Federal Office
for Information Security

# Common Criteria Protection Profile

# Electronic Health Card Terminal (eHCT)

Common Criteria

BSI-CC-PP-0032

Approved by the
Federal Ministry of Health

**Foreword**

This ‘Protection Profile - Protection Profile – electronic Health Card Terminal - is issued by Bundesamt für Sicherheit in der Informationstechnik, Germany.

The document has been prepared as a Protection Profile (PP) following the rules and formats of Common Criteria version 2.3 [1], [2], [3].

Correspondence and comments to this Protection Profile should be referred to:

CONTACT ADDRESS

Bundesamt für Sicherheit in der Informationstechnik
Godesberger Allee 185-189
D-53175 Bonn, Germany

Tel.: +49 228 99 9582-0
Email bsi@bsi.bund.de

Internet: http://www.bsi.bund.de

**Change history**

| Version | Date | Reason | Remarks |
|---|---|---|---|
| 1.73 | 29th November 2007 | Final Version | Final Version |

Last Version: (1.73 29th November 2007)

| Name | Value | Display |
|---|---|---|
| File name and sizes | Set automatically | ehc_kl_PP_1.73 (702464 Byte) |
| Last Version | 1.73 | 1.73 |
| Date | 29th November 2007 | 29th November 2007 |
| Classification | Unclassified | Unclassified |
| Authors | Nils Tekampe | Nils Tekampe |

This page is intentionally left blank.

**Table of Content**

# 1 PP Introduction

## 1.1 PP reference

| | |
|---|---|
| Title: | Protection Profile – Electronic Health Card Terminal |
| Sponsor: | Bundesamt für Sicherheit in der Informationstechnik (BSI) |
| Editor(s): | Nils Tekampe, TÜV Informationstechnik GmbH |
| CC Version: | 2.3 |
| Assurance Level: | EAL 3 augmented by ADO_DEL.2, ADV_LLD.1, ADV_IMP.1, ADV_SPM.1, ALC_TAT.1, AVA_MSU.3 and AVA_VLA.4. |
| General Status: | draft |
| Version Number: | 1.73 |
| Date: | 29th November 2007 |
| Registration: | BSI-CC-PP-0032 |
| Keywords: | Protection Profile, Electronic health card terminal |

## 1.2 PP Overview

This protection profile defines the security objectives and requirements for the **Electronic Health Card Terminal** based on the regulations for the German healthcare system. It addresses the security services provided by this terminal, mainly:

- The access to one or more slots for smart cards,
- Secure network connectivity,
- Secure PIN entry functionality,
- Encryption of communication,
- User authentication,
- Management of network settings and
- Update of Firmware

## 1.3 Conformance Claim

This PP is claimed to be conformant to part II and III of Common Criteria ([2], [3]). This PP does not claim conformance to any other PP. The CC version in use is: ISO/IEC 15408: Common Criteria, Version 2.3, August 2005.

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

The assurance level is EAL3 augmented by ADO_DEL.2, ADV_LLD.1, ADV_IMP.1, ADV_SPM.1, ALC_TAT.1, AVA_MSU.3 and AVA_VLA.4

# 2 TOE Description

The TOE described in this Protection Profile is a smart card terminal which fulfils the requirements to be used with the German electronic Health Card (eHC) and the German Health Professional Card (HPC) based on the regulations of the German healthcare system.

This terminal bases on the specification for a “Secure Interoperable ChipCard terminal” ([11]) extended and limited by the specifications for the e-health terminal itself ([10]).

In its core functionality the TOE is not different from any other smart card terminal which provides an interface to one or more smart cards including a mean to securely enter a PIN. Additionally the TOE provides a network interface which allows routing the communication of a smart card to a remote IT product outside the TOE.

The TOE provides the following main functions:

- Access to one or more slots for smart cards,
- Secure network connectivity,
- Secure PIN entry functionality,
- Enforcement of the encryption of communication,
- User authentication,
- Management of network settings and
- Update of Firmware

Two different architectures of such a terminal are in principle possible:

- A stand alone smart card terminal can be directly or indirectly attached to a LAN. In this case all specifications need to be fulfilled by this terminal and the terminal has to provide all the Security Features as required by this PP.
- A virtual card terminal is built out of a combination of a smart card terminal without network interface or a smart card terminal which does not support the complete interface specification and a piece of software which runs on a different machine and provides the missing functionality (The additional software and the machine it runs on are also referred to as Proxy). In this configuration the combination of
  - The smart card reader,
  - The supporting software and
  - The execution environment for the software

  builds the virtual card terminal and thus the TOE as described in this PP.

Figure 1 shows these two possible architectures of the TOE including the TOE boundaries and their immediate environment.

In its environment the TOE communicates with a so called Connector. This Connector is the secure connection between the local network of the medical supplier and the remote network of the telematic infrastructure. It provides the medical supplier with secure access to the services of the telematic infrastructure. The Connector is the only entity in the environment of the TOE which is planned to communicate with it.

To protect the communication between the Connector and the TOE the TOE has to possess a cryptographic identity (in form of a X.509 certificate) and functionality for encryption/decryption as well as signature creation based on RSA (see also [10]).

For its cryptographic functionality the TOE relies on the services of the so called SM-KT[1].

The SM-KT (Secure Module Kartenterminal) is a secure module that represents the cryptographic identity of the TOE in form of a X.509 certificate.

This module - in form of an ID-000 smart card - provides:

- Protection of the private key,
- Cryptographic functions based on RSA for encryption/decryption and signature creation,
- A random number generator and
- A function to read out the public key

Though this SM-KT will usually be physically within the cage of the TOE it does not belong to the logical and physical scope of the TOE as to see in the following figure. More information about the SM-KT can be found in the corresponding Protection Profile.

## 2.1 Physical scope of the TOE

In case of a stand-alone card terminal the physical scope of the TOE comprises

- The hardware and cage of the smart card terminal,
- The firmware of the smart card terminal and
- The related guidance documents

In case of a virtual card terminal the scope additionally includes the relevant parts of the proxy (hardware and software).

Please note that though – depending on a concrete realization – the SM-KT may be physically within the cage of the terminal this module does not belong into the scope of the TOE as described in this PP.

## 2.2 Logical scope of the TOE

The logical scope of the TOE is represented by its core security features:

- Access to one or more slots for smart cards,
- Secure network connectivity,
- Secure PIN entry functionality,
- Enforcement of the encryption of communication,
- User authentication,
- Management of network settings and
- Update of Firmware

And is limited by the functionality for which the TOE relies on the services of the SM-KT.

[1] Please note that the SM-KT is only responsible for the core functions of the asymmetric cryptography based on RSA. The TOE will be responsible for negotiating the session with the Connector and for encryption/decryption using a symmetric AES key. More details can be found in [10] and the following chapters.

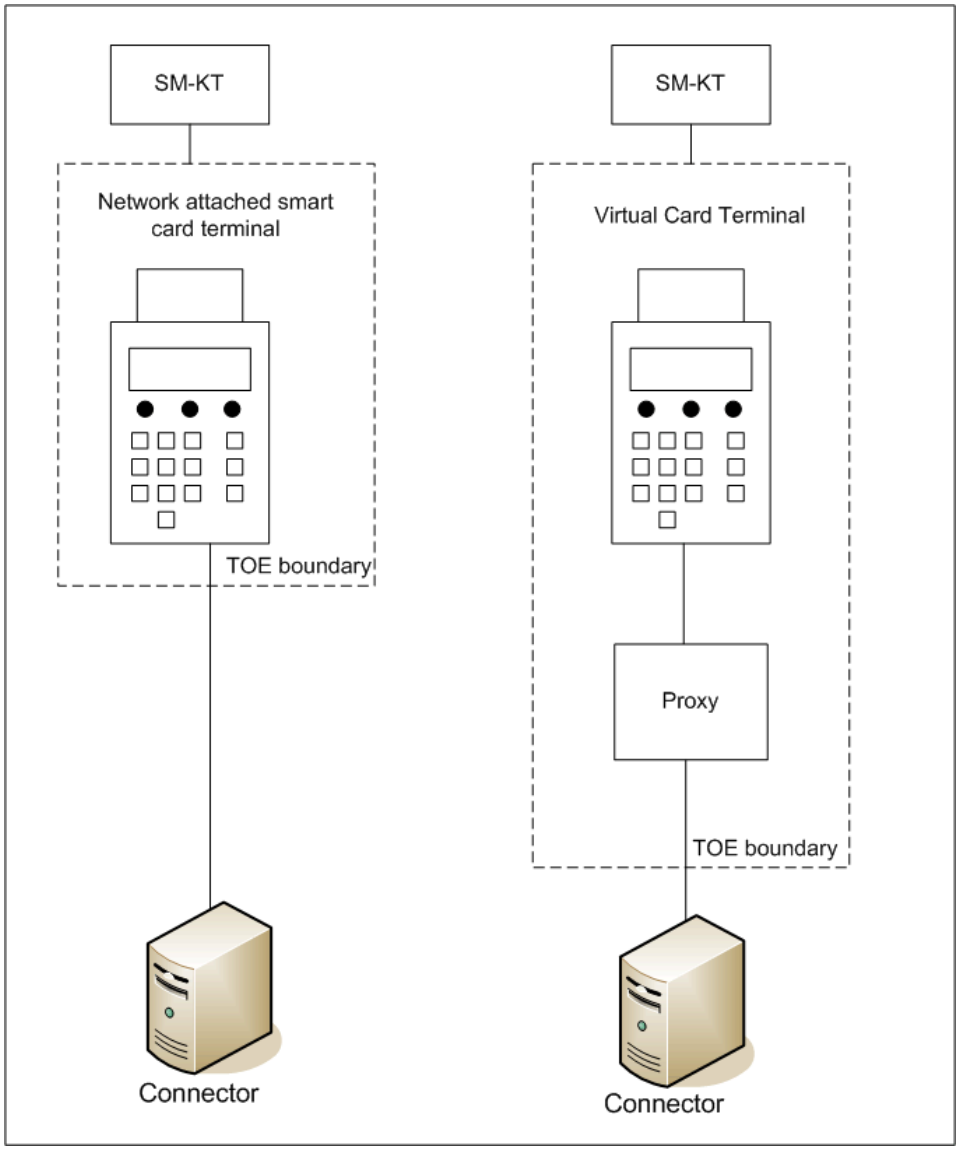


**Figure 1: TOE architecture (logical perspective)**

According to [10] compliance with this PP does only represent a part of the registration process for an Electronic Health Card Terminal. Additionally [10] requires:

- That the terminal has to be compliant to the requirements in [10] and [11] and
- That the terminal has to undergo a registration process of the gematik.

It should be mentioned that according to [10] it would be allowed that a terminal, claiming compliance to this PP, implements more functionality than defined in this PP and that a terminal temporarily operates in an insecure state. In such a state parts of the security functionality as required by this PP may not be available. However for these cases the terminal has to indicate to the user, whether it is currently working in a secure state or not.

It should further be mentioned that the TOE as described in this Protection Profile does not cover the functionality of batch signatures and remote PIN entry. Please refer to chapter 7 for more details.

# 3 TOE Security Environment

This chapter introduces the security environment of the TOE. This comprises:

- The **assets** which have to be protected by the TOE.
- The **subjects** which are interacting with the TOE.
- The **assumptions** which have to be made about the environment of the TOE.
- The **threats** which exist against the assets of the TOE
- The **organisational security policies** the TOE has to comply to.

## 3.1 Assets

The following assets need to be protected by the TOE and its environment:

| Asset | Description |
|---|---|
| PIN | The TOE interacts with the user to acquire a PIN and sends this PIN to one of the cards in a slot of the TOE. The TOE has to ensure the confidentiality of the PIN. |
| User Data | The TOE gets data from the cards in its slots, encrypts this data and sends it to the Connector. Further the TOE accepts data from the Connector back, decrypts it and sends it to the corresponding card in its slot. The TOE has to ensure the confidentiality and authenticity of this data. |
| TSF Data | The TOE stores TSF data which is necessary for its own operation. |

**Table 1: Assets**

## 3.2 Subjects

The following subjects are interacting with the TOE:

| Subject | Description |
|---|---|
| User | A user is communicating with the TOE in order to use its primary services, i.e. to access a smart card which has been put into one of the slots of the TOE before. The TOE is used by different kinds of users including medical suppliers, patients and administrators. |
| Administrator | The administrator is in charge of managing the security functions of the TOE. |
| Patient | The patient uses the TOE together with his EHC. The patient is not able to generate qualified digital signatures with the EHC but uses the TOE for other services of the EHC. A patient will never use the services of the TOE alone but will always be guided by the medical supplier. |
| Medical supplier | The medical supplier (e.g. a physician) uses the TOE together with his HPC and is able to generate qualified digital signatures. Other than the patient the medical supplier can be held responsible for the secure operation of the TOE. |
| Attacker | A human, or a process acting on his behalf, located outside the TOE. The main goal of the attacker is to access or modify application sensitive information. The attacker has a high level attack potential. |
| Connector | The Connector is the only entity in the environment of the TOE which is foreseen to communicate with the TOE. It is the interface for the TOE to communicate with the telematic infrastructure of the German healthcare system. |
| Card | The TOE is handling the communication for one or more smart cards in its card slots. |
| SM-KT | The SM-KT represents the cryptographic identity of the TOE. It is a secure module that carries a X509 certificate and provides :<br>• Protection of the private key<br>• Cryptographic functions based on RSA for encryption/decryption and signature creation<br>• A random number generator<br>• A function to read out the public key |

**Table 2: Subjects**

## 3.3 Assumptions

The following assumptions need to be made about the environment of the TOE to allow the secure operation of the TOE.

| Assumption | Description |
|---|---|
| A.ENV | It is assumed that the TOE is used in a controlled environment.<br>Specifically it is assumed:<br>• That the user handles their PIN with care; specifically that the user will keep their PIN secret,<br>• That the user can enter the PIN in a way that nobody else can read it,<br>• That no unauthorized access to the TOE is possible in a way that would allow an attacker to manipulate the terminal without a medical supplier detecting this modification.<br>More information about the characteristics of such a controlled environment can be found in [10]. |
| A.ADMIN | The administrator of the TOE and the medical supplier are non hostile, well trained and know the existing guidance documentation of the TOE. The administrator and the medical supplier are responsible for the secure operation of the TOE. |
| A.CONNECTOR | The Connector in the environment is assumed to be trustworthy and provides the possibility to establish a Trusted Channel with the TOE including a mean for a mutual authentication. It is assumed that the Connector has undergone an evaluation and certification process in compliance with the corresponding Protection Profiles. |
| A.SM | The TOE will use a secure module (SM-KT) that represents the cryptographic identity of the TOE in form of a X.509 certificate.<br>It is assumed that the cryptographic keys in this module are of sufficient quality and the process of key generation and certificate generation is appropriately secured to ensure the confidentiality, authenticity and integrity of the private key and the authenticity and integrity of the public key/certificate.<br>It is further assumed that the secure module is secured in a way that protects the communication between the TOE and the module from eavesdropping and manipulation and that the SM-KT is securely connected with the TOE.<br>It is assumed that the secure module has undergone an evaluation and certification process in compliance with the corresponding Protection Profile. |

**Table 3: Assumptions**

## 3.4 Threats

This chapter describes the threats that have to be countered by the TOE.

The attack potential of the attacker behind those threats is in general characterized in terms of their motivation, expertise and the available resources.

As the TOE handles and stores information with a very high need for protection with respect to their authenticity, integrity and confidentiality it has to be assumed that an attacker will have a high motivation for their attacks.

Further it has to be assumed that an attacker has deep knowledge over the internals of the TOE and nearly unlimited resources to perform their attacks. In this way the possibilities for an attacker are only limited by the characteristics of the environment (specifically addressed by A.ENV).

Summarizing this means that an attacker with a high attack potential has to be assumed.

The assets that are threatened and the path for each threat are defined in the following table.

| **Threat** | **Description** |
|---|---|
| T.COM | An attacker may try to intercept the communication between the TOE and the Connector in order to gain knowledge about secret data which is transmitted between the TOE and the Connector or in order to manipulate this communication. As part of this threat an authorized user, who is communicating with the TOE (via a Connector) could try to influence communications of other users with the TOE in order to manipulate this communication or to gain knowledge about secret data. |
| T.PIN | An attacker may try to release the PIN which has been entered by a user from the TOE. As part of this attack the attacker may try to route a PIN, which has been entered by a user, to a wrong card slot. |
| T.DATA | An attacker may try to release or modify protected data from the TOE.<br>This data may comprise:<br>- Configuration data the TOE relies on for its secure operation<br>- User data (including medical data) that is received from a card and stored within the terminal before it is submitted to the Connector<br>The attack path for this threat cannot be limited to any specific scenario but includes any scenario that is possible in the assumed environment of the TOE.<br>Specifically an attacker may<br>- use any interface that is provided by the TOE<br>- physically probe or manipulate the TOE |

**Table 4: Threats**

## 3.5 Organisational Security Policies

The TOE shall be implemented according to the following specifications:

| Policy | Description |
|---|---|
| OSP.SIGG | The TOE shall fulfill the requirements to be used as a secure PIN pad entry device for applications according to [5].<br>This specifically means that a PIN, which has been entered by a user at the TOE must never leave the TOE in clear text.<br>For the case that a terminal implements an insecure mode (e.g. a mode, in which it cannot be guaranteed that the PIN will no leave the TOE or a mode in which not trustworthy entities are allowed to communicate with the TOE) the TOE has to be able to inform the medical supplier whether it is currently in a secure state or not. |

**Table 5: Organisational Security Policies**

# 4 Security Objectives

This chapter describes the security objectives for the TOE (in chapter 4.1) and the security objectives for the environment of the TOE (in chapter 4.2).

## 4.1 Security Objectives for the TOE

The following security objectives have to be met by the TOE

| **Objective** | **Description** |
|---|---|
| O.ACCESS_CONTROL | To protect the configuration of the TOE against unauthorized modifications only an authorized user shall be able to read out information about the current configuration of the TOE and only the administrator shall be able to modify the settings of the TOE.<br>Therefore the TOE shall provide an access control function based on the identity of the current user.<br>Further the access control mechanism of the TOE has to ensure that the PIN cannot be read from the TOE. |
| O.PIN_ENTRY | The TOE shall serve as a secure pin entry device for the user and the administrator.<br>Thus the TOE has to provide the user and administrator with the functionality to enter a PIN and ensure that the PIN is never released from the TOE in clear text. Further the TOE shall inform the user to which card slot the PIN will be sent. |
| O.I&A | For its access control policy and for parts of the management functionality the TOE has to be aware of the identity of the current user.<br>Thus the TOE has to provide a mean to identify and authenticate the current user. The TOE shall maintain at least two distinct roles for administrators and users[2].<br>When establishing a connection between the TOE and the Connector both parties may have to be aware of the identity of their communication partner. Thus the TOE has to provide a mean to authenticate the Connector and to authenticate itself against the Connector in accordance with [10]. |

[2] It should be noted that the scope of the identification and authentication of the user is only to determine the role the current user belongs to. According to [10] there is no requirement to maintain the ID of the user.

| Objective | Description |
|---|---|
| O.MANAGEMENT | In order to protect its configuration the TOE shall provide only an authorized administrator with the necessary management functions.<br>An update of the firmware of the TOE shall only be possible<br>• If the version of the firmware to install is higher than the version of the current firmware and<br>• After the integrity and authenticity of the firmware has been verified<br>The TOE shall ensure that for all security attributes, which can be changed by an administrator or the user, only secure values are accepted. |
| O.SECURE_CHANNEL | For all communications which fall into the context of the electronic health card application the TOE shall only accept communication via this secure channel to ensure the integrity, authenticity and confidentiality of the transmitted data.<br>Only functions to identify the TOE in the network (service discovery) may be available without a secure channel. |
| O.STATE | In principle it would be possible that a card terminal compliant to this Protection Profile realizes more than just the necessary set of functionality as required by this PP.<br>However such additional functionality may lead to an insecure state of the TOE as the medical supplier may be not aware of the fact that they are using a functionality, which doesn't fall into the scope of the certified TOE or because a part of the security functionality as required by this PP is not working.<br>Thus the TOE shall be able to indicate whether it is currently in a secure state, i.e. whether all TSP as required by this PP are actually enforced. |

| Objective | Description |
|---|---|
| O.PROTECTION | The TOE shall be able to verify the correct operation of the TSF. To ensure the correct operation of the TSF the TOE shall verify the correct operation of all security functions at startup and specifically verify the correct operation of the secure module (see A.SM).<br>The TOE shall provide an adequate level of physical protection to protect the stored assets and the SM-KT[3]. It has to be ensured that any kind of physical tampering that might compromise the TSP can be detected by the medical supplier.<br>To avoid interference the TOE has to ensure that each connection is held in its own security context where more than one connection of a TOE to a Connector is established.<br>Also if more than one smart card in the slots of the TOE is in use the TOE has to ensure that each connection is held in its own security context.<br>The TOE shall delete secret data in a secure way when it is not longer used.<br>For the case that a TOE comprises physically separated parts, the TOE shall prevent the disclosure and modification of data when it is transmitted between physically separated parts of the TOE. |

**Table 6: Security Objectives for the TOE**

[3] Please note that the SM-KT provides its own physical protection for the stored keys. However according to [10] it has to be ensured that the SM-KT is securely connected with the TOE. Thus the physical protection provided by the TOE has to cover the SM-KT.

## 4.2 Security Objectives for the Environment

The following security objectives have to be met by the environment of the TOE.

| Objective | Description |
|---|---|
| OE.ENV | The TOE shall only be used in a controlled environment.<br>Specifically it has to be ensured:<br>• That the user handles their PIN with care; specifically that the user will keep their PIN secret,<br>• That the user can enter the PIN in a way that nobody else can read it<br>• That no unauthorized access to the TOE is possible in a way that would allow an attacker to manipulate the terminal or that a medical supplier would detect this modification.<br>More information about the characteristics of such a controlled environment can be found in [10]. |
| OE.ADMIN | The administrator of the TOE and the medical supplier shall be non hostile, well trained and have to know the existing guidance documentation of the TOE.<br>The administrator and the medical supplier shall be responsible for the secure operation of the TOE. |
| OE.CONNECTOR | The Connector in the environment has to be trustworthy and provides the possibility to establish a Trusted Channel with the TOE including a mean for mutual authentication. The Connector has to undergo an evaluation and certification process in compliance with the corresponding Protection Profiles. |
| OE.SM | The TOE shall use a secure module (SM-KT) that represents the cryptographic identity of the TOE in form of a X.509 certificate.<br>The cryptographic keys in this module shall be of sufficient quality and the process of key generation and certificate generation shall appropriately secured to ensure the confidentiality, authenticity and integrity of the private key and the authenticity and integrity of the public key/certificate.<br>Further the secure module shall be secured in a way that protects the communication between the TOE and the module from eavesdropping and manipulation and the SM-KT shall be securely connected with the TOE.<br>The secure module has to be certified in compliance with the corresponding Protection Profiles. |

**Table 7: Security Objectives for the environment of the TOE**

# 5 Security Requirements

This chapter defines the functional requirements and the security assurance requirements for the TOE and its environment.

Operations for assignment, selection, refinement and iteration have been made. Operations not performed in this PP are identified in order to enable instantiation of the PP to a Security Target (ST).

All operations which have been performed from the original text of [2] are written in italics for assignments, underlined for selections and bold text for refinements. Furthermore the [brackets] from [2] are kept in the text.

All operations which have to be completed by the ST author are marked with the words: "assignment" or "selection" respectively.

## 5.1 Security Functional Requirements for the TOE

The TOE has to satisfy the SFRs delineated in the following table. The rest of this chapter contains a description of each component and any related dependencies.

| **Cryptographic Support (FCS)** | |
|---|---|
| FCS_CKM.1 | Cryptographic key generation |
| FCS_CKM.4 | Cryptographic key destruction |
| FCS_COP.1/TLS | Cryptographic operation for TLS |
| FCS_COP.1/SIG_VER | Cryptographic operation for signature verification |
| **User data protection (FDP)** | |
| FDP_ACC.1 | Subset access control |
| FDP_ACF.1 | Security attribute based access control |
| FDP_IFC.1/PIN | Subset information flow control for PIN |
| FDP_IFF.1/PIN | Simple security attributes for PIN |
| FDP_IFC.1/NET | Subset information flow control for network connections |
| FDP_IFF.1/NET | Simple security attributes for network connections |
| FDP_RIP.1 | Subset residual information protection |
| **Identification and Authentication (FIA)** | |
| FIA_ATD.1 | User attribute definition |
| FIA_UAU.1 | Timing of authentication |
| FIA_UAU.5 | Multiple authentication mechanisms |
| FIA_UID.1 | Timing of identification |

| **Security Management (FMT)** | |
|---|---|
| FMT_MOF.1 | Management of security functions behaviour |
| FMT_MSA.1 | Management of security attributes for terminal SFP |
| FMT_MSA.2 | Secure security attributes |
| FMT_MSA.3 | Static attribute initialisation for terminal SFP |
| FMT_SMF.1 | Specification of Management Functions |
| FMT_SMR.1 | Security roles |
| **Protection of the TSF (FPT)** | |
| FPT_AMT.1 | Abstract machine testing |
| FPT_FLS.1 | Failure with preservation of secure state |
| FPT_ITT.1 | Basic internal TSF data transfer protection |
| FPT_PHP.1 | Passive detection of physical attack |
| FPT_SEP.1 | TSF domain separation |
| FPT_TST.1 | TSF testing |
| **TOE Access (FTA)** | |
| FTA_TAB.1/PIN | Default TOE access banners for PIN |
| FTA_TAB.1/SEC_STATE | Default TOE access banners for secure state |
| **Trusted path/channels (FTP)** | |
| FTP_ITC.1 | Inter-TSF trusted channel |

**Table 8: Security Functional Requirements for the TOE**

### 5.1.1 Cryptographic Support (FCS)

#### 5.1.1.1 FCS_CKM.1 Cryptographic key generation

**FCS_CKM.1.1** The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment*: cryptographic key generation algorithm* ***for AES****]* and specified cryptographic key sizes [*128 or 256 bit*] that meet the following: [*[8] and [10]*].

Hierarchical to: No other components.

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

**Application Note**: The cryptographic session keys, generated by FCS_CKM.1 shall be used for the TLS encryption/decryption between the TOE and the Connector (see also chapter 5.1.1.3). The generation (actually negotiation) of this key shall be done in accordance with the TLS handshake protocol (see [8]), extended and limited by [10].

It should be noted that this negotiation may include a mutual authentication of the TOE and the Connector. Depending on the concrete realization of the mutual authentication this negotiation may require functionality for hashing, Random Number generation, Signature generation and Signature Verification, which shall be defined by the ST author if necessary.

As some of this additional functionality may be provided by the -KT (see also A.SM) it has not been possible to decide in the level of this Protection Profile whether this functionality is to be provided by the TOE or its environment (in form of the SM-KT).

#### 5.1.1.2 FCS_CKM.4 Cryptographic key destruction

**FCS_CKM.4.1** The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: *cryptographic key destruction method*] that meets the following: [assignment: *list of standards*].

Hierarchical to: No other components.

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

#### 5.1.1.3 FCS_COP.1/TLS Cryptographic operation for TLS

**FCS_COP.1.1/TLS** The TSF shall perform [*TLS encryption/decryption*] in accordance with a specified cryptographic algorithm [*AES*] and cryptographic key sizes [*128 or 256 bit*] that meet the following: [*[7]and [8]*].

Hierarchical to: No other components.

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

**Application Note**: The cryptographic functionality in FCS_COP.1/TLS and FCS_CKM.1 shall be used to establish the trusted channel with a Connector (see also chapter 5.1.7.1 for the definition of the trusted channel itself).

#### 5.1.1.4 FCS_COP.1/SIG_VER Cryptographic operation for signature verification

**FCS_COP.1.1/SIG_VER** The TSF shall perform [*signature verification*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*]] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [*[6]*].

Hierarchical to: No other components.

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

**Application Note**: The functionality for signature verification is used to check the integrity and authenticity of a potential firmware update. Such functionality usually relies on hashing and encryption using a public key. It is possible that the TOE uses the services of the SM-KT for this encryption and hashing functionality. For a TOE that implements this functionality itself the ST author should consider to add the corresponding SFRs to the ST.

### 5.1.2 User data protection (FDP)

#### 5.1.2.1 FDP_ACC.1 Subset access control

**FDP_ACC.1** The TSF shall enforce the [*terminal SFP*] on [
*Subjects: all subjects*
*Objects: PIN, firmware, cryptographic keys,* [assignment: *other objects*]
*Operations: Read, modify,* [assignment: *other operations*]].

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

#### 5.1.2.2 FDP_ACF.1 Security attribute based access control

**FDP_ACF.1.1** The TSF shall enforce the [*terminal SFP*] to objects based on the following: [
*Subjects: all subjects, attribute: user identity resp. group membership*
*Objects: PIN, firmware, cryptographic keys, attribute: object type (PIN, firmware, confidential key), firmware version,*
[assignment: *other objects and related attributes*]].

**FDP_ACF.1.2** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [
*A modification of the firmware[4] of the TOE must only be allowed:*

- *If the version of the firmware to install is higher than the version of the current firmware*
- *After the integrity and authenticity of the firmware has been verified using the mechanism as described in FCS_COP.1/SIG_VER*

[assignment: *other rules or none*]].

**FDP_ACF.1.3** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*].

**FDP_ACF.1.4** The TSF shall explicitly deny access of subjects to objects based on the [*No subject must read out the PIN or secret cryptographic keys while they are temporarily stored in the TOE*].

[4] In the case of a Virutal Card terminal the term “firmware” includes the Supporting Software as described in chapter 2.

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialisation

**Application Note:** Specific implementations of a TOE compliant to this PP may require more objects that are subject to Access Control and more granular rules for Access Control. Therefore the open assignment in FDP_ACF.1.2 should allow the ST author to specify the Access Control Policy for the TOE in more detail.

#### 5.1.2.3 FDP_IFC.1/PIN Subset information flow control for PIN

**FDP_IFC.1.1/PIN** The TSF shall enforce the [*PIN SFP*] on [
*Subjects: user, card*
*Information: PIN*
*Operation: Entering the PIN*].

Hierarchical to: No other components.

Dependencies: FDP_IFF.1 Simple security attributes

#### 5.1.2.4 FDP_IFF.1/PIN Simple security attributes for PIN

**FDP_IFF.1.1/PIN** The TSF shall enforce the [*PIN SFP*] based on the following types of subject and information security attributes: [
*Subject attribute: slot*[5] , [assignment: *other attributes*]].

**FDP_IFF.1.2/PIN** The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [*The PIN entered by the user shall only be sent to the card in the slot as indicated by the display of the TOE*].

**FDP_IFF.1.3/PIN** The TSF shall enforce the [assignment: *additional information flow control SFP rules*].

**FDP_IFF.1.4/PIN** The TSF shall provide the following [assignment: *list of additional SFP capabilities*].

**FDP_IFF.1.5/PIN** The TSF shall explicitly authorise an information flow based on the following rules: [*none*].

[5] This is the slot the user plugged his smart card in

**FDP_IFF.1.6/PIN** The TSF shall explicitly deny an information flow based on the following rules: [*none*].

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control
FMT_MSA.3 Static attribute initialisation

**Application Note**: Please note that the term "display" in this and other SFR refers to a generic display device and does not require any specific realization. Specifically this term does not require any display based on text or graphics but could e.g. also be realized as a simple LED as long as the requirements are fulfilled. However [10] may specify more detailed requirements about the display device.

#### 5.1.2.5 FDP_IFC.1/NET Subset information flow control for network connections

**FDP_IFC.1.1/NET** The TSF shall enforce the [*NET SFP*] on [
*Subjects: Connector, the TOE,*
*Information: all information arriving at the network interface*
*Operation: accept the communication*].

Hierarchical to: No other components.

Dependencies: FDP_IFF.1 Simple security attributes

#### 5.1.2.6 FDP_IFF.1/NET Simple security attributes for network connections

**FDP_IFF.1.1/NET** The TSF shall enforce the [*NET SFP*] based on the following types of subject and information security attributes: [
*Subject: Connector*
*Information: any*
*Information attribute: sent via the trusted channel,* [assignment: *other attributes*]].

**FDP_IFF.1.2/NET** The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [*Any information arriving at the network interface must only be accepted if the communication path is encrypted and the Connector has been successfully authenticated*].

**FDP_IFF.1.3/NET** The TSF shall enforce the [assignment: *additional information flow control SFP rules*].

**FDP_IFF.1.4/NET** The TSF shall provide the following [assignment: *list of additional*

*SFP capabilities*].

**FDP_IFF.1.5/NET** The TSF shall explicitly authorise an information flow based on the following rules: [*commands to identify the TOE in the network (service discovery) may be accepted and processed without an encrypted connection*].

**FDP_IFF.1.6/NET** The TSF shall explicitly deny an information flow based on the following rules: [*none*].

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control
FMT_MSA.3 Static attribute initialisation

**Application Note**: Please note that the information flow policy defined in FDP_IFC.1/NET and FDP_IFF.1/NET is focused on the communications, which fall into the scope of the application for the electronic health card and which happen between the Connector and the TOE.

Connections for administration of the TOE may not be initiated by a connector. Therefore such a connection may not be covered by this policy.

Further, according to [10] the terminal is free to accept unencrypted communications for other applications, which may be additionally realized by the terminal (or during the migration phase). In these cases the terminal would have to indicate to the medical supplier that it is working in an insecure state.

#### 5.1.2.7 FDP_RIP.1 Subset residual information protection

**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: [*PIN, cryptographic keys, all information that is received by a card in a slot of the TOE or by the Connector except the information that is absolutely necessary for the operation of the TOE,* [assignment: *other object or none*]].

Hierarchical to: No other components.

Dependencies: No dependencies.

**Application Note:** The functionality, defined in FPD_RIP.1 defines that the TOE is not allowed to save any information that was received by the Connector or a card in a slot of the TOE permanently. This is necessary as the TOE

relies on a controlled environment (A.ENV) to provide an adequate level of protection for the assets. If a TOE was e.g. stolen an attacker must not be able to read any of the information that was received from the Connector or a card in a slot of the TOE. Only information that is absolutely indispensible for the operation of the TOE (e.g. a secret that may be used for a pairing as part of the authentication with the Connector) may be stored permanently within the TOE.

The PIN and the cryptographic keys are explicitly mentioned in the assignment as they are not covered by the generic description (as they are neither received from the Connector nor from a card).

### 5.1.3 Identification and Authentication (FIA)

#### 5.1.3.1 FIA_ATD.1 User attribute definition

**FIA_ATD.1.1** The TSF shall maintain the following list of security attributes belonging to individual users: [*Role,* [assignment: *list of security attributes]*].

Hierarchical to: No other components.

Dependencies: No dependencies

**Application Note:** For the case that no further user attributes are needed for any policy of a TOE "none" should be considered as a valid assignment in FIA_ATD.1.1

#### 5.1.3.2 FIA_UAU.1 Timing of authentication

**FIA_UAU.1.1** The TSF shall allow [assignment: *list of TSF mediated actions*] on behalf of the user to be performed before the user is authenticated.

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

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

#### 5.1.3.3 FIA_UAU.5 Multiple authentication mechanisms

**FIA_UAU.5.1** The TSF shall provide [

- *A PIN[6] based authentication mechanism for the user/administrator*
- *A mutual authentication mechanism for the Connector*
- *An unidirectional authentication mechanism for the Connector*
- *[selection: mutual authentication with the Connector based on white lists, none]*
- *[assignment: additional authentication mechanism]*

] to support user authentication.

**FIA_UAU.5.2** The TSF shall authenticate any user's claimed identity according to the [*the user/administrator of the TOE will always be authenticated via PIN based mechanism, for the authentication of/against the Connector the authentication mechanism as configured by the administrator will be used*].

Hierarchical to: No other components.

Dependencies: No dependencies

**Application Note:** Please note that FIA_UID.1 and FIA_UAU.1 primarily refer to the authentication of the user/administrator of the TOE for use by the management or access control function of the terminal. According to [10] this should not be seen as a requirement to maintain the ID of the current user for access control. The scope of these requirements is to determine to which group the current user belongs as the access control mechanism of the TOE primarily works on the basis of the group membership rather than the user ID.

FIA_UAU.5 also describes mechanism to authenticate a Connector and to authenticate the terminal against the Connector. However as the mutual authentication of Connector and terminal can be configured by the administrator it cannot be covered by FIA_UID.1 and FIA_UAU.1.

#### 5.1.3.4 FIA_UID.1 Timing of identification

**FIA_UID.1.1** The TSF shall allow [assignment: *list of TSF-mediated actions*] on behalf of the user to be performed before the user is identified.

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

Hierarchical to: No other components

[6] Please note that instead of a PIN also a password may be used.

Dependencies: No dependencies.

**Application Note**: Although the ST author is in charge of defining the TSF mediated actions, which are allowed without having the user successfully authenticated before, the assignments in FIA_UAU.1.1 and FIA_UID.1.1 have to be performed in a way that none of the TSP of the TOE is violated.

### 5.1.4 Security Management (FMT)

#### 5.1.4.1 FMT_MOF.1 Management of security functions behaviour

**FMT_MOF.1.1** The TSF shall restrict the ability to [*modify the behaviour of*] the functions [*all security functions*] to [*Administrators*].

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

#### 5.1.4.2 FMT_MSA.1 Management of security attributes for terminal SFP

**FMT_MSA.1.1** The TSF shall enforce the [*terminal SFP*] to restrict the ability to [selection: *change_default, query, modify, delete,[*assignment*: other operations]*] the security attributes [*all security attributes of the terminal SFP*] to [assignment: *the authorised identified roles*].

Hierarchical to: No other components.

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

#### 5.1.4.3 FMT_MSA.2 Secure security attributes

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

Hierarchical to: No other components.

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

#### 5.1.4.4 FMT_MSA.3 Static attribute initialisation for terminal SFP

**FMT_MSA.3.1** The TSF shall enforce the [*terminal SFP*] to provide [*restrictive*] default values for security attributes that are used to enforce the SFP.

**FMT_MSA.3.2** The TSF shall allow the [*no roles*] to specify alternative initial values to override the default values when an object or information is created.

Hierarchical to: No other components.

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

#### 5.1.4.5 FMT_SMF.1 Specification of Management Functions

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

- *Change the security relevant network configuration*
- *Management of the available card slots*
- *Perform a firmware update*
- *Reset the configuration of the TOE*
- *Manage the settings for the authentication function*

[assignment: *other relevant management functions or none*]].

Hierarchical to: No other components.

Dependencies: No dependencies.

**Application Note:** For the case that an ST describes a TOE that implements an authentication of the Connector based on white lists the ST author shall add the necessary management functionality for this mechanism to FMT_SMF.1.1

**Application Note:** As part of the authentication between the TOE and the Connector it will be necessary to register the TOE with a Connector before a connection will be accepted by the Connector. If this pairing process requires a functionality of the TOE this shall be defined by the ST author in FMT_SMF.1.1 More details about the pairing process can be found in [10].

#### 5.1.4.6 FMT_SMR.1 Security roles

**FMT_SMR.1.1** The TSF shall maintain the roles [*user, administrator and* [assignment: oth*er roles or none*]].

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

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

### 5.1.5 Protection of the TSF (FPT)

#### 5.1.5.1 FPT_AMT.1 Abstract machine testing

**FPT_AMT.1.1** The TSF shall run a suite of tests [*during initial start-up*, *[*assignment*: other conditions]*] to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF.

Hierarchical to: No other components.

Dependencies: No dependencies.

**Application Note:** FPT_AMT.1 requires – as a minimum – a check of the correct operation of the secure module (see A.SM) during startup. However the ST author is free to add additional scenarios to this SFR.

#### 5.1.5.2 FPT_FLS.1 Failure with preservation of secure state

**FPT_FLS.1.1** The TSF shall preserve a secure state when the following types of failures occur: [assignment: *list of types of failures in the TSF*].

Hierarchical to: No other components.

Dependencies: ADV_SPM.1 Informal TOE security policy model

**Application Note**: As [10] does not define the list of errors for which a secure state has to be preserved. Thus the assignment in FPT_FLS.1.1 is left to the ST author. However – as a minimum the failure of any of the self tests as defined in FPT_TST.1 shall be considered for this assignment.

#### 5.1.5.3 FPT_ITT.1 Basic internal TSF data transfer protection

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

Hierarchical to: No other components.

Dependencies: No dependencies

**Application Note**: Please note that this SFR is easily fulfilled for the cases where a TOE does not comprise physically separated parts or a protection of the communication between those parts is obviously not relevant.

#### 5.1.5.4 FPT_PHP.1 Passive detection of physical attack

**FPT_PHP.1.1** The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF.

**FPT_PHP.1.2** The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.

Hierarchical to: No other components.

Dependencies: No dependencies

#### 5.1.5.5 FPT_SEP.1 TSF domain separation

**FPT_SEP.1.1** The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

**FPT_SEP.1.2** The TSF shall enforce separation between the security domains of subjects in the TSC.

Hierarchical to: No other components

Dependencies: No dependencies.

**Application Note**: Please note that FPT_SEP.1 applies to the communication of the TOE to one or more Connector(s) as well as to the communication of the TOE to one or more smart card(s) in its slots.

#### 5.1.5.6 FPT_TST.1 TSF testing

**FPT_TST.1.1** The TSF shall run a suite of self tests [during initial start-up, at the conditions [assignment*: conditions under which self test should occur*]] to demonstrate the correct operation of [the TSF].

**FPT_TST.1.2** The TSF shall provide authorised users with the capability to verify the integrity of [selection: [*assignment: parts of TSF*], TSF data]].

**FPT_TST.1.3** The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code.

Hierarchical to: No other components.

Dependencies: FPT_AMT.1 Abstract machine testing

**Application Note:** Please note that [10] does not define any concrete requirements for the

minimum functionality that has to be covered by the self test of the TOE. However, as the focus of this requirement is to demonstrate the correct operation of the complete TSF the ST author will have to describe test functionality for all important aspects of all Security Functions that the TOE provides.

### 5.1.6 TOE Access

#### 5.1.6.1 FTA_TAB.1/PIN Default TOE access banners for PIN

**FTA_TAB.1.1/PIN** **Before PIN entry** the TSF shall display **a message** indicating, **which card slot is in use.**

Hierarchical to: No other components.

Dependencies: No dependencies.

**Application Note:** Please note that this requirement only applies for the case that the PIN is sent to a smart card in a slot of the TOE but not for the case that the PIN is used for the authentication of the local user/administrator.

#### 5.1.6.2 FTA_TAB.1/SEC_STATE Default TOE access banners for secure state

**FTA_TAB.1.1/SEC_STATE** Before establishing a user session, **the TSF shall display a message indicating, whether the TOE is in a secure state or not.**

Hierarchical to: No other components.

Dependencies: No dependencies.

**Application Note**: In the context of FAT_TAB.1/SEC_STATE the term “Before establishing a user session” refers to every situation a user is about to use the TOE.

**Application Note**: This SFR is used to meet O.STATE. The “secure state” refers to a mode of operation in which all TSPs of this PP are met and no additional functionality (as allowed by [10]) is active that could compromise a TSP. Specifically the TOE will guarantee a secure PIN entry within such a secure state.

For example according to [10] a TOE could in principle accept unencrypted communications by a third party for applications that are outside the scope of the German Health System. However as long as an unencrypted connection is established the TOE cannot be considered being in a secure state.

This SFR may be implicitly fulfilled for cases, where a TOE doesn’t provide any additional functionality than the functionality, required by this PP and can’t operate in an insecure state.

### 5.1.7 Trusted path/channels (FTP)

#### 5.1.7.1 FTP_ITC.1 Inter-TSF trusted channel

**FTP_ITC.1.1** The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

**FTP_ITC.1.2** The TSF shall permit [*the remote trusted IT product*[7]] to initiate communication via the trusted channel.

**FTP_ITC.1.3** The TSF shall initiate communication via the trusted channel for [*all communications with the Connector*].

Hierarchical to: No other components.

Dependencies: No dependencies.

## 5.2 Security Assurance Requirements for the TOE

The following table lists the assurance components which are applicable to this PP

| **Assurance Class** | **Assurance Components** |
|---|---|
| ACM | ACM_CAP.3 ACM_SCP.1 |
| ADO | **ADO_DEL.2** ADO_IGS.1 |
| ADV | ADV_FSP.1 ADV_HLD.2 ADV_RCR.1 **ADV_IMP.1 ADV_LLD.1, ADV_SPM.1** |
| AGD | AGD_ADM.1 AGD_USR.1 |
| ALC | ALC_DVS.1 **ALC_TAT.1** |
| ATE | ATE_COV.2 ATE_DPT.1 ATE_FUN.1 ATE_IND.2 |
| AVA | **AVA_MSU.3** AVA_SOF.1 **AVA_VLA.4** |

**Table 9: Chosen Evaluation Assurance Requirements**

These assurance components represent EAL 3 augmented by the components marked in bold text. The complete text for these requirements can be found in [3].

The minimum strength of function claim for this Protection Profile is **SOF-high**. This Protection Profile does not contain any security functional requirement for which an explicit strength of function claim is required.

[7] i.e. the Connector

# 6 Rationales

## 6.1 Security Objectives Rationale

The following table provides an overview for security objectives coverage. The following chapters provide a more detailed explanation of this mapping.

| | O.ACCESS_CONTROL | O.PIN_ENTRY | O.I&A | O.MANAGEMENT | O.SECURE_CHANNEL | O.STATE | O.PROTECTION | OE.ENV | OE.ADMIN | OE.CONNECTOR | OE.SM |
|---|---|---|---|---|---|---|---|---|---|---|---|
| T.COM | | | X | | X | | X | | | | |
| T.PIN | X | X | | | | | X | | | | |
| T.DATA | X | | X | X | | | X | | | | |
| OSP.SIGG | | X | | | | X | X | | | | |
| A.ENV | | | | | | | | X | | | |
| A.ADMIN | | | | | | | | | X | | |
| A.CONNECTOR | | | | | | | | | | X | |
| A.SM | | | | | | | | | | | X |

**Table 10: Security Objective Rationale**

### 6.1.1 Countering the threats

The threat **T.COM** which describes that an attacker may try to intercept the communication between the TOE and the Connector is countered by a combination of the objectives *O.I&A*, *O.SECURE_CHANNEL* and *O.PROTECTION*. *O.SECURE_CHANNEL* describes the secure channel, which is used to protect the communication between the TOE and the Connector. This objective basically ensures that an attacker is not able to intercept the communication between the TOE and the connector and removes this threat. *O.I&A* requires that the TOE has to be able to authenticate the Connector. This authentication is part of the establishment of the secure communication between the TOE and the connector and contributes to removing the threat. Finally *O.PROTECTION* ensures that each communication of the TOE with a Connector or cards in its slots is held in a separate security context so that authorized users of the TOE can't influence the communication of other users.

The threat **T.PIN**, which describes that an attacker may try to release the PIN from the TOE, is countered by a combination of the objectives *O.ACCESS_CONTROL*, *O.PIN_ENTRY* and *O.PROTECTION*. *O.ACCESS_CONTROL* defines that according to the access control policy of the TOE nobody must be allowed to read out the PIN. In this way it can be ensured that an attacker cannot read out the PIN via one of the logical interfaces of the TOE *O.PIN_ENTRY*

defines that the TOE shall serve as a secure pin entry device for the user and the administrator and contributes to countering T.PIN as it ensures that the PIN cannot be released from the TOE. This is the main objective that serves to remove the threat. Finally *O.PROTECTION* contributes to countering T.PIN as it ensures that the TOE provides an adequate level of physical protection for the PIN. It further protects the PIN when it is transmitted between physically separated parts, ensures that the PIN is securely deleted when it is not longer used and ensures that the PIN is sent to the correct card as the communication to every card slot is held in a separate context.

The threat **T.DATA**, which describes that an attacker may try to release or change protected data of the TOE, is countered by a combination of *O.ACCESS_CONTROL, O.I&A, O.MANAGEMENT* and *O.PROTECTION*. *O.ACCESS_CONTROL* ensures that only authorized users are able to access the data stored in the TOE. *O.I&A* authenticates the user as the access control mechanism will need to know about the role of the user for every decisions in the context of access control. *O.MANAGEMENT* ensures that only the authorized administrator is able to manage the TSF data and removes the aspect of the threat where an attacker could try to access sensitive data of the TOE via its management interface. Finally *O.PROTECTION* provides the necessary physical protection for the data stored in the TOE and defines additional mechanisms to ensure that secret data cannot be released from the TOE (delete secret data in a secure way, keep communication channels separate and protect data when transmitted between physically separated parts of the TOE). The combination of these objectives removes the threat completely.

### 6.1.2 Covering the OSPs

The organizational security policy **OSP.SIGG** requires that the TOE has to fulfill the requirements to be used as a secure PIN entry device for applications according to [5].

From a functional perspective this means that the TOE has to serve as a secure pin entry device (i.e. that the PIN can never be released from the TOE) and that the TOE has to be able to indicate whether it is working in a secure state.

The secure pin entry device is specified in *O.PIN_ENTRY*. This objective defines that the TOE has to provide a function for secure PIN entry and (as the TOE has more than one card slot) that the TOE will inform the user to which card slot the PIN will be sent. *O.STATE* ensures that the TOE is able to indicate to the medical supplier, whether it is currently working in a secure state as required by OSP.SIGG. Such a secure state includes (but is not limited to) that the secure PIN entry can be guaranteed. Finally *O.PROTECTION* ensures that the TOE is able to verify the correct operation of the TSF and that an adequate level of physical protection is provided.

Further the fact that the TOE shall be compliant to [5] is the major reason for the chosen assurance level as the use of EAL 3 + AVA_MSU.3 and AVA_VLA.4 is required by [5]. Please see chapter 6.2.3 for more details.

### 6.1.3 Covering the assumptions

The assumption **A.ENV** is covered by *OE.ENV* as directly follows.

The assumption **A.ADMIN** is covered by *OE.ADMIN* as directly follows.

The assumption **A.CONNECTOR** is covered by *OE.CONNECTOR* as directly follows.

The assumption **A.SM** is covered by *OE.SM* as directly follows.

## 6.2 Security Requirements Rationale

### 6.2.1 Security Functional Requirements Rationale

The following table provides an overview for security functional requirements coverage.

| | O.ACCESS_CONTROL | O.PIN_ENTRY | O.I&A | O.MANAGEMENT | O.SECURE_CHANNEL | O.STATE | O.PROTECTION |
|---|---|---|---|---|---|---|---|
| FCS_CKM.1 | | | | | X | | |
| FCS_CKM.4 | | | | | | | X |
| FCS_COP.1/TLS | | | | | X | | |
| FCS_COP.1/SIG_VER | | | | X | | | |
| FDP_ACC.1 | X | X | | X | | | |
| FDP_ACF.1 | X | X | | X | | | |
| FDP_IFC.1/PIN | | X | | | | | |
| FDP_IFF.1/PIN | | X | | | | | |
| FDP_IFC.1/NET | | | | | X | | |
| FDP_IFF.1/NET | | | | | X | | |
| FDP_RIP.1 | | | | | | | X |
| FIA_ATD.1 | | | X | | | | |
| FIA_UAU.1 | | | X | | | | |
| FIA_UAU.5 | | | X | | | | |
| FIA_UID.1 | | | X | | | | |
| FMT_MOF.1 | | | | X | | | |
| FMT_MSA.1 | X | | | X | | | |
| FMT_MSA.2 | | | | X | | | |
| FMT_MSA.3 | X | | | | | | |
| FMT_SMF.1 | | | | X | | | |
| FMT_SMR.1 | | | X | | | | |
| FPT_AMT.1 | | | | | | | X |
| FPT_FLS.1 | | | | | | | X |

| | O.ACCESS_CONTROL | O.PIN_ENTRY | O.I&A | O.MANAGEMENT | O.SECURE_CHANNEL | O.STATE | O.PROTECTION |
|---|---|---|---|---|---|---|---|
| FPT_ITT.1 | | | | | | | X |
| FPT_PHP.1 | | | | | | | X |
| FPT_SEP.1 | | | | | | | X |
| FPT_TST.1 | | | | | | | X |
| FTA_TAB.1/PIN | | X | | | | | |
| FTA_TAB.1/SEC_STATE | | | | | | X | |
| FTP_ITC.1 | | | | | X | | |

**Table 11: Coverage of Security Objective for the TOE by SFR**

The Security Objective **O.ACCESS_CONTROL** is met by a combination of the SFR *FDP_ACC.1*, *FDP_ACF.1*, *FMT_MSA.1* and *FMT_MSA.3*. *FDP_ACC.1* defines the access control policy for the terminal and *FDP_ACF.1* defines the rules for the access control policy. It is specifically defined in *FDP_ACF.1* that nobody must be allowed to read out the PIN or private cryptographic keys from the terminal. *FMT_MSA.1* defines, who will be allowed to manage the attributes for the access control policy while *FMT_MSA.3* defines that the terminal has to provide restrictive default values for the access control policy attributes.

The Security Objective **O.PIN_ENTRY** is met by a combination of the SFR *FDP_ACC.1*, *FDP_ACF.1*, *FDP_IFC.1/PIN*, *FDP_IFF.1/PIN*, and *FTA_TAB.1/PIN*. As part of the access control policy of the terminal *FDP_ACC.1* and *FDP_ACF.1* define that nobody must be able to read out the PIN from the terminal, which is required by O.PIN_ENTRY. *FPD_IFC.1/PIN* and *FDP_IFF.1/PIN* build an information flow control policy for the PIN and define that the PIN, which is entered by the user will only be sent to the card slot as indicated. Finally *FTA_TAB.1/PIN* requires that the TOE is able to display a message to inform, which of the card slots of the TOE is in use. This will allow to check that the PIN is sent to the correct card slot.

The Security Objective **O.I&A** is met by a combination of *FIA_ATD.1*, *FIA_UAU.1*, *FIA_UAU.5*, *FIA_UID.1 and FMT_SMR.1*. The Security Objective requires two authentication mechanisms, one for the user of the terminal and one for the communication with the Connector in the environment. *FIA_UID.1* and *FIA_UAU.1* require each user to be authenticated and identified before allowing any relevant actions on behalf of that user. Further the objective requires that the TOE will at least maintain the roles user and administrator. This is defined in *FMT_SMR.1*, which defines the roles and *FIA_ATD.1*, which defines the user attribute for the role. *FIA_UAU.5* defines all the authentication mechanism that shall or can be implemented by the TOE. While *FIA_UAU.1* only refers to the authentication of the user/administrator, *FIA_UAU.5* also list mechanisms to authenticate the Connector and to authenticate the TOE against the Connector (using the secure module for the cryptographic operation).

The Security Objective **O.MANAGEMENT** is met by a combination of *FCS_COP.1/SIG_VER, FDP_ACC.1, FDP_ACF.1, FMT_MOF.1, FMT_MSA.1, FMT_MSA.2* and *FMT_SMF.1*. *FCS_COP.1/SIG_VER* is used to define the mechanism to check the authenticity of a firmware update. The access control policy defined in *FDP_ACC.1* and *FDP_ACF.1* defines the rules under which a firmware update is possible. FMT_MOF.1 defines that only the administrator is allowed to change the behavior of all Security Functions. *FMT_MSA.1* defines, which roles are allowed to administer the attributes of the access control and the information flow control policies. *FMT_MSA.2* requires that only secure values are accepted for security attributes. Finally *FMT_SMF.1* describes the minimum set of management functionality, which has to be available according to the Security Objective.

The Security Objective **O.SECURE_CHANNEL** is met by a combination of the SFR *FCS_CKM.1, FCS_COP.1/TLS, FDP_IFF.1/NET and FDP_IFC.1/NET.,* and *FTP_ITC.1*. *FCS_CKM.1 and FCS_COP.1/TLS* define the cryptographic operations, which are necessary for this objective. *FCS_CKM.1* defines that the TOE has to be able to generate (negotiate) cryptographic keys, which can be used to secure the communication with the Connector, *FCS_COP.1/TLS* defines the functionality for encryption and decryption itself. The information flow control policy in *FDP_IFF.1/NET* and *FDP_IFC.1/NET* defines that at the network interface only a command to locate the TOE may be available without an encrypted connection and that all other communications must only be accepted if the secure channel to the Connector has been established before. Finally *FTP_ITC.1* defines the trusted channel itself, which is used to secure the communication between the TOE and the Connector.

**O.STATE** is directly and completely met by *FTA_TAB.1/SEC_STATE* as this SFR requires that the TOE shall be able to indicate, whether it is working in a secure state.

The Security Objective **O.PROTECTION** is met by a combination of the SFR *FCS_CKM.4, FDP_RIP.1, FPT_ITT.1, FPT_PHP.1, FPT_SEP.1, FPT_AMT.1, FPT_FLS.1 and FPT_TST.1*. *FCS_CKM.4* defines that cryptographic keys have to be securely deleted when they are not longer used. *FDP_RIP.1* defines the same additionally for the PIN and also ensures that an attacker cannot read other protected information from the TOE even if the TOE is not longer in its protected environment. *FPT_ITT.1* defines that the TOE has to protect TSF data when it is transmitted between physically separated parts of one TOE. *FPT_PHP.1* builds the physical protection for the stored assets. *FPT_SEP.1* defines that the TOE shall provide domain separation for communications with more than one Connector and more than one card. *FPT_AMT.1* defines the necessary test functionality for the underlying abstract machine. *FPT_FLS.1* defines a list of failures in the TSF for which the TOE has to preserve a secure state. Finally *FPT_TST.1* defines that the TSF have to run a suite of self tests to demonstrate the correct operation of the TSF at startup and during the normal operation of the TOE.

### 6.2.2 Dependency Rationale

| SFR | Dependencies | Support of the Dependencies |
|---|---|---|
| FCS_CKM.1 | [FCS_CKM.2 Cryptographic key distribution, or<br>FCS_COP.1 Cryptographic operation]<br>FCS_CKM.4 Cryptographic key destruction<br>FMT_MSA.2 Secure security attributes | Fulfilled by the use of FCS_COP.1/TLS, FCS_CKM.4 and FMT_MSA.2 |
| FCS_CKM.4 | [FDP_ITC.1 Import of user data without security attributes, or<br>FDP_ITC.2 Import of user data with security attributes, or<br>FCS_CKM.1 Cryptographic key generation]<br>FMT_MSA.2 Secure security attributes | Fulfilled by the use of FCS_CKM.1 and FMT_MSA.2 |
| FCS_COP.1/TLS | [FDP_ITC.1 Import of user data without security attributes, or<br>FDP_ITC.2 Import of user data with security attributes, or<br>FCS_CKM.1 Cryptographic key generation]<br>FCS_CKM.4 Cryptographic key destruction<br>FMT_MSA.2 Secure security attributes | Fulfilled by the use of FCS_CKM.1, FCS_CKM.4 and FMT_MSA.2 |
| FCS_COP.1/SIG_VER | [FDP_ITC.1 Import of user data without security attributes, or<br>FDP_ITC.2 Import of user data with security attributes, or<br>FCS_CKM.1 Cryptographic key generation]<br>FCS_CKM.4 Cryptographic key destruction<br>FMT_MSA.2 Secure security attributes | Fulfilled by FMT_MSA.2<br>See chapter 6.2.2.1 for FDP_ITC.1 and FCS_CKM.4 |
| FDP_ACC.1 | FDP_ACF.1 Security attribute based access control | Fulfilled |
| FDP_ACF.1 | FDP_ACC.1 Subset access control<br>FMT_MSA.3 Static attribute initialization | Fulfilled |
| FDP_IFC.1/PIN | FDP_IFF.1 Simple security attributes | Fulfilled by FDP_IFF.1/PIN |
| FDP_IFF.1/PIN | FDP_IFC.1 Subset information flow control<br>FMT_MSA.3 Static attribute initialisation | Fulfilled by FDP_IFC.1/PIN<br>See chapter 6.2.2.1 |

| SFR | Dependencies | Support of the Dependencies |
|---|---|---|
| | | for FMT_MSA.3 |
| FDP_IFC.1/NET | FDP_IFF.1 Simple security attributes | Fulfilled by FDP_IFF.1/NET |
| FDP_IFF.1/NET | FDP_IFC.1 Subset information flow control<br>FMT_MSA.3 Static attribute initialisation | Fulfilled by FDP_IFC.1/NET<br>See chapter 6.2.2.1 for FMT_MSA.3 |
| FDP_RIP.1 | No dependencies | - |
| FIA_ATD.1 | No dependencies | - |
| FIA_UAU.1 | FIA_UID.1 Timing of identification | Fulfilled |
| FIA_UAU.5 | No dependencies | - |
| FIA_UID.1 | No dependencies | - |
| FMT_MOF.1 | FMT_SMR.1 Security roles<br>FMT_SMF.1 Specification of Management Functions | Fulfilled |
| FMT_MSA.1 | [FDP_ACC.1 Subset access control, or<br>FDP_IFC.1 Subset information flow control]<br>FMT_SMR.1 Security roles<br>FMT_SMF.1 Specification of Management Functions | Fulfilled by FDP_ACC.1, FMT_SMR.1 and FMT_SMF.1 |
| FMT_MSA.2 | ADV_SPM.1 Informal TOE security policy model<br>[FDP_ACC.1 Subset access control, or<br>FDP_IFC.1 Subset information flow control]<br>FMT_MSA.1 Management of security attributes<br>FMT_SMR.1 Security roles | Fulfilled by ADV_SPM.1, FPD_ACC.1, FDP_IFC.1/PIN, FDP_IFC.1/NET, FMT_MSA.1, and FMT_SMR.1 |
| FMT_MSA.3 | FMT_MSA.1 Management of security attributes<br>FMT_SMR.1 Security roles | Fulfilled by FMT_MSA.1 and FMT_SMR.1 |
| FMT_SMF.1 | No dependencies | - |
| FMT_SMR.1 | FIA_UID.1 Timing of identification | Fulfilled |
| FPT_AMT.1 | No dependencies | - |
| FPT_FLS.1 | ADV_SPM.1 Informal TOE security policy model | Fulfilled |

| SFR | Dependencies | Support of the Dependencies |
|---|---|---|
| FPT_ITT.1 | No dependencies | - |
| FPT_PHP.1 | No dependencies | - |
| FPT_SEP.1 | No dependencies | - |
| FPT_TST.1 | FPT_AMT.1 Abstract machine testing | Fulfilled |
| FTA_TAB.1/PIN | No dependencies | - |
| FTA_TAB.1/SEC_STATE | No dependencies | - |
| FTP_ITC.1 | No dependencies | - |

**Table 12: Dependencies of the SFR for the TOE**

#### 6.2.2.1 Justification for missing dependencies

The dependencies of the information flow policies FDP_IFF.1/PIN and FDP_IFF.1/NET to FMT_MSA.3 was considered to be not applicable as both information flow policies do not require any security attributes.

For the case that the ST author would extend these information flow policies in a way that they require security attributes they shall consider the dependency to FMT_MSA.3.

The dependencies FDP_ITC.1 and FMT_MSA.2 of FCS_COP.1/SIG_VER result out of the original scope of FCS_COP.1 to specify the implementation of encryption functionality within a TOE. These dependencies deal with the import (or creation) and destruction of a secret key that is needed for encryption. However, as in the context of this PP FCS_COP.1/SIG_VER is used for a requirement on signature verification for which no secret key is necessary these dependencies do not need to be considered.

### 6.2.3 Security Assurance Requirements Rationale

The Evaluation Assurance Level for this Protection Profile is EAL 3 augmented by ADO_DEL.2, ADV_LLD.1, ADV_IMP.1, ADV_SPM.1, ALC_TAT.1, AVA_MSU.3 and AVA_VLA.4.

The main decision about the Evaluation Assurance Level has been taken based on the fact that the TOE described in this Protection Profile shall serve as a secure PIN entry device according to [5] (see also OSP.SIGG).

This leads to an Evaluation Assurance Level of 3 augmented by the following components:

- AVA_MSU.3
- AVA_VLA.4

These components have the following direct and indirect dependencies:

- ADV_IMP.1
- ADV_LLD.1
- ALC_TAT.1

Further the evaluation of a secure PIN entry device according to [5] shall be comparable to an ITSEC Evaluation E2 high. According to [4] this made it necessary to choose one additional augmentation: ADO_DEL.2.

Finally the use of the SFR FMT_MSA.2 resulted in an augmentation by ADV_SPM.1.

Considering the use of AVA_VLA.4 and the requirements from [5] the minimum strength of function for any TOE claiming compliance to this PP has to be SOF-high.

### 6.2.4 Security Requirements – Mutual Support and Internal Consistency

The core TOE functionality in this PP is represented by the requirements for access control (FDP_ACC.1 and FDP_ACF.1) and information flow control (FDP_IFC.1/PIN, FDP_IFF.1/PIN, FDP_IFC.1/NET and FDP_IFF.1/NET).

Further functionality to protect the communication is defined by the requirements for cryptographic support and the trusted channel.

In the end this PP contains a set of SFRs which deal with the detection and defeating of attacks to the TOE, resp. SFRs which are used to show that the TOE is working correctly (e.g. FPT_PHP.1, FPT_TST.1) In this way the SFRs in this PP mutually support each other and form a consistent whole.

From the details given in this rationale it becomes evident that the functional requirements form an integrated whole and, taken together, are suited to meet all security objectives. Requirements from [2] are used to fulfil the security objectives.

# 7 Extended Functionality

The present Protection Profile describes the Security Requirements for the E-health card terminal in the context of [10]. Concepts and specifications of further procedures and functionality for E-health terminals – specifically the descriptions of batch signatures ("Stapelsignaturen") and Remote-PIN entry – were not available by the time this PP was developed.

Thus this Protection Profile does not contain any requirements associated with such extended functionalities of the E-health card terminal.

However, a TOE claiming compliance to this Protection Profile may provide additional functionality in the context of batch signatures and/or Remote-PIN entry. If this is the case the additional functionality shall be modelled in the Security Target and addressed during evaluation. In this way it can be ensured that the security policies as defined by this Protection Profile are not violated by the extended functionality.

The author of this Protection Profile would like to highlight the following aspects to be considered during evaluation of a TOE that supports batch signatures and/or Remote PIN entry:

For batch signatures:

- Batch signatures should not require any additional Security Functionality of the card terminal as they are implemented by the smart card that is generating the signatures; specifically the terminal shall not be used to store the PIN after is has been acquired from the medical supplier.
- However it has to be ensured that medical suppliers are informed about the fact that they are about to start a batch signature process rather than to create just one signature[8].

For Remote PIN entry:

- The concept of Remote PIN entry requires two terminals:
  - A terminal to acquire the PIN from the medical supplier and
  - A terminal that receives the (encrypted) PIN and forwards it to a card
- For the terminal that receives the PIN there should be no need for any additional Security Function as the data packet that contains the PIN will be treated as any other information
- For the terminal that acquires the PIN from the medical supplier the following aspects shall be considered:
  - The PIN has to be handled in a way that does not allow any misuse[9]. Usually the security of the PIN is guaranteed by ensuring that it never leaves the TOE (which is not longer true for the Remote-PIN concept). The mechanisms that are used for protection of the Remote-PIN have to provide a comparable level of protection.

[8] This can be done by the card terminal but also by another system in the context of the German health card system (e.g. the Connector)

[9] To be precise: the secure handling of the PIN would be done by the terminal with support of a SMC

- It has to be ensured that the medical supplier is informed about the fact that his PIN will be sent to a remote terminal.
- The medical supplier has to have the (organisatorical or technical) possibility to ensure that the PIN is only sent to a trustworthy entity (HPC) and that the connection with the HPC is appropriately secured[10].

Once final concepts and descriptions for the extended functionality of batch signatures and Remote-PIN are available an update of this Protection Profile shall be considered.

[10] As it is likely that at least a part of this confidence will be achieved by organistorical measures this does also imply that a remote PIN entry must only be possible for a HPC and not for a EHC as only the medical supplier will have the necessary control over the infrastructure that is involved with the remote PIN entry.

# 8 Glossary and Acronyms

| Term | Definition |
|---|---|
| AES | Advanced Encryption Standard |
| *BSI* | Bundesamt für Sicherheit in der Informationstechnik |
| *EHC* | Electronic Health Card |
| *HPC* | Health Professional Card |
| *LAN* | Local Area Network |
| *PP* | Protection Profile |
| *SFP* | Security Function Policy |
| *SFR* | Security Functional Requirement |
| *ST* | Security Target |
| *SM-KT* | Sicherheits Modul Karten Terminal |
| *TOE* | Target of Evaluation |
| *TSF* | TOE Security Function |

# 9 Literature

**Common Criteria**

[1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 2.3, August 2005

[2] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; Version 2.2, August 2005

[3] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; Version 2.2, August 2005

[4] AIS 27, Version 2, Transition from ITSEC to CC, Certification body of the BSI in the context of the certification scheme, June, 23rd 2005

**Cryptography**

[5] Gesetz über Rahmenbedingungen für elektronische Signaturen und zur Änderung weiterer Vorschriften, 16. Mai 2001, Bundesgesetzblatt Nr. 22

[6] BSI TR-03116, Technische Richtlinie für die eCard-Projekte der Bundesregierung, Version 1.0 23.03.2007

[7] RFC 3268 Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security (TLS), June 2002, http://www.ietf.org/rfc/rfc3268.txt

[8] RFC4346 The TLS Protocol, Version 1.1

[9] Federal Information Processing Standards Publication 180-2 SECURE HASH STANDARD U.S. DEPARTMENT OF COMMERCE/National Institute of Standards and Technology, 2002 August 1

**Specifications**

[10] Spezifikation eHealth-Kartenterminal, Version 2.2.0 24.08.2007

[11] TeleTrusT SICCT-Spezifikation as referenced by [10]