evaluation; the chip traceability information is used to identify the composite TOE.
P.Add-Functions	Additional Specific Security Components: The TOE shall provide the following specific security functionality to the Smartcard Embedded Software: • Advanced Encryption standard (AES) • Triple Data Encryption Standard (3DES) • Rivest-Shamir-Adleman Cryptography (RSA) • Elliptic Curve Cryptography (EC) • Secure Hash Algorithm SHA-2	Platform provides the following specific security functionality to the Security IC Embedded Software: - Triple Data Encryption Standard (3DES), - *Rivest-Shamir-Adleman Cryptography (RSA)* - Area based Memory Access Control - Secure Hash Algorithm SHA-2 The ES uses the hardware accelerators for cryptographic computations.

Table 21 – Composition – OSPs part

8.4 ASSUMPTIONS PART

IC assumption label	IC assumption title	IC assumption content	IrPA	CfPA	SgPA	Link to the composite product
A.Process-Sec-IC	Protection during Packaging, Finishing and Personalisation	It is assumed that security procedures are used after delivery of the TOE by the TOE Manufacturer up to delivery to the endconsumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorized use).			X	A.Perso OT.AC_Pers
A.Plat-Appl	Usage of Hardware Platform	The Security IC Embedded Software is designed so that the requirements from the following documents are met: (i) TOE guidance documents (refer to the Common Criteria assurance class AGD) such as the hardware data sheet, and the hardware application notes, and (ii) findings of the TOE evaluation reports relevant for the Security IC Embedded Software as documented in the certification report.		**X**		Fulfilled by the composite-SAR ADV_COMP.1 (cf [CCDB], Appendix 1.2, §72 and §73)
A.Resp-Appl	Treatment of User Data	All User Data are owned by Security IC Embedded Software. Therefore, it must be assumed that security relevant User Data (especially cryptographic keys) are treated by the Security IC Embedded Software as defined for its specific appli*cation* context.		**X**		OT.Prot_Phys_Tamper OT.Prot_Inf_Leak OT.Prot_Abuse_Func OT.Access_rights
A.Key-Function	Usage of key-dependent functions	Key-dependent functions (if any) shall be implemented in the Smartcard Embedded Software in a way that they are not susceptible to leakage attacks (as described under T.Leak-Inherent and T.Leak-Forced). Note that here the routines which may compromise		**X**		OT.Prot_Phys_Tamper OT.Prot_Inf_Leak

IC assumption label	IC assumption title	IC assumption content	IrPA	CfPA	SgPA	Link to the composite product
		keys when being executed are part of the Smartcard Embedded Software. In contrast to this the threats T.Leak-Inherent and T.Leak-Forced address (i) the cryptographic routines which are part of the TOE.				

Table 22 – Composition – Assumptions part

IrPA means "*The assumptions being not relevant for the Composite-ST, e.g. the assumptions about the developing and manufacturing phases of the platform.*"
CfPA means "*The assumptions being fulfilled by the Composite-ST automatically. Such assumptions of the Platform-ST can always be assigned to the TOE security objectives of the Composite-ST. Due to this fact they will be fulfilled either by the Composite-TSF or by the Composite-TAM automatically.*"
SgPA means "*The remaining assumptions of the Platform-ST belonging neither to the group IrPA nor CfPA. Exactly this group makes up the significant assumptions for the Composite-ST, which shall be included into the Composite-ST.*"

8.5 SECURITY OBJECTIVES FOR THE TOE PART

IC TOE security objective Label	IC TOE security objective Title	IC TOE security objective Content	Link to the composite-product
O.Phys-Manipulation	Protection against Physical Manipulation	The TOE must provide protection against manipulation of the TOE (including its software and Data), the Security IC Embedded Software and the User Data. This includes protection against - reverse-engineering (understanding the design and its properties and functions), - manipulation of the hardware and any data, as well as - controlled manipulation of memory contents (Application Data).	OT.Prot_Phys_tamper OT.Prot_Inf_Leak
O.Phys-Probing	Protection against Physical Probing	The TOE must provide protection against disclosure of User Data, against the disclosure/reconstruction of the Security IC Embedded Software or against the disclosure of other critical information about the operation of the TOE. This includes protection against - measuring through galvanic contacts which is direct physical probing on the chips surface except on pads being bonded (using standard tools for measuring voltage and current) or - measuring not using galvanic contacts but other types of physical interaction between charges (using tools used in solid-state physics research and IC failure analysis) with a prior reverse-engineering to understand the design and its properties and functions.	OT.Prot_Phys_tamper OT.Prot_Inf_Leak
O.Malfunction	Protection against Malfunctions	The TOE must ensure its correct operation. The TOE must indicate or prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent malfunctions. Examples of environmental conditions are voltage, clock frequency, temperature, or external energy fields.	OT.Prot_Malfunction
O.Leak-Inherent	Protection against Inherent Information Leakage	The TOE must provide protection against disclosure of confidential data stored and/or processed in the Security IC - by measurement and analysis of the shape and amplitude of signals (for example on the power, clock, or I/O lines) and - by measurement and analysis of the time between events found by measuring signals (for instance on the power, clock, or I/O lines). This objective pertains to measurements with subsequent complex signal processing whereas O.Phys-Probing is about direct measurements on elements on the chip surface. Details correspond to an analysis of attack scenarios which is not given here.	OT.Prot_Inf_Leak

IC TOE security objective Label	IC TOE security objective Title	IC TOE security objective Content	Link to the composite-product
O.Leak-Forced	Protection against Forced Information Leakage	The Security IC must be protected against disclosure of confidential data processed in the Security IC (using methods as described under O.Leak-Inherent) even if the information leakage is not inherent but caused by the attacker - by forcing a malfunction (refer to "Protection against Malfunction due to Environmental Stress (O.Malfunction)" and/or - by a physical manipulation (refer to "Protection against Physical Manipulation (O.Phys-Manipulation)". If this is not the case, signals which normally do not contain significant information about secrets could become an information channel for a leakage attack.	OT.Prot_Phys_tamper OT.Prot_Malfunction.
O.Abuse-Func	Protection against Abuse of Functionality	The TOE must prevent that functions of the TOE which may not be used after TOE Delivery can be abused in order to (i) disclose critical User Data, (ii) manipulate critical User Data of the Security IC Embedded Software, (iii) manipulate Soft-coded Security IC Embedded Software or (iv) bypass, deactivate, change or explore security features or security services of the TOE. Details depend, for instance, on the capabilities of the Test Features provided by the IC Dedicated Test Software which are not specified here.	No influence from the ES.
O.Identification	TOE Identification	The TOE must provide means to store Pre-Initialisation Data and Initialization Data in its non-volatile memory. The Pre-Initialisation Data (or parts of them) are used for TOE identification.	The ES does not access the chip identification data.
O.RND	Random Numbers	The TOE will ensure the cryptographic quality of random number generation. For instance random numbers shall not be predictable and shall have a sufficient entropy. The TOE will ensure that no information about the produced random numbers is available to an attacker since they might be used for instance to generate cryptographic keys.	OT.Cryptography OT.Cryptography include random generation
O.Add-Functions	Additional Specific Security Functionality	The TOE must provide the following specific security functionality to the smartcard Embedded Software : • Advanced Encryption Standard (AES) • Triple Data Encryption Standard (3DES) • Rivest-Shamir-Adleman (RSA) • Elliptic Curve Cryptography (EC) • Secure Hash Algorithm (SHA-2)	ES does not use these functionalities.
O.Mem-Accesss	Area based Memory Access Control	The TOE must provide the Smartcard Embedded Software with the capability to define restricted access memory areas. The TOE must then enforce the partitioning of such memory areas so that access of software to	Platform provides capability to define restricted access memory area

IC TOE security objective Label	IC TOE security objective Title	IC TOE security objective Content	Link to the composite-product
		memory areas and privilege levels is controlled as required, for example, in a multi-application environment.	But the ES does not use these functionalities.

Table 23 – Composition – Security objectives for the TOE part

8.6 SECURITY OBJECTIVES FOR THE ENVIRONMENT PART

IC ENV security objective Label	IC ENV security objective Title	IC ENV security objective Content	Link to the composite-product
OE.Plat-Appl	Usage of Hardware Platform	To ensure that the TOE is used in a secure manner the Security IC Embedded Software shall be designed so that the requirements from the following documents are met: – (i) hardware data sheet for the TOE, – (ii) data sheet of the IC Dedicated Software of the TOE, – (iii) TOE application notes, other guidance documents, and – (iv) findings of the TOE evaluation reports relevant for the Security IC Embedded Software as referenced in the certification report.	Fulfilled by the composite-SAR ADV_COMP.1 (cf [CCDB], Appendix 1.2, §72 and §73)
OE.Resp-Appl	Treatment of User Data	Security relevant User Data (especially cryptographic keys) are treated by the Security IC Embedded Software as required by the security needs of the specific application context. For example the Security IC Embedded Software will not disclose security relevant User Data to unauthorized users or processes when communicating with a terminal.	Platform provides cryptographic services not used by the ES.
OE.Process-Sec-IC	Protection during composite product manufacturing	Security procedures shall be used after TOE Delivery up to delivery to the end-consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorized use). This means that Phases after TOE Delivery up to the end of Phase 6 (refer to Section 1.2.3) must be protected appropriately.	There is no objective stated for the environment in the phases mentioned in OE.Process-Sec-IC.

Table 24 – Composition – Security objectives for the environment part

9. ABBREVIATIONS

Name	Definition
AC	Access Conditions
ADF	Application DFs
ALR	Anomaly List Report
APC	Subsystem “APDU Container”
APDU	Application Protocol Data Unit
API	Application Programming Interface
APL	Acceptance Plan
ARGOS	Acceptance and Requirements for GEMALTO Organization System
ATM	Automatic Teller Machine
ATR	Answer To Reset
BLK	Module “Block”
CAR	Card Acceptance Report
CC	Common Criteria (referenced as CC)
CEPS	Common Electronic Purse Specifications
CI	Configuration Item
CIS	Card Initialisation Specification
CLI	Command Line Interface
COS	Card Operating System
CM	Configuration Management
CMP	Configuration Mangement Plan
CMS	Configuration Management System
CSP	Certification-Service provider
CUD	Client User Document
CVC	Card Verifiable Certificate
DAR	DIL Acceptance Report
DESCRY	Module “DES-crypto”
DF	Dedicated File
DIL	Dual In Line
EAL	Evaluation Assurance Level
EC	Electronic Cash
EEPROM	Electrically Erasable and Programmable Read Only Memory
EF	Elementary File
eGK	elektronische Gesundheitskarte
eHC	electronic Health Card
EMV	Europay-Mastercard-Visa
ERR	Subsystem “Error Handling”
ES	Embedded Software
FRS	Functional Requirement Specifications
FS	Subsystem “File System”
HAL	Subsystem “Hardware Abstraction Layer”
HBCI	HomeBanking Computer Interface
HEC	Health Employee Card (technically a type of HPC)
HSH	Module “Hash”

HSM	Hardware Security Module
HPC	Health Professional Card
IC	Integrated circuit
ID	Identifier
IFD	Interface device
INS	Instruction code
I/O	Input/Output
IT	Information Technology
IUD	Internal User Documentation
LRC	Longitudinal Redundancy Checksum
MAC	Message Authentication Code
MAR	Mask Acceptance Report
MF	Master File
OS	Operating System
OSP	Operational Security Policy
*OSP.****	Naming convention for organisational security policies in this PP, e. g. OSP.User_Information
*OT.****	Naming convention for security objectives for the TOE in this PP, e. g. OT.Access_Rights
PIN	Personal Identification Number (authentication feature)
PKI	Public Key Infrastructure
PL	Project Leader
PP	Protection Profile
PROC	Subsystem “Process Handling”
PUC	PIN Unblocking Code
PVCS	Product Version Control System
RAD	Reference Authentication Data
RAM	Random Access Memory
ROM	Read Only Memory
SAR	Security assurance requirements
RSA	Rivest Shamir Adleman (algorithm)
SCM	Software Configuration Mangement
SCMA	Software Configuration Mangement Administrator
SCU	Smart Card Utility
SDD	Software Design Description
SDD1	Preliminary Software Design Description
SDD2	Detailed Software Design Description
SDO	Signed Data Object
SF	Security Function
SFP	Security Function Policy
SFP_access_rules	Name of the security functional policy defining the access rights to assets (data) in the TOE. It is defined in OT.Access_Rights and used by access control SFRs
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SMS	Software Masking Specification
SOF	Strength Of Function
SK	Subsystem “Security Kernel”
SM	Module “secure messaging”

SMC	Security Module Card
ST	Security Target
SVA	Software Validation Approval
TBX	Subsystem “Toolbox”
TDM	Technical Data Management
TOE	Target of Evaluation
TOE_App	Application Part of the TOE
TOE_ES	TOE Embedded Software (operating system of the TOE)
TOE_IC	The integrated circuit of the TOE, the hardware part together with IC dedicated software
TSC	TSF Scope of Control
TSF	TOE Security Functions
TSP	TOE Security Policy
UART	Universal Asynchronous Receiver Transmitter
UTP	Unitary Test Plan
UTR	Unitary Test Report
VAD	Verification authentication data
VCC	Voltage at the Common Collector
VLR	Validation Review
VTP	Validation Test Plan
VTP1	Preliminary Validation Test Plan
VTP2	Detailled Validation Test Plan
VTR	Validation Test Report
VTS	Validation Test Specification
X.509	A certificate format

Table 25 – Abbreviation table

10. GLOSSARY

The glossary elements for this development project are given in the table below:

Administrator means an user authorized to the TOE for personalisation, or other TOE administrative functions.
Archive. PVCS or VSS file which contains the evolution history of a work file. PVCS or VSS is able to rebuild any revision of the work file. Historical information includes description of changes, who made them, and when they were made. The archive also contains information about the status and attributes of the archive and its associated work file
Authentication data is information used to verify the claimed identity of a user.
Branch. Separate line of development consisting of one or more revisions that diverge from a revision on the trunk or from another development branch
Check-In. Action of storing a new revision in an archive.
Check-Out. Action of getting a revision from an archive. Then the archive is locked, and can be modified to do another revision.
Component. The hardware component of the Operating System.
Evolution Index (VSS). Symbolic reference used to uniquely identify a preliminary software version.
Evolution Index (PVCS). This number (integer) is used to uniquely identify a software version. Take note that the EI is different from the revision number that is automatically generated by PVCS.
Filter. A set of bug fixes and adjustments of the ROM code, residing in EEPROM
Folder (VSS/PVCS). A folder enables to organise archives in the Version Manager MMI. It logically links some archives
IC dedicated software. The part of the TOE's software, which is provided by the hardware manufacturer
IC Dedicated Support Software. That part of the IC Dedicated Software (refer to above) which provides functions after TOE Delivery. The usage of parts of the IC Dedicated Software might be restricted to certain phases.
IC Dedicated Test Software. That part of the IC Dedicated Software (refer to above) which is used to test the TOE before TOE Delivery but which does not provide any functionality thereafter.
Initialisation Data. Any data defined by the TOE Manufacturer and injected into the non-volatile memory by the Integrated Circuits manufacturer (Phase 2). These data are for instance used for traceability and for IC identification (IC identification data).
Integrated circuit (IC) Electronic component(s) designed to perform processing and/or memory functions. The eHC's chip is a integrated circuit.
Label. Symbolic name assigned to a revision in one or more archives. Labels provide a convenient way to refer to several archives with different revisions by a single name
Mask. Software developed by GEMALTO to be implemented in the chip

Module. Subset of commands and/or mechanisms. A module groups several routines allowing a logical function. A module cannot be broken up. Most of the time, a module will contain only one source file in the OS referential while it may involve several tests in the Test referential. [examples of modules for the Administrative Kernel brick are Record, Authentication, Secure Messaging, ...]
Mutual Authentication. Type of those cryptographic protocols, were two entities mutually verify the authenticity of each other, for smart cards this is realised by suitable sequences of amt card commands and responses
Personalisation. The process by which personal data are brought into the TOE before it is handed to the card holder
Product. Set of modules that constitute a final mask or a final filter (final release)
Project. See VSS/PVCS project
Reference authentication data (RAD) means data persistently stored by the TOE for verification of the authenti95uthorizedempt as authorised user.
Referential. Set of software components which are used by several Teams such as the OS software or the Test environment. The Referential contains all the archives of a project
Revision. Particular iteration of a work file in an archive. Each time a work file is modified and checked back into the archive, VSS/PVCS creates a new revision and assigns it automatically a new revision number
Rule_*. Naming convention for access control rules in this PP, defined in SFP_access_rules.
Secure Channel. A connection between two devices, which is secured against interception or modification of the transmitted data. The TOE realises a secure channel to other devices using secure messaging.
secure messaging in encrypted mode. Secure messaging using encryption and message authentication code according to ISO/IEC 7816-4
Service_**.** Services provided by the TOE (e. g. Service_Privacy)
Signature attributes means additional information that is signed together with the user message.
Sub-Referential. Consistent set of software components (Example: test scripts, specification documents,). A Sub-referential belongs to a Referential.
Tip Revision. The latest revision of a line of development (the trunk or a branch)
TSF data. Data created by and for the TOE, that might affect the operation of the TOE
User means any entity (human user or external IT entity) outside the TOE that interacts with the TOE.
User data. Data created by and for the user, that does not affect the operation of the TSF
Verification authentication data (VAD) means authentication data provided as input by knowledge or authentication data derived from user's biometric characteristics.
VSS/PVCS Project. Logical set of folders and archives
Work File. Copy of an archive revision, usually for working with it on a local PC. If the archive is "checked out" this copy can be modified and "checked in" again as the new revision of the archive.
Work File Directory. Local folder to hold the archive copies generated by "Check Out" or "Get" actions (in German: "Auscheckordner"). A folder in VSS must be linked to a work file directory, so that "Get" actions can be performed.

Table 26 – Glossary table

11. REFERENCES

The documents and reference elements for this development project are given in the table below:

Reference	Title of document	Author
Common Criteria Documents		
CCPART1	Common Criteria for Information Technology Security Evaluation. Part 1: Introduction & general model, CCMB-2009-07-001. Version 3.1. July 2009.	Common Criteria Project Sponsoring Organizations
CCPART2	Common Criteria for Information Technology Security Evaluation. Part 2: Security Functional requirements, CCMB-2009-07-002. Version 3.1. July 2009.	Common Criteria Project Sponsoring Organizations
CCPART3	Common Criteria for Information Technology Security Evaluation. Part 3: Security Assurance requirements, CCMB-2009-07-003. Version 3.1. August 2009.	Common Criteria Project Sponsoring Organizations
CEM	Common Methodology for Information Technology Security Evaluation CCIMB-2009-07-004, version 3.1 Release 3, July 2009.	Common Criteria Project Sponsoring Organizations
JIL	Application of attack potential to smartcards, version 2.7 february 2009	Joint Interpretation Library
AIS 34	Application Notes and Interpretation of the Scheme (AIS), AIS 34, Evaluation Methodology for CC Assurance Classes for EAL5+, Version 3., 03.09.2009, Bundesamt für Sicherheit in der Informationstechnik.	BSI
AIS 36	Composite product evaluation for Smart Cards and similar devices, Version 1, Rev 1, September 2007, CCDB-2007-09-001	Common Criteria
AAPSC	Application of Attack Potential to Smartcards, Version 2.7, February 2009	Common Criteria Project Sponsoring Organizations
AMSRP	Attack Methods for Smartcards and Similar Devices, Version 1.5, February 2009	Common Criteria Project Sponsoring Organizations
ETR_Lite Annex A	ETR-lite for composition: Annex A Composite smartcard evaluation : Recommended best practice, Version 1.2, March 2002	Common Criteria Project Sponsoring Organizations
PP eHC	The Protection profile - “Electronic Health Card (eHC)” rev 2.9, 19/04/2011	BSI
GeGKOS_PERS	GeGKOS_A6: Specification for Chip card Personalization	Gemalto
EMV-CPS	EMV card personalisation specification, Version 1.0, June 2003.	EMV
Chip Documents		
ST IC	Security Target M7801 A12 Including optional software libraries RSA – EC – SHA 2 – Version 0.8 - 2010-07-26	Infineon technologies AG
CER IC	Certification report for Infineon technologies smart card IC	BSI

	(security controller) M7801 A12 with optional RSA2048/4096 V1.1.18, EC V1.1.18 and SHA 2 V1.1 libraries and with specific IC dedicated software from Infineon - 10/06/2010 And BSI-DSZ-CC-0606-2010 Confirmation of the reassessment 17 May 2011	
DB IC	SLx 70 family Hardware Reference Manual, Nov 2010	Infineon
eHC Documents		
eHC spec part 1	Die Spezifikation des elektronischen Gesundheitskarte Teil 1: Spezifikation der elektrishen Schnittstelle, Version 2.2.0, 20.03.2008 supplemented by SRQ 1070, 1069, 1067, 1066, 1065, 1064, 1047, 0959, 0842, 0841, 0840, 0838, 0837, 0836, 0835, 0834, 0833, 0832, 0831, 0829, 0828, 0827, 0826, 0825, 0824, 0823, 0822, 0821, 0820, 0819, 0818, 0817, 0816, 0815, 0814, 0810, 0809, 1154, 1153, 1094	GEMATIK
eHC spec part 2	Die Spezifikation des elektronischen Gesundheitskarte Teil 2: Grundlegende Applikationen, Version 2.2.0, 25.03.2008 supplemented by SRQ 1030, 950, 949, 948, 947, 946, 945, 944, 890, 889, 888, 887, 886, 885, 884, 883, 882, 881, 1085	GEMATIK
SICCT	SICCT (28.2.2006): TeleTrusT, SICCT Secure Interoperable ChipCard Terminal, Version 1.0.0	
Reference	**Title of document**	**Author**
ISO Documents		
ISO C1	ISO 7816 – 3, Identification cards - Integrated circuit(s) cards with contacts, Part 1: Physical characteristics. 1997	
ISO C3	ISO 7816 - 3, Identification cards - Integrated circuit(s) cards with contacts, Part 3: Electronic signals and transmission protocols. 1997	ISO
ISO C4	ISO 7816 - 4, Identification cards - Integrated circuit(s) cards with contacts, Part 4: Inter-industry commands for interchange. 1995	ISO
ISO C4'	ISO 7816 - 4, Identification cards - Integrated circuit(s) cards with contacts, Part 4: Inter-industry commands for interchange, AMENDMENT 1: Impact of secure messaging on the structures of APDU messages. 1996	ISO
ISO C8	ISO 7816 - 8, Identification cards - Integrated circuit(s) cards with contacts, Part 8: Security related inter-industry commands. 1997	ISO
ISO C9	ISO 7816 - 9, Identification cards - Integrated circuit(s) cards with contacts	ISO
ISO HF3	ISO 10118 - 3, Information technology - Security techniques - Hash-functions, Part 3: Dedicated hash functions, 1998	ISO
RSA Laboratories Documents		

PKCS1	RSA Encryption Standard . Version 2.1 June 14, 2002	RSA Laboratories
Nist Document		
FIPS	Federal Information Processing Standards Publication 46-3 (FIPS PUB 46-3) of U.S. DEPARTMENT OF COMMERCE/National Institute of Standards and Technology Data encryption standard (DES) – Reaffirmed 1999 October 25	NIST
Hash document		
SHA	FIPS 180-3 Secure Hash Standard (SHS) - October 2008	NIST
Random generators		
AIS31	Functionality classes and evaluation methodology for true (physical) random number generators. Version 3.1, September 25, 2001. http://www.bsi.de/zertifiz/zert/interpr/trngk31e.pdf	
Algo document		
ALGO	Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung (Übersicht über geeignete Algorithmen) Vom 20. May 2011 -	Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen

Table 27 – Reference table

<END OF DOCUMENT>