Electronic Health Card Terminal eHealth GT900 Security Target

GT German Telematics
Gesellschaft für Telematikdienste mbH
Libellenstraße 9
14129 Berlin

Revision History

Version	Date	Description	Author
1.4	19.02.2009	BCS zugelassenes Security Target für das Chipkartenterminal GT900	Jan Mihalyovics
1.5		Überarbeitung des gesamten Security Targets gemäß BSI-CC-PP-0032 v.2.3	Jan Mihalyovics
1.6	09.12.2013	Überarbeitung auf aktuelles PP v1.1	Jan Mihalyovics
1.6.1		Simplified Layout Converted to .doc Format for better compatibility. Added TOE hardware versions Rewritten TOE Overview Updated References	Peter Wippich
1.6.2		Updated TOE description	Peter Wippich
1.6.3	24.01.2013	Completion of Sec. 2 (including application notes) Sec. 3 (reference to BSI-CC-PP-0032) Sec. 4 (reference to BSI-CC-PP-0032) Sec. 5 (no extended components) Sec. 6.1/6.2 (all operations performed, all application notes processed) Sec. 6.3. (reference to BSI-CC-PP-0032)	Roland Vogt (DFKI)
1.6.4	17.2.2014	Rewritten 1.4 Added Zertifizierungskennung Used consistent naming for TOE Added TOE Type description Updated references Added TOE scope of delivery	Peter Wippich
1.6.5	21.02.2014	Adapted Sec. 1.3 to PP V3.0 Changes due to dsc comments (ASE DokuCheck, V1.0, 31.01.2014)	Roland Vogt (DFKI)
1.6.6	28.03.2014	Corrections w.r.t. ETR Part ASE V1.0 Minor changes in SFR operations: FCS_CKM.1.1/Connector FCS_COP.1.1/Connector FDP_ACF.1.2/Management FIA_AFL.1.2 FMT_SMF.1.1	Roland Vogt (DFKI)
1.7.0	01.07.2014	Changes due to BSI comments (ZK_0594_V2_ASE, V1.0, 25.06.2014) Sec. 7 rewritten (aligned with functional specification)	Roland Vogt (DFKI)
1.7.1	29.07.2014	Minor corrections due to evaluator findings	Roland Vogt (DFKI)
1.7.2	30.7.2014	Corrected Cipher Suite Naming	Peter Wippich

1.8.0	13.02.2015	Adaptation to PP V3.5 (all sections);
		Alignment of references with PP V3.5;
		Corrections w.r.t. ETR Part ASE V1.2;
		Changes due to dsc comments on development documentation.
			Roland Vogt (DFKI)

1.8.1	24.02.2015	Changes in TOE summary specification	Roland Vogt (DFKI)
1.8.2	18.03.2015	Corrections w.r.t. ETR Part ASE V1.4	Roland Vogt (DFKI)
1.8.3	22.05.2015	Minor editorial corrections w.r.t. ETR Part ASE V1.5	Roland Vogt (DFKI)
1.8.4	28.05.2015	Correction of revision history (description of V1.7.2 was lost)	Roland Vogt (DFKI)
1.8.5	02.10.2015	Adaptation to PP V3.6 (all sections).	Roland Vogt (DFKI)
1.8.6	27.11.2015	Minor corrections due to evaluator findings	Roland Vogt (DFKI)
1.8.7	28.01.2019	Adaptation to PP 3.7 (all sections)	GT German Telematics
1.8.8	06.05.2020	Adaption to FW-Version 1.22.1	Frank Reinl
1.8.9	11.08.2020	FW-Version 1.22.2 due to gematik request Adaption to: HW-Version 2.1.0; BSI TR-03120 Version 1.1 Secure messaging added Corrections: Form of the TOE in sec. 1.2 and sec. 1.3.5; CC rev. in sec. 2.1; list of application notes in sec. 2.3, minor corrections	Frank Reinl
1.9.0	15.02.2021	Added: Active monitoring of the housing of the TOE	Frank Reinl
1.9.1	14.04.2021	Corrections due to Observation_report_ASE_0.2, Certificate-ID added	Frank Reinl
1.9.2	27.04.2021	Added: USB device interface is part of the LAN interface	Frank Reinl
1.9.3	29.07.2021	FW-Version 1.22.3 due to gematik request Changes due to BSI and ITSEF comments (ZK_1171_ASE, V1.1, 15.06.2021)	Frank Reinl
1.9.4	04.08.2022	Changes due to BSI (ZK_1171_ASE, V2.0, 23.11.2021)	Frank Reinl
1.9.5	07.12.2022	FW-Version 2.0.1. Signatures of update files are generated with a 4096 bit RSA key. Cryptographic parameters precised.	Frank Reinl

Table of Contents

1 ST Introduction

1.1 ST reference

Title:	Electronic Health Card Terminal eHealth GT900 Security Target
Version:	1.9.5
Date:	07.12.2022
Authors:	Jan Mihalyovics (up to V1.6), Peter Wippich (up to V1.7.x), Roland Vogt (DFKI, since V1.6.x), GT German Telematics (V1.8.7), Frank Reinl (since V1.8.8)
Developer:	GT German Telematics Gesellschaft für Telematikdienste mbH Libellenstraße 9 14129 Berlin
Certification ID:	BSI-DSZ-CC-1171

1.2 TOE reference

The target of evaluation (TOE) is the electronic Health Card Terminal eHealth GT900 (short name: eHealth GT900), manufactured by German Telematics GmbH in the following versions:

Hardware - Version:	**2.1.0**	colour white
Hardware - Version:	**2.1.0 SW**	colour black
Hardware - Version:	**2.1.0 SI**	colour silver
Firmware - Version:	**2.0.1**	

1.3 TOE overview

This Security Target defines the security objectives and requirements for the Electronic Health Card Terminal eHealth GT900 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 functionality including update and downgrade of Firmware, and
- Active physical protection.

1.3.1 TOE definition and operational usage

The Target of Evaluation (TOE) described in this Security Target 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. Please refer to [17] for further information about card compatibility.

The TOE fulfils the requirements to be used as a secure PIN pad entry device for applications according to [6] and [7], which specifically means that a PIN, which has been entered by a user at the TOE, never leaves the TOE in clear text, except to smart cards in local card slots.

This terminal does not implement any insecure mode (e.g. in any mode, it can be guaranteed that the PIN will not leave the TOE and only trustworthy entities are allowed to communicate with the TOE). Thus, the TOE does not need to be able to inform the user whether it is currently in a secure state or not.

This terminal bases on the specification for a “Secure Interoperable ChipCard terminal” ([18]) extended and limited by the specifications for the e-health terminal itself (see [17]).

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 functionality including update and downgrade of Firmware, and
- Active physical protection.

The TOE for use in the German health care is based on the specification SICCT, which is adapted for operation by profiling as eHealth card terminal (see [18]).

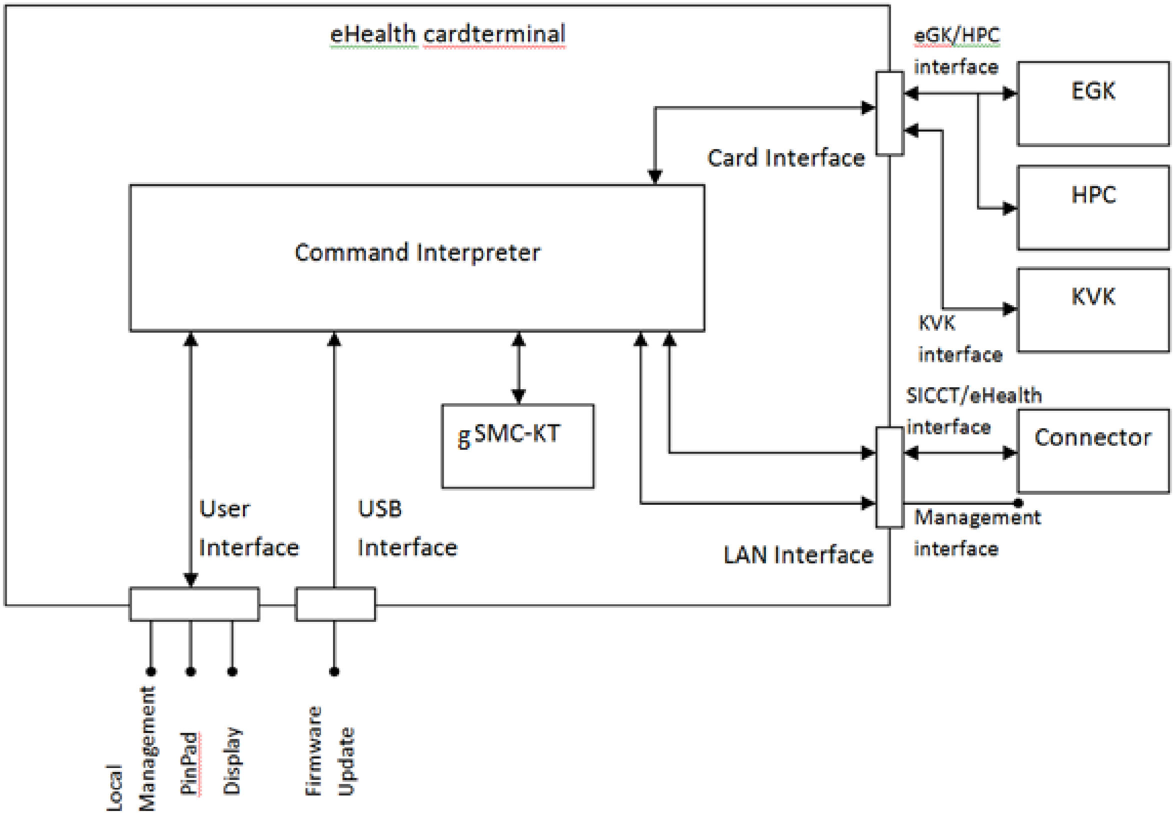
The derivatives of the physical characteristics of each card terminal types are presented in Figure 1 which is based on the architecture of the model specification SICCT [18].

A PIN pad and a display are provided by the TOE (see [18]). Also provided are two ID-1 and two ID-000 contact units.

The TOE must work with a cryptographic key for i.e. authentication, integrity assurance and to ensure the confidentiality of data transmitted over the LAN interface. Due to the very high protection requirements of the information objects transmitted over the LAN interface, a secure key store (SM-KT) is required for the key. As physical characteristics of the SM-KT the TOE supports gSMC-KT cards. IPv4 is supported. To ensure the sustainability of the TOE, it will be able to support IPv6 in addition to IPv4 only with a future firmware update[1].

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.

For the connection of the TOE to a connector via the LAN interface, the protocol with the SICCT commands is mandatory. The interfaces of the TOE and communication partners using them are provided in Figure 1.



[1] Please note that firmware update could also mean a firmware downgrade. Both actions are possible. In case of a downgrade of the firmware the TOE will warn the Administrator (e.g. within the Guidance) before the installation that the action to be performed is not an upgrade. The TOE will offer a chance to cancel the installation.

Figure 1: TOE architecture (logical perspective)

1.3.2 TOE major security features for operational use

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 (see also [17]).

For its cryptographic functionality the TOE relies on the services of the so called SM-KT[2].

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 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 Figure 1. More information about the SM-KT can be found in the Protection Profile Card Operating System 2 (PP COS G2) [15] and the gematik specification on the gSMC-KT object system [19].

For the case the TOE uses a DF.KT of a gSMC-KT as SM-KT, which is addressable via the connector, the TOE will access this DF.KT via the base-channel 0. During use of the SM-KT by the TOE, the terminal card commands of the TOE are priorized and the processing of possibly existing client SICCT commands will be interrupted and continued only after completion of the internal command sequence. The connector has to make sure that a DF.KT of a gSMC-KT as SM-KT which is addressable via the connector shall only be accessed by the TOE and not be used by any other system than the TOE.

The TOE provides functionality to update and downgrade its firmware. This includes both the change to a newer firmware as a downgrade to a firmware which is approved with the concept of firmware-group. The configuration, such as terminal type, IP address or pairing- information will be preserved and indicated after a firmware update or a downgrade (see [17] for further information). The developer of the TOE will ensure that in case of a downgrade of the firmware of the TOE the Administrator shall be warned (e.g. within the Guidance) before the installation that the action to be performed is not an upgrade. The TOE will offer a chance to cancel the installation. The update component will warn the administrator about taking the responsibility in case of performing a downgrade.

Firmware update can also be triggered remotely from a trusted Push Server in the internal network of the medical supplier. The TOE allows initiating batch signatures for the creation of more than one signature at a time without providing the PIN for each signature process. Batch signature is a functionality of the signing card.

[2] Please note that the SMC-KT is only responsible for the core functions of the asymmetric cryptography (RSA) and for random number generation. 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 [17] and the following chapters.

In addition to the cryptographic identity of the TOE, the TOE stores a shared secret which is generated by the connector and transferred to the TOE during the pairing process of TOE and connector. This shared secret is not stored in the SM-KT, but in a separate storage area of the TOE. As the SM-KT might be removed and placed into another card terminal, the shared secret is necessary to ensure that communication to the connector is performed using the already paired card terminal (the TOE). The whole identity of the TOE is therefore represented by the SM-KT certificate AND the shared secret. Please note that as part of the pairing process, there are three processes:

- Initial pairing:
  This provides a logical connection from the perspective of the connector by using shared secret between card terminal and SM-KT
- Review of pairing-information:
  The connector checks as a second step of authentication, if the card terminal is in the possession of the shared secret after establishing the TLS connection.
- Maintenance-pairing:
  Announcement of a new connector certificate on the card terminal by using a known shared secret. Please see [17] for further information on the pairing process.

The TOE is also able to send/receive a PIN to/from a remote card terminal. This communication is routed via the connector. The connector never sees the PIN in clear text, as the gSMC-KT is used to encrypt the PIN in the local card terminal and the authorized card (SMC-B, HPC) in the remote card terminal decrypts the PIN itself.

1.3.3 TOE Type

The TOE is an Electronic Health Card Terminal based on the regulations for the German healthcare system. It is a smart card terminal with secure PIN entry functionality and fulfils the requirements for the use within the German telematics infrastructure...

The TOE is a stand-alone card terminal. The physical scope of the TOE comprises

- The hardware and sealed cage of the smart card terminal,
- The firmware of the smart card terminal, and
- The related guidance document (Kartenterminal eHealth GT900 - Benutzerhandbuch - Version 2.1.4)

Please note that though the SM-KT will be physically within the cage of the terminal this module does not belong into the scope of the TOE as described in this Security Target.

One or more seals are attached to the cage of the terminal allowing the user of the TOE to detect whether the TOE has been tampered with. The description on how to check the sealing is part of the TOE guidance documentation.

Further note, that the SM-KT is a necessary requirement in the operational environment of the TOE. During the delivery and setup phase the SM-KT may have to be installed into the card terminal. Functionality that is relying on the SM-KT for secure operation may not work as intended before the SM-KT is installed.

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 including update of Firmware, and
- Active physical protection.

And is limited by the functionality for which the TOE relies on the services of the SM-KT, which is not part of the TOE.

According to [17] compliance with this ST does only represent a part of the registration process for an Electronic Health Card Terminal. Additionally [17] requires:

- That the terminal has to be compliant to the requirements in [17] and [18] and
- That the terminal has to undergo a registration process of the gematik.

It should be mentioned that according to [17] it would be allowed that a terminal, claiming compliance to this ST, implements more functionality than defined in this ST and that a terminal temporarily operates in an insecure state. In such a state parts of the security functionality as required by this ST may not be available. However, the eHealth GT900 terminal does not implement such additional functionality and always operates in a secure state.

1.3.4 Required non-TOE hardware/software/firmware

The TOE is intended to be used as 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.

The following non-TOE hardware is required for the use the TOE:

- An ID-000 smart card as a secure module representing the cryptographic identity of the TOE in form of an X.509 certificate. The secure module can be a DF.KT of a gSMC-KT as SM-KT. Although this secure module is usually physically placed within the cage of the TOE it does not belong to the logical and physical scope of the TOE.
- A connector as a secure connection between the local network of the medical supplier and the remote network of the telematic infrastructure. The connector further observes the TOE and is the only entity which can interact with a DF.KT of a gSMC-KT as SM-KT as mentioned above.

1.3.5 TOE Scope of delivery

When the TOE is delivered the scope of delivery includes

- The physical device with pre-installed firmware
- external power supply

The user guidance documentation "Chipkartenterminal GT900 – Benutzerhandbuch" is available in electronic form. It can be downloaded at the manufacturer's website www.germantelematics.de.

Even though an SM-KT is required to operate the TOE, the SM-KT is out the scope of delivery of the TOE.

1.4 TOE description

The operation of the TOE is defined by its firmware. This firmware is stored in non volatile memory inside the TOE. The firmware can be updated in a secure way.

The TOE will only accept firmware updates which are properly signed and include the required versioning information as specified by [20] and [21]. The signature is generated using SHA256 for integrity and RSA4096 encryption for authenticity. The signature is generated over all parts of the firmware including the versioning information.

For its secure operation the TOE heavily relies on the SM-KT [6] which provides the cryptographic identity of the TOE in the form of an X509 certificate. The SM-KT also provides basic cryptographic functions like signature generation, RSA encryption and decryption and random number generation. The SM-KT is not part of the TOE but installed by the TOE operator during the installation process. To protect the SM-KT and, as such, the cryptographic identity of the TOE after the SM-KT has been installed, the card slot containing the SM-KT must be sealed by the operator. This process is described in the guidance documentation.

The TOE supports different operation modes. Modes which require authentication (PIN/PUK/password entry) are left automatically after a specified time of user inactivity. These modes are Administrator Mode and Reset Administrator mode.

A Un-configured Mode

This is the only mode available in delivery state or after a factory reset. The user has to enter an administrator PIN and a reset administrator pin (PUK) before any other modes can be used.

B Normal Mode

During normal operation the TOE communicates only with a so called connector over a LAN connection. The communication is protected by TLS. The TOE uses the SM-KT as its cryptographic identity and for symmetric key generation (for generating random numbers) for this TLS connection. The certificate presented by the connector during connection set up is verified against a root certificate (CA) which is stored inside the TOE. If the certificate is not a valid connector certificate only a minimal set of commands, as specified in [17], will be executed. To allow full operation and, especially, access to chip cards, the connector must be paired with the TOE. The process of pairing is described in [17]. During this process a shared secret is exchanged between the TOE and the connector which is stored encrypted[3] inside the TOE.

[3] The shared secret will only be decrypted in the main processor when it is used and after that the decrypted secret is immediately deleted in a secure way (FDP_RIP.1).

This is used to perform a challenge - response authentication each time a new connection is set up. Full access is only granted if the authentication was successful.

During normal operation the display of the TOE will be used to display status information and messages. Status display includes current network connection status and PIN entry security to make sure the user can verify that the TOE is in a secure PIN entry state before entering a card PIN.

C Administrator Mode

To enter this mode the user has to enter the local administrator PIN, the remote administrator password or the SICCT password. Only in this mode settings like the administrator PIN/password, the network configuration and others can be changed. Further, the user may perform firmware updates or a factory reset in administrator mode.

D Reset Administrator mode

To enter this mode the user has to enter the reset administrator PUK or the reset response to a corresponding challenge. In this mode the only operation possible is to reset the TOE to factory defaults. All stored data including the connector pairing information and user settings are lost, but the firmware itself remains unchanged.

1.4.1 Physical Scope of the TOE

The TOE comprises the following physical components:

- electronic hardware including non volatile memory for persistent storage of firmware, CA certificates for connector verification, pairing information and configuration data in a single sealed housing.
- firmware stored inside the TOEs persistent storage
- user guidance

The TOE provides the following physical interfaces:

- a graphical LC display for user guidance and administrative purposes
- a pin pad for secure pin entry and administrative purposes
- two ID1 chip card slots for patients and medical supplier cards
- two ID000 chip card slots for SMC-B or SM-KT cards which can be sealed
- an Ethernet interface to communicate with the connector
- an USB host interface to access USB storage devices for firmware update
- an USB device interface to communicate with the connector
- an RS232 interface (logically inactive; not used by any parts of the firmware)

The following components are important in the context of this ST but are not part of the TOE:

- chip cards SM-KT, SMC-B
- connector

1.4.2 Logical Scope of the TOE

The logical scope of the TOE can be defined by its security functionality:

- Secure remote management over the network interface (Ethernet or USB).
- Secure network connectivity over the network interface (Ethernet or USB).
- Encryption enforcement of the communication over the network interface (Ethernet or USB).
- Secure PIN entry functionality.
- Management functionality including firmware update functionality.
- User authentication for accessing the management functionality and factory reset.
- Secure deletion of unused confidential data when it is no longer used.
- Self-test and protection of physical tampering.

The logical scope of the TOE is limited in functionality where it relies on functions provided by an SM-KT.

2 Conformance claims

2.1 CC conformance claim

The ST and the TOE claim conformance to CC Version 3.1 Revision 5.

The ST is ***CC Part 2 conformant***.

The ST is ***CC Part 3 conformant***.

The ST is ***EAL 3 augmented*** with components
ADV_FSP.4, ADV_IMP.1, ADV_TDS.3, ALC_TAT.1 and AVA_VAN.4.

2.2 PP conformance claim

The ST claims strict conformance to the following PP:

Common Criteria Protection Profile
Electronic Health Card Terminal (eHCT), BSI-CC-PP-0032-V2-2015-MA-01, V3.7, 21th September 2016

2.3 Conformance claim rationale

The TOE type of this ST is identical to the TOE type of BSI-CC-PP-0032-V2-2015-MA-01.

The security problem definition of this ST is identical to the security problem definition of BSI-CC-PP-0032-V2-2015-MA-01.

The security objectives of this ST are identical to the security objectives of BSI-CC-PP-0032-V2-2015-MA-01.

The security requirements of this ST are identical to the security requirements of BSI-CC-PP-0032-V2-2015-MA-01. In particular, the TOE does not offer any extended functionality, although so called value-added modules (VAM) would be explicitly permitted by BSI-CC-PP-0032-V2-2015-MA-01.

The PP application notes are processed as follows:

1) Inherited
2) Inherited
3) Inherited
4) Inherited
5) Inherited
6) Inherited
7) Inherited
8) Applied (no additional rules, no unauthorized reset), then partially deleted/inherited
9) Applied (no further management functions), then partially deleted/inherited
10) Inherited
11) Inherited
12) Inherited
13) Inherited
14) Inherited
15) Inherited
16) Inherited
17) Inherited
18) Applied (no other TSF mediated actions), then deleted
19) Inherited
20) Applied (no unauthorized reset), then partially deleted/inherited
21) Applied (secure state in case of self-test failures), then deleted
22) Inherited
23) Inherited (recommendation followed)
24) Applied (description in TOE summary specification), then deleted
25) Inherited
26) Inherited
27) Inherited

3 TOE Security problem definition

This chapter describes

- the assets that need to be protected by the TOE,
- the subjects that are interacting with the TOE,
- the threats that have to be countered by the TOE,
- the organizational security policy that TOE shall comply with, and
- the assumptions that need to be ensured for the environment of the TOE.

These descriptions are literally identical to the respective sections of BSI-CC-PP-0032-V2-2015-MA-01. In particular, the definition of assets does not identify further communication data, configuration data or TSF data, although this would be explicitly permitted by BSI-CC-PP-0032-V2-2015-MA-01.

3.1 Assets

The following assets need to be protected by the TOE as long as they are in the scope of the TOE:

Asset	Description
Card PIN (short 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. For remote-PIN verification the TOE sends/receives the PIN to/from another card terminal via the connector. This asset is user data.
Management credentials	The TOE stores credentials (e.g. passwords) to authenticate TOE administrators for management activities. The TOE has to ensure the confidentiality and integrity of these credentials. This asset is user data.
Shared secret	The TOE stores a shared secret which is generated by the connector during the initial pairing process. The shared secret and the SM-KT represent the identity of the card terminal. This identity is used for secure identification and authentication of the card terminal by the connector. The TOE has to ensure the confidentiality and integrity of the shared secret. This asset is TSF data.
Patient data	This data comprises health information and billing data that is related to patients. The TOE gets patient data from the cards in its slots, encrypts this data and sends it to the connector. Further the TOE accepts patient data from the connector, decrypts it, and sends it to the corresponding eHC in its slot. The TOE has to ensure the confidentiality and authenticity of this data. This asset is user data.
Communication data	Confidential data that is transmitted between the TOE and the connector. This data comprises at least patient data and PINs for remote-PIN verification. The TOE has to ensure the confidentiality and authenticity of this data. This asset is user data.
Configuration data	Data on which the TOE relies on for its secure operation. This data comprises at least the management credentials for local and remote management and the list of TSP CAs. The TOE has to ensure the integrity, confidentiality, and authenticity of the management credentials. It has to ensure integrity and authenticity of the list of TSP CAs. This asset is user data.

TSF data	The TOE stores TSF data which is necessary for its own operation. The TOE has to ensure the confidentiality and authenticity of this data. This asset is TSF data.

Table 1: Assets

3.2 Subjects

The following subjects are interacting with the TOE:

Subject	Description
TOE Administrator	The TOE administrator is in charge of managing the security functions 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 moderate level attack potential.
Authorized card	Authorized cards (HPC, SMC-B) are able to perform card-to-card authentication which is used for remote-PIN verification.
Card	The TOE is handling the communication for one or more smart cards in its card slots.
Connector	The connector is the only entity in the environment of the TOE (except for users of the management interface) which is foreseen to communicate with the TOE. It is the interface for the TOE to securely communicate with the telematic infrastructure of the German healthcare system.
Medical supplier	The medical supplier (e.g. a physician) uses the TOE together with his HPC (or SMC-B). With the HPC it is also possible for medical suppliers to generate qualified digital signatures. Other than the patient the medical supplier can be held responsible for the secure operation of the TOE.
Patient	The patient uses the TOE together with his eHC. The patient 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.
Push Server	The Push Server is a trusted entity in the internal network of the medical supplier which updates firmware on card terminals that are connected to that network. The Push Server uses the SICCT interface or another network interface of the card terminal for remote update. See A.PUSH_SERVER for assumptions on the Push Server.
SM-KT	The SM-KT represents the cryptographic identity of the TOE. It is a secure module that carries a X509 certificate and provides : • Protection of the private key • Cryptographic functions for encryption/decryption and signature creation • A random number generator • A function to read out the public key
TOE Reset Administrator	The TOE Reset Administrator is the only user role that is able to perform a reset of the TOE settings when management credentials are lost. The type of authentication for this role depends on the particular implementation. The TOE Reset Administrator could be the developer himself.
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.

Table 2: Subjects

3.3 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. On the other hand, the possibilities for an attacker are limited by the characteristics of the controlled environment (specifically addressed by A.ENV).

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

The assets that are threatened and the paths 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 communication 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 the transmitted data.
T.PIN	An attacker may try to release the PIN which has been entered by a user from the TOE in clear text. 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. This data may comprise: • Configuration data the TOE relies on for its secure operation • The shared secret of TOE and connector • Communication data that is received from a card and stored within the terminal before it is submitted to the connector An 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. Specifically an attacker may • use any interface that is provided by the TOE • physically probe or manipulate the TOE
T.F-CONNECTOR	Unauthorized personnel may try to initiate a pairing process with a fake connector after an unauthorized reset to factory defaults, e.g. to initiate an unauthorized firmware update or to receive confidential (patient) data.

Table 3: Threats

3.4 Organizational Security Policies

The TOE shall be implemented according to the following specifications:

Policy	Description
OSP.PIN_ENTRY	The TOE shall fulfil the requirements to be used as a secure PIN pad entry device for applications according to [22]. This specifically means that a PIN, which has been entered by a user at the TOE, must never leave the TOE in clear text, except to smart cards in local card slots. For the case that a terminal implements an insecure mode (e.g. a mode, in which it cannot be guaranteed that the PIN will not 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 4: Organisational Security Policies

3.5 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. Specifically it is assumed: • The card terminal prevents (not visible) physical manipulations for at least 10 minutes. The environment ensures beyond these 10 minutes that the card terminal is protected against unauthorized physical access or such is perceptible, • That the user handles his PIN with care; specifically that the user will keep their PIN secret, • That the user can enter the PIN in a way that nobody else can read it, • That the user only enters the card PIN when the TOE indicates a secure state, • That the medical supplier checks the sealing and the physical integrity of the TOE regularly before it is used, • That the network of the medical supplier is appropriately secured so that authorized entities are trustworthy, see also [16].
A.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. The administrator and the medical supplier shall be responsible for the secure operation of the TOE. Specifically it shall be ensured: • That they enforce the requirements on the environment (see A.ENV), • That the administrator ensures that the medical supplier received the necessary guidance documents (especially for firmware updates), • That the physical examination of the TOE is performed according to the process described by the manufacturer in the evaluation process (e.g. seal checking), • That the administrator checks the integrity of the terminal before the initial start-up procedure (every new pairing process) and the medical supplier checks the integrity of the terminal before every start-up procedure, • That they react to breaches of environmental requirements according to the process described by the manufacturer in the evaluation process (e.g. reshipment to the manufacturer).

Assumption	Description
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 [16]. Further it is assumed that for the case the TOE uses a DF.KT of a gSMC-KT as SM-KT which are addressable via the connector, the TOE accesses this DF.KT via the base-channel 0. During the use of the SM-KT by the TOE the terminal card commands of the TOE have to be given precedence and the processing of possibly existing client SICCT commands has to be interrupted and continued only after completion of the internal command sequence. The developer may queue the interrupts internally or implement error messages as answers to the commands. It is also assumed that the connector makes sure that a DF.KT of a gSMC-KT as SM-KT which is addressable via the connector can only be accessed by the TOE and cannot be used by any other system than the TOE. Further, it is assumed that the connector periodically monitors the pairing state with the TOE and provides warning mechanisms to indicate unexpected results like paired terminals which lack the shared secret.
A.SM	The TOE will use a secure module (SM-KT) that represents the cryptographic identity of the TOE in form of an X.509 certificate. 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. The random number generator of the SM-KT is assumed to provide entropy of at least 100 bit for key generation. 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 (according to TR-03120 [7]). The secure module has undergone an evaluation and certification process in compliance with the corresponding Protection Profile [15] and complies with the specification [19].
A.PUSH_SERVER	It is assumed that the internal network of the medical supplier is equipped with a so called Push Server for automatic firmware updates according to the push update mechanism described in [17]. The TOE administrator is assumed to be responsible for the operation of the Push Server and able to select the particular firmware version that the server is allowed to install on the card terminals. It is further assumed that every time an update process is performed for a card terminal the Push Server logs the following information: identifier of involved card terminal, version of firmware to install, result of the update process.
A.ID000_CARDS	It is assumed that all smartcards of form factor ID000 are properly sealed after they are brought into the TOE. Further, the developer is assumed to provide guidance documentation on how a TOE administrator could renew a sealing after an ID000 card is replaced by another one.

Table 5: Assumptions

4 Security Objectives

This chapter describes the security objectives for the TOE and the security objectives for the environment of the TOE.

The security objectives for the TOE, the security objectives for the environment and the security objectives rationale are literally identical to the respective sections of BSI-CC-PP-0032-V2-2015-MA-01.

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. Therefore the TOE shall provide an access control function based on the identity of the current user. Further the access control mechanism of the TOE has to ensure that the PIN cannot be read from the TOE. The TOE shall also ensure that the TOE administrator's credentials for local management are set before access to other TOE functionality is possible.
O.PIN_ENTRY	The TOE shall serve as a secure pin entry device for the user and the administrator. 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, except to smart cards in each addressed local card slot. For remote-PIN verification the PIN shall be encrypted, by a local gSMC-KT, controlled by the Connector, so that it can only be decrypted by the receiving smart card (HPC or SMC-B).
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. Thus the TOE has to provide a mean to identify and authenticate the current user. The TOE shall maintain at least three distinct roles: administrators, the TOE Reset Administrator, and users[4].
O.MANAGEMENT	In order to protect its configuration the TOE shall provide only an authenticated and authorized administrator with the necessary management functions. The TOE shall enforce an access control policy for management functions, as some functions shall only be accessible by administrators authenticated by the local management interface. Further, the following management functions can be used by unauthenticated users • Display the product version number of the TOE • View card terminal name for card terminal The TOE shall provide a local management interface, and management over SICCT interface.

[4] 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.

Objective	Description
	A firmware consists of two parts: (1) the so-called “firmware list” and (2) the “firmware core” which includes the whole firmware except the firmware list. Firmware lists and cores have to versioned independently. The firmware list states all firmware core versions to which a change is allowed: An update of the firmware core is only allowed if the core version is included in the firmware list. A firmware update of the TOE shall only be possible after the integrity and authenticity of the firmware has been verified and the following holds: • The TOE provides functionality to update and downgrade its firmware. This includes both the change to a newer firmware as a downgrade to a firmware which is approved with the concept of firmware-group. • The configuration, such as terminal type, IP address or pairing- information shall be preserved and indicated after a firmware update or a downgrade (see [17] for further information). • The developer of the TOE shall ensure that in case of a downgrade of the firmware the TOE must warn the Administrator (e.g. within the Guidance) before the installation that the action to be performed is not an upgrade. The TOE must offer a chance to cancel the installation. The developer- specific update component shall warn the administrator about taking the responsibility in case of performing a downgrade. The administrator shall be able to manage the list of TSP CAs which is used to verify the authenticity of connectors. An update of the TSP CA list shall only be possible after the integrity and authenticity of the list has been verified. The TOE shall ensure that for all security attributes, which can be changed by an administrator or the user, only secure values are accepted. This includes the enforcement of a password policy for the management interfaces. In addition to the developer-specific update component the TOE supports update features of the SICCT specification, whereby a trigger component is able to update the TOE (e.g. the Configuration and Software Repository- Service (KSR) of the telematic infrastructure).
O.SECURE_ CHANNEL	When establishing a connection between the TOE and the connector both parties shall 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 [17]. The TOE in each security context shall only have one connection to one connector at a time. 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. 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 Security Target realises more than just the necessary set of functionality as required by this ST. However, additional functionality that is not security functionality (e.g. value-added modules) may lead to an insecure state of the TOE as the user 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 ST is not working during its use. Thus the TOE shall be able to indicate whether it is currently in a secure state, i.e. whether all TSF as required by this ST 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 start-up and specifically verify the correct operation of the secure module (see A.SM). The TOE shall provide an adequate level of physical protection to protect the stored assets and the SM-KT[5]. It has to be ensured that any kind of physical tampering that might compromise the TOE Security Policy within 10 minutes can be afterwards detected by the medical supplier. 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. 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. The TOE shall delete • PINs, • cryptographic keys, and • all information that is received by a card in a slot of the TOE or by the connector (except the shared secret) in a secure way when it is no longer used. In case 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

[5] Please note that the SM-KT provides its own physical protection for the stored keys. However according to [17] 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 Operational Environment

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

Objective	Description
OE.ENV	It is assumed that the TOE is used in a controlled environment. Specifically it is assumed: • The card terminal prevents (not visible) physical manipulations for at least 10 minutes. The environment ensures beyond these 10 minutes that the card terminal is protected against unauthorized physical access or such is perceptible, • That the user handles his PIN with care; specifically that the user will keep their PIN secret, • That the user can enter the PIN in a way that nobody else can read it, • That the user only enters the card PIN when the TOE indicates a secure state, • That the medical supplier checks the sealing and the physical integrity of the TOE regularly before it is used, • The medical supplier sends the TOE back to the manufacturer in case he suspects an unauthorized reset to factory defaults has been performed by unauthorized personnel, and • That the network of the medical supplier is appropriately secured so authorized entities are trustworthy, see also [16].
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. The administrator and the medical supplier shall be responsible for the secure operation of the TOE. Specifically it shall be ensured: • That they enforce the requirements on the environment (see A.ENV), • That the administrator ensures that the medical supplier received the necessary guidance documents (especially for firmware updates), • That the physical examination of the TOE is performed according to the process described by the manufacturer in the evaluation process (e.g. seal checking), • That the administrator checks the integrity of the terminal before the initial start-up procedure (every new pairing process) and the medical supplier checks the integrity of the terminal before every start-up procedure, • That they react to breaches of environmental requirements according to the process described by the manufacturer (e.g. reshipment to the manufacturer), and • That the administrator checks the secure state of the TOE regularly[6].
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 [16]. Further the connector has to periodically check the pairing state with the TOE and warn the administrator accordingly.

[6] The secure state can be indicated by e.g. the pairing information with the connector, the firmware version or other security events which the developer has to define within the Guidance documentation.

Objective	Description
OE.SM	The TOE will use a secure module (SM-KT) that represents the cryptographic identity of the TOE in form of an X.509 certificate. 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. The random number generator of the SM-KT shall provide entropy of at least 100 bit for key generation. 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 (according to TR-03120 [7]). The secure module has undergone an evaluation and certification process in compliance with the corresponding Protection Profile [15] and complies with the specification [19].
OE.PUSH_SERVER	The internal network of the medical supplier is equipped with a so called Push Server for automatic firmware updates according to the push update mechanism described in [17]. The TOE administrator is responsible for the operation of the Push Server and able to select the particular firmware version that the server is allowed to install on the card terminals. Every time an update process is performed for a card terminal the push server logs the following information: identifier of involved card terminal, version of firmware to install, result of the update process.
OE.ID000_CARDS	All smartcards of form factor ID000 shall be properly sealed after they are brought into the TOE.[7] Further, the developer shall provide guidance documentation on how a TOE administrator could renew a sealing after an ID000 card is replaced by another one.

Table 7: Security Objectives for the environment of the TOE

[7] Please see TIP1-A_3192 in [17].

4.3 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	OE.PUSH_SERVER	OE.ID000_CARDS
T.COM			X		X		X	X					
T.PIN	X	X					X	X					
T.DATA	X		X	X			X	X					
T.F-CONNECTOR								X	X	X			
OSP.PIN_ENTRY		X				X	X						
A.ENV								X					
A.ADMIN									X				
A.CONNECTOR										X			
A.SM											X		
A.PUSH_SERVER												X	
A.ID000_CARDS													X

Table 8: Security Objective Rationale

4.3.1 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 since both parties have to be aware of the identity of their communication partner. *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. *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. It further protects the TOE against physical tampering for 10 minutes. *OE.ENV* finally ensures that the network of the medical supplier is appropriately secured so that it cannot be accessed by unauthorized entities and that the TOE is protected against physical tampering if the TOE is unobserved for more than 10 minutes.

Furthermore *OE.ENV* assures that the medical supplier checks the sealing and the physical integrity of the TOE regularly before it is used.

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 TOE administrator and contributes to countering T.PIN as it ensures that the PIN cannot be released from the TOE in clear text. This is the main objective that serves to remove the threat. *O.PROTECTION* contributes to countering T.PIN as it ensures that the TOE provides an adequate level of physical protection for the PIN for 10 minutes. It further protects the PIN when it is transmitted between physically separated parts, ensures that the PIN is securely deleted when it is no 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. *OE.ENV* finally ensures that that the network of the medical supplier is appropriately secured so that it cannot be accessed by unauthorized entities. The TOE is protected against physical tampering if it is unobserved for more than 10 minutes and that the medical supplier checks the sealing and the physical integrity of the TOE regularly before it is used. Furthermore *OE.ENV* contributes to countering T.PIN by ascertaining that the user enters the PIN in a way that nobody else can read it and that this can only be done when the TOE indicates a secure state.

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 decision in the context of access control. *O.MANAGEMENT* ensures that only the TOE 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. *O.PROTECTION* provides the necessary physical protection for the data stored in the TOE for 10 minutes 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). *OE.ENV* finally ensures that the network of the medical supplier is appropriately secured so that it cannot be accessed by unauthorized entities and that the TOE is protected against physical tampering if the TOE is unobserved for more than 10 minutes. Furthermore *OE.ENV* assures that the medical supplier checks the sealing and the physical integrity of the TOE regularly before it is used and that the user only enters the card PIN when the TOE indicates a secure state.

The threat **T.F-CONNECTOR,** which describes that unauthorized personnel may try to initiate a pairing process with a fake connector after an unauthorized reset to factory defaults, is countered by a combination of *OE.ENV*, *OE.ADMIN* and *OE.CONNECTOR*. *OE.ENV* ensures that the medical supplier sends the TOE back to the developer in case he suspects an unauthorized reset to factory defaults has been performed by unauthorized personnel. *OE.ADMIN* ensures that the administrator checks the secure state of the TOE regularly before it is used. *OE.CONNECTOR* ensures that the connector in the environment is trustworthy and provides the possibility to establish a Trusted Channel with the TOE including a mean for mutual authentication. It further ensures that the connector has to undergo an evaluation and certification process in compliance with the corresponding Protection Profiles. *OE.CONNECTOR* further ensures that the connector periodically checks the pairing state with the TOE and warns the administrator accordingly.

4.3.2 Covering the OSPs

The organizational security policy **OSP.PIN_ENTRY** requires 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 or not.

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 in case of a card PIN 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.PIN_ENTRY. 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.

4.3.3 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.

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

The assumption **A.ID000_CARDS** is covered by *OE.ID000_CARDS* as directly follows.

5 Extended Components Definition

This security target uses no components which are not defined in CC part 2 or CC part 3.

6 Security Requirements

This chapter defines the TOE security functional requirements and the TOE security assurance requirements.

On TOE security assurance requirements, no operations have been performed. On TOE security functional requirements, operations for assignment, selection, refinement and iteration have been completely performed. Operations not performed in BSI-CC-PP-0032-V2-2015-MA-01 are identified in order to indicate their instantiation by the ST author.

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

All operations which have been completed by the ST author are indicated by {braces} instead of [brackets].

6.1 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/Connector	Cryptographic key generation for connector communication
FCS_CKM.1/Management	Cryptographic key generation for remote management
FCS_CKM.4	Cryptographic key destruction for communication
FCS_COP.1/Con_Sym	Cryptographic operation for connector communication (symmetric algorithm)
FCS_COP.1/SIG	Cryptographic operation for signature generation/verification
FCS_COP.1/Management	Cryptographic operation for remote management
FCS_COP.1/SIG_FW	Cryptographic operation for firmware signature verification
FCS_COP.1/SIG_TSP	Cryptographic operation for signature verification of TSP CA lists
User data protection (FDP)	
FDP_ACC.1/Terminal	Subset access control for terminal functions
FDP_ACC.1/Management	Subset access control for management
FDP_ACF.1/Terminal	Security attribute based access control for terminal functions
FDP_ACF.1/Management	Security attribute based access control for management
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_AFL.1	Authentication failure handling
FIA_ATD.1	User attribute definition
FIA_SOS.1	Verification of secrets
FIA_UAU.1	Timing of authentication
FIA_UAU.5	Multiple authentication mechanisms

FIA_UAU.7	Protected authentication feedback
FIA_UID.1	Timing of identification
Security Management (FMT)	
FMT_MSA.1/Terminal	Management of security attributes for Terminal SFP
FMT_MSA.1/Management	Management of security attributes for management SFP
FMT_MSA.2	Secure security attributes
FMT_MSA.3/Terminal	Static attribute initialisation for terminal SFP
FMT_MSA.3/Management	Static attribute initialisation for management SFP
FMT_SMF.1	Specification of Management Functions
FMT_SMR.1	Security roles
Protection of the TSF (FPT)	
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_PHP.2[8]	Notification of physical attack
FPT_TST.1	TSF testing
TOE Access (FTA)	
FTA_TAB.1/SEC_STATE	Default TOE access banners for secure state
Trusted path/channels (FTP)	
FTP_ITC.1/Connector	Inter-TSF trusted channel for connector communication
FTP_TRP.1/Management	Trusted path for remote management

Table 9: Security Functional Requirements for the TOE

6.1.1 Cryptographic Support (FCS)

6.1.1.1 FCS_CKM.1/Connector Cryptographic key generation for connector communication

FCS_CKM.1.1/Connector The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm {*Ephemeral Diffie-Hellman with RSA signatures, using group 14 [RFC 3526] as DH parameters or Ephemeral ECDH with RSA signatures (TLS_EC/DHE_RSA_WITH_AES_128_CBC_SHA/256 and TLS_EC/DHE_RSA_WITH_AES_256_CBC_SHA/256/384)*} and specified cryptographic key sizes {*AES: 128 bit and 256 bit, HMAC-SHA1: 160/256/384 bit*} that meet the following: [*[17]*].

Hierarchical to: No other components.

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

Application Note 1: The cryptographic session keys, generated by FCS_CKM.1/Connector shall be used for the TLS encryption/decryption between the TOE and the connector (for further information see [17] also chapter 6.1.1.4). The generation (actually negotiation) of this key shall be done in accordance with the Diffie-Hellman protocol.
It should be noted that this negotiation includes a mutual authentication of the TOE and the connector based on certificate validation (see [17]) and validation of a shared secret. The TOE shall

[8] SFR has been added by the ST author

determine the role from the connector certificate presented during the buildup of the TLS connection. The Toe shall check that the determined role corresponds with the role "Signature Application Component (SAC)" (see [17]).
The TOE shall use the SM-KT for Random Number generation, Signature generation and Signature Verification (see also A.SM) or its own functionality required by FCS_COP.1/SIG.
{Subject to the regulations of gematik} the connection to network based management interfaces {must} be secured with { TLS Version 1.2 (see GS-A_4385 in [23]). In difference to BSI-CC-PP-0032-V2-2015-MA-01 the TOE may not support TLS Version 1.1 (see A_18464 in [23]).}

6.1.1.2 FCS_CKM.1/Management Cryptographic key generation for remote Management

FCS_CKM.1.1/Management The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm {*Ephemeral Diffie-Hellman with RSA signatures, using group 14 [RFC 3526] as DH parameters or Ephemeral ECDH with RSA signatures [RFC 4492] using elliptic curves P-256 or P-384 (TLS_DHE_RSA_WITH_AES_128_CBC_SHA, TLS_DHE_RSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256, TLS_DHE_RSA_WITH_AES_128_CBC_SHA256, TLS_DHE_RSA_WITH_AES_256_CBC_SHA256, TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 and TLS_DHE_RSA_WITH_AES_256_GCM_SHA384)*} and specified cryptographic key sizes {*AES: 128 bit and 256 bit, HMAC-SHA1: 160 bit, HMAC-SHA256: 256 bit, HMAC-SHA384: 384 bit*} that meet the following: [*[17]*].

Hierarchical to: No other components.

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

Application Note 2: The cryptographic session keys, generated by FCS_CKM.1/Management shall be used for the TLS encryption/decryption for remote management (for further information see [17] (see also chapter 6.1.1.6). The generation (actually negotiation) of this key shall be done in accordance with the TLS handshake protocol (for further information see [10]), extended and limited by [17].

The TOE should use the functionality of the SM-KT for random number generation. Note, that the SM-KT is physically integrated into the TOE in the evaluated TOE configuration.
{Subject to the regulations of gematik} the connection to network based management interfaces {must} be secured with {TLS Version 1.2 (see GS-A_4385 in [23]). In difference to BSI-CC-PP-0032-V2-2015-MA-01 the TOE may not support TLS Version 1.1 (see A_18464 in [23]).}
This SFR can implicitly be fulfilled by the mechanisms for cryptographically secured communication with the connector.

6.1.1.3 FCS_CKM.4 Cryptographic key destruction for connector communication

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method {*zeroisation, i.e. overwriting memory areas with zeros,*} that meets the following: {*none*}.

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]

6.1.1.4 FCS_COP.1/Con_Sym Cryptographic operation for connector communication (symmetric algorithm)

FCS_COP.1.1/Con_Sym The TSF shall perform {*encryption/decryption*} in accordance with a specified cryptographic algorithm {*Advanced Encryption Standard (AES) using Cipher Block Chaining (CBC) mode of operation*} and cryptographic key sizes {*128 or 256 bit*} that meet the following: [*[17]*].

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

Application Note 3: The symmetric cryptographic algorithm in FCS_COP.1/Con_Sym shall be used to set up the trusted channel with a connector (see also chapter 6.1.7.1 for the definition of the trusted channel itself).

6.1.1.5 FCS_COP.1/SIG Cryptographic operation for signature generation/verification

FCS_COP.1.1/SIG The TSF shall perform [*signature generation/verification*] in accordance with a specified cryptographic algorithm {*SHA256 with RSA using RSASSA-PKCS1-v1_5 signature scheme*} and cryptographic key sizes {*2048 bit*} that meet the following: [*[17]*].

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

Application Note 4: The algorithm for signature generation/verification in FCS_COP.1/SIG shall be used to establish the trusted channel with the connector (see also chapter 6.1.7.1 for the definition of the trusted channel itself). Serving this purpose, the TOE shall use the support of the SM-KT for signature generation (see also A.SM). Further the TOE also shall verify that the connector certificate is trusted by the TSP CA using signature verification of FCS_COP.1/SIG.

6.1.1.6 FCS_COP.1/Management Cryptographic operation for remote management

FCS_COP.1.1/Management The TSF shall perform {*encryption/decryption (CBC/GCM mode), Authenticated Encryption with Associated Data (GCM mode only)*} in accordance with a specified cryptographic algorithm {*Advanced Encryption Standard (AES) using Cipher Block Chaining (CBC) or Galois/Counter Mode (GCM) mode of operation*} and cryptographic key sizes {*128 or 256 bit*} that meet the following: [*[17]*].

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

Application Note 5: The cryptographic functionality in FCS_COP.1/Management and FCS_CKM.1/Management shall be used to establish the trusted path for remote management. See chapter 6.1.7.1 for the definition of the trusted path.
It is recommended that the cryptographic functionality in FCS_CKM.1/Management complies with the requirements of the PKCS#1 standard described in [12].
This SFR can implicitly be fulfilled by the mechanisms for cryptographically secured communication with the connector.

6.1.1.7 FCS_COP.1/SIG_FW Cryptographic operation for firmware signature verification

FCS_COP.1.1/SIG_FW The TSF shall perform [*signature verification for firmware updates*] in accordance with a specified cryptographic algorithm {*SHA256 with RSA*} and cryptographic key sizes {*4096 bit*} that meet the following: [*[17]*].

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

Application Note 6: 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. The public key must be part of the installed firmware.
It is recommended that the cryptographic functionality complies with the requirements of the PKCS#1 standard described in [12].

6.1.1.8 FCS_COP.1/SIG_TSP Cryptographic operation for verification of TSP CA lists

FCS_COP.1.1/SIG_TSP The TSF shall perform {*signature verification of TSP CA lists*} in accordance with a specified cryptographic algorithm {*SHA256 with RSA*} and cryptographic key sizes {*4096 bit*} that meet the following: [*[17]*].

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

Application Note 7: The functionality is used to verify the integrity and authenticity of a potential update of the TSP CA list. Such functionality may rely on hashing and encryption using a public key (signature verification) but could also require the interaction of an administrator (verification of hash value).
It is recommended that the cryptographic functionality complies with the requirements of the PKCS#1 standard described in [12] (if applicable).
Please also note that if the vendor choses to provide TSP CA list updates via the firmware update mechanism, this SFR is to be considered to be fulfilled accordingly.

6.1.2 User data protection (FDP)

6.1.2.1 FDP_ACC.1/Terminal Subset access control for terminal functions

FDP_ACC.1.1/Terminal The TSF shall enforce the [*Terminal SFP*] on [
Subjects: all subjects
Objects: PIN, TSP CA list, shared secret, management credentials, firmware, cryptographic keys, Communication data, {cross-certificates}
Operations: Read, modify, {no other operations among subjects and objects covered by the SFP}].

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

6.1.2.2 FDP_ACC.1/Management Subset access control for management

FDP_ACC.1.1/Management The TSF shall enforce the [Management SFP] on [
Subjects: users, {no other subjects}
Objects: manageable objects, i.e. management functions
Operations: execute].

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

6.1.2.3 FDP_ACF.1/Terminal Security attribute based access control for terminal functions

FDP_ACF.1.1/Terminal The TSF shall enforce the [*Terminal SFP*] to objects based on the following: [
Subjects: all subjects, attribute: user role[9]
Objects: PIN, TSP CA list, cross-certificates, shared secret, management credentials, firmware, cryptographic keys, attribute: firmware version, Enable/Disable the functionality of an unauthorized reset to factory defaults[10]
{no other objects and no related attributes}
].

FDP_ACF.1.2/Terminal The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [
If a firmware update is initiated, a modification of the firmware of the TOE shall only be allowed after the integrity and authenticity of the firmware has been verified according to FCS_COP.1/SIG_FW and:

- *The card terminal shall recognize non-authentic transmissions. The security anchor required for this action shall be placed in a writing-protected area of the external interfaces of the TOE.*
- *Furthermore, the security anchor shall be located in a read-only area of the device and shall only be able to be replaced with an administrative action.*
- *The transmission mechanism shall be in a position to detect transmission errors independently.*
- *An update of the firmware of the TOE shall only be allowed by an authenticated administrator:*
    - *A firmware consists of two parts: firstly the so-called “firmware list” and secondly the “firmware core” which includes the whole firmware except the firmware list. The firmware list states all firmware core versions to which a change is allowed. Firmware lists and cores have to be versioned independently.*
    - *An update of the firmware core is only allowed if the core version is included in the firmware list. Firmware lists must only contain version numbers of firmware cores which are certified according this Protection Profile. For the use in the German Healthcare System*

[9] The role of the user (e.g. medical supplier, TOE administrator)
[10] i.e. its configuration status

the named versions must also be approved by the gematik.

- *In case of downgrades of the firmware the TOE must warn the administrator before the installation that he is doing a downgrade, not an upgrade. The TOE must offer him the chance to cancel the installation.*
- *In case of a common update the TOE has to install the new firmware list at first. The new list is used to decide whether an update to the accompanying firmware core is allowed.*
- *Updates of the firmware list are only allowed to newer versions. Use higher version numbers to distinguish newer versions.*
- *Installation of firmware cores and lists are only allowed after the integrity and authenticity of the firmware has been verified using the mechanism as described in FCS_COP.1/SIG_FW.*

If a TSP CA list update is initiated, a modification of the list shall only be allowed after the integrity and authenticity of the new TSP CA list has been verified according to FCS_COP.1/SIG_TSP.

The developer of the TOE shall ensure that in case of a downgrade of the firmware the TOE must warn the Administrator (e.g. within the Guidance) before the installation that the action to be performed is not an upgrade. The TOE must offer a chance to cancel the installation. A downgrade of the TOE shall only be possible after warning the administrator about the risks of this action. This warning shall be performed by the developer-specific update component.

The following management functions shall be executable by authenticated TOE administrators (excluding SICCT interface):

- *{none}*

{no other rules}

].

FDP_ACF.1.3/Terminal The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: {*none*}

FDP_ACF.1.4/Terminal The TSF shall explicitly deny access of subjects to objects based on the **following additional rules** [

- *No subject shall access any object but the TOE administrator's local management credentials before the TOE administrator's credentials are initially set.*
- *No subject shall read out the PIN, shared secret, management credentials or secret cryptographic keys while they are temporarily stored in the TOE*
- *No subject shall modify the public key for the signature verification of firmware updates unless a new public key is part of a firmware update.*

].

Hierarchical to: No other components.

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

Application Note 8: Note that “firmware version” in FDP_ACF.1.2/Terminal could also be interpreted as a firmware group version. This allows the use of the firmware group concept described in [17] making downgrades possible.

6.1.2.4 FDP_ACF.1/Management Security attribute based access control for management

FDP_ACF.1.1/Management The TSF shall enforce the [*Management SFP*] to objects based on the following: [
Subjects: user, {no other subjects}
Subject attribute: role(s), management interface[11]*, {no other subject attributes}*
Objects: management functions,
Object attribute: none
].

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

The following management functions shall be executable by all roles:

- *Display the product version number of the TOE*
- *Manage own login credentials*
- *View card terminal name for card terminal*
- *{View the available network configuration}*
- *{generate challenge data for challenge-response mechanism and enter response if the alternative reset mechanism is enabled}*
- *{none (not applicable: Display the MAC-address(es) of the TOEs network interface(s))}*
- *{none (not applicable: Reset the TOE settings to factory defaults (unauthorized reset to factory defaults))}*
- *{no further management functions}*

The following management functions shall be executable by authenticated TOE administrators (excluding SICCT interface):

- *{Manage the available network configuration}*
- *{none (not applicable: Set card terminal name for card terminal)}*
- *{Enable/Disable remote update functionality for firmware update}*
- *Manage local and remote management login credentials*

[11] The subject attribute management interface specifies the interface from which the user is connecting (i.e. local, remote, SICCT).

- *Secure deletion of pairing information from all three possible pairing processes (initial pairing, review of pairing-information and maintenance-pairing)*
- *Manage the list of TSP Cas*
- *Perform a firmware update*
- *Reset the TOE settings to factory defaults*
- *{Enable/Disable alternative reset mechanism}*
- *{Enable/Disable administrative SICCT commands}*
- *{none (not applicable: Enable/Disable the functionality of unauthorized reset to factory defaults)}*
- *{no further management functions}*

The following management functions shall be executable by TOE administrators that were authenticated using the SICCT interface:

- *{Set card terminal name for card terminal}*
- *Perform a firmware update*

The following management functions shall be only executable by TOE administrators that were authenticated using the local management interface:

- *Enable/disable the remote management interface (if applicable)*
- *Perform the initial pairing process with the connector*
- *{Enable/Disable alternative reset mechanism}*

The TOE Reset Administrator shall only be able to execute the following management function:

- *Reset the TOE settings to factory defaults (fallback)*
- *{Reset the TOE settings to factory defaults (alternative reset mechanism)}*

{no further rules}

].

FDP_ACF.1.3/Management The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: {*Reset the TOE settings to factory defaults by the TOE Reset Administrator authenticated by entering a vaild reset code (alternative reset mechanism)*}.

FDP_ACF.1.4/Management The TSF shall explicitly deny access of subjects to objects based on the following additional rules {*none*}

Hierarchical to: No other components.

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

Application Note 9: FDP_ACF.1/Management was used to define the access control for management functionality of the TOE. It applies to all local, remote or SICCT interfaces, which are capable of management functionality.

6.1.2.5 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, connector, remote card terminal[12]
Information: PIN
Operation: Entering the PIN].

Hierarchical to: No other components.

Dependencies: FDP_IFF.1 Simple security attributes

6.1.2.6 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 identifier[13], {no 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: [
PINs shall never be stored in the non-volatile memory of the TOE.
The PIN entered by the user shall only be sent via the secure channel targeting the card in the card slot of the TOE or a remote card terminal for remote-PIN verification.
In the latter case the TOE shall assure that the connection to the connector is TLS secured.
].

FDP_IFF.1.3/PIN The TSF shall enforce the [*PIN digits shall never be displayed on the display during entry of the PIN. The TOE shall rather present asterisks as replacement for digits.*].

FDP_IFF.1.4/PIN The TSF shall explicitly authorise an information flow based on the following rules: [*none*].

FDP_IFF.1.5/PIN The TSF shall explicitly deny an information flow based on the following rules: [

- *The PIN shall never leave the TOE in clear text for remote-PIN verification.*

].

Hierarchical to: No other components.

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

Application Note 10: 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, [17] may specify more detailed requirements about the display device.

[12] A remote card terminal either sends or receives a PIN for remote-PIN verification

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

For remote-PIN verification the TOE may send the PIN to another card terminal via the connector. The PIN is then encrypted and transferred using card-to-card authentication of the smart cards in both card terminals.
Remote-PIN verification is initiated by the connector. Therefore, it is responsible to select the participating card terminals and to initiate card-to-card authentication between both.
Communication between TOE and connector is additionally secured using FCS_COP.1/Con_Sym.

6.1.2.7 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

6.1.2.8 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: Passwords, patient data, shared secret, any other information*
- *Information attribute: sent via the trusted channel, {no 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 from the connector must only be accepted if the communication path is encrypted and the Connector has been successfully authenticated*[14]
- *The TOE shall have only one connection to one connector at a time.*

].

FDP_IFF.1.3/NET The TSF shall enforce the {*none*}.

FDP_IFF.1.4/NET The TSF shall explicitly authorise an information flow based on the following rules: [
The TOE shall accept the following SICCT commands arriving at the network interface even if no pairing process is established and no valid connector certificate is presented:

- *SICCT CT INIT CT SESSION*

[14] See the trusted channel in section 6.1.7.1 and the verification in section 6.1.1.5

- *SICCT CT CLOSE CT SESSION*
- *SICCT GET STATUS*
- *SICCT SET STATUS*
- *SICCT CT DOWNLOAD INIT*
- *SICCT CT DOWNLOAD DATA*
- *SICCT CT DOWNLOAD FINISH*

The TOE shall additionally accept the following EHEALTH commands (please refer to [17]) arriving at the network interface if no pairing process is established but a valid connector certificate[15] *is presented:*

- *EHEALTH TERMINAL AUTHENTICATE*

Commands to identify the TOE in the network (service discovery) may be accepted and processed even without an encrypted or authenticated connection.

].

FDP_IFF.1.5/NET The TSF shall explicitly deny an information flow based on the following rules: [

- *Passwords for management interfaces shall never leave the TOE*
- *The shared secret shall never leave the TOE in clear text (even over trusted channel)*
- *Patient data shall not be transferred via the management interfaces*

].

Hierarchical to: No other components.

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

Application Note 11: 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 [17] 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 user that it is working in an insecure state.

Please note that as a limitation to [18] the control byte for the bits b2..b1 of the Command-To-Perform Data Object CMD DO shall not contain other values than {b2 = 1, b1 = 0} or {b2 = 1, b1 = 1}.

[15] For the steps in verifying signatures of the certificate application component see [17], Table 2.

6.1.2.9 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 shared secret), {*no other object*}*].

Hierarchical to: No other components.

Dependencies: No dependencies.

Application Note 12: The functionality, defined in FDP_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 indispensable for the operation of the TOE (e.g. a secret that may be used for an initial review or the review of pairing information as part of the authentication with the connector) may be stored permanently within the TOE.

If the TOE performs Batch Signatures, it shall use the functionality of the authorized card rather than implement its own batch signature loop. In particular, this means that the PIN shall not be stored temporarily to trigger single signature processes using the stored PIN. The PIN shall be sent to the card once only and be made unavailable immediately after the batch signing process is initiated.

6.1.3 Identification and Authentication (FIA)

6.1.3.1 FIA_AFL.1 Authentication failure handling

FIA_AFL.1.1 The TSF shall detect when [[*at least 3*]] unsuccessful authentication attempts occur related to [*management authentication excluding authentication for the TOE Reset Administrator*].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been [met, surpassed], the TSF shall [*lock the particular management interface for that account for a time period according to* Table 10 *depending on the number of consecutive unsuccessful authentication attempts*].

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

Application Note 13: The assignment in FIA_AFL.1.2 implies that each management interface shall have its own counters for unsuccessful authentication attempts.

Consecutive unsuccessful authentication attempts	Lockout time

3-6	1 minute
7-10	10 minutes
11-20	1 hour
> 20	1 day

Table 10: Lockout times

6.1.3.2 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*[16], *{none}*].

Hierarchical to: No other components.

Dependencies: No dependencies.

Application Note 14: 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.

6.1.3.3 FIA_SOS.1 Verification of secrets

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet [**the following**]:

[

Passwords for management shall

- *Have a length of at least 8 characters,*
- *Be composed of at least the following characters: "0"-"9",*
- *Not contain the User ID/logon name shall not be a part of the password for the management interface,*
- *Not be saved on programmable function keys,*
- *Not be displayed as clear text during entry,*

].

Hierarchical to: No other components.

Dependencies: No dependencies.

Application Note 15: Note that the requirements on passwords hold for all management interfaces. Passwords for management interfaces (user authentication mechanism) may be implemented separately for each management interface.

6.1.3.4 FIA_UAU.1 Timing of authentication for management

FIA_UAU.1.1 The TSF shall allow [

- *Display the product version number of the TOE*
- *{none (not applicable: Display the MAC-address(es) of the TOEs network interface(s))}*
- *{none (not applicable: Reset the TOE settings to factory defaults (unauthorized reset to factory defaults))}*
- *{View card terminal name for card terminal*

[16] The role (attribute) of the user (e.g. medical supplier, TOE administrator).

- *no other 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

6.1.3.5 FIA_UAU.5 Multiple authentication mechanisms

FIA_UAU.5.1 The TSF shall provide: [

- *A password based authentication mechanism*
- *A remote authentication mechanism using the SICCT interface*
- *An authentication mechanism for the TOE Reset Administrator*
- *{An challenge-response based authentication mechanism for the TOE Reset Administrator for the alternative reset mechanism}*

] to support user authentication.

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

- *The local authentication mechanism is used for authentication of TOE administrators for management and other users*
- *The remote authentication mechanism is used for authentication of TOE administrators for management (if applicable)*
- *The remote authentication for the SICCT interface is used for authentication of TOE administrators for management*
- *The authentication mechanism for the TOE Reset Administrator is used to authenticate the TOE Reset Administrator who alone is able to reset the TOE settings to factory defaults (fallback) when the management credentials are lost*
- *{The challenge-response based authentication mechanism for the TOE Reset Administrator is used to authenticate the TOE Reset Administrator who is able to reset the TOE settings to factory defaults (alternative reset mechanism) when the management credentials are lost}*

].

Hierarchical to: No other components.

Dependencies: No dependencies.

Application Note 16: Please note that FIA_UID.1 and FIA_UAU.1 refer to the authentication of TOE administrators, the TOE Reset Administrator and users of the TOE. According to [17] 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 user role.

The authentication mechanism for the TOE Reset Administrator could be a challenge-response-mechanism. It is important that replay attacks

are not possible. Therefore, an authentication token for a card terminal (if applicable) is either distinct from those for other card terminals or additionally protected by other means to avoid misuse.
As part of authentication a possible introduction of a secure certificate into the client can be considered. In case of the use of such a secure certificate for a management connection the developer shall describe the procedure of the authentication in the user documentation.

6.1.3.6 FIA_UAU.7 Protected authentication feedback

FIA_UAU.7.1 The TSF shall provide only [*asterisks for password characters during PIN entry*] to the user while the authentication is in progress.

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

Application Note 17: This SFR covers the management authentication feedback.

6.1.3.7 FIA_UID.1 Timing of identification

FIA_UID.1.1 The TSF shall allow [

- *Display the product version number of the TOE*
- *View card terminal name for card terminal*
- *{none (not applicable: Display the MAC-address(es) of the TOEs network interface(s))}*
- *{none (not applicable: Reset the TOE settings to factory defaults (unauthorized reset to factory defaults))}*
- *{no other TSF-mediated actions}*

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

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

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

Application Note 18: – deleted –

6.1.4 Security Management (FMT)

6.1.4.1 FMT_MSA.1/Terminal Management of security attributes for Terminal SFP

FMT_MSA.1.1/Terminal The TSF shall enforce the [*Terminal SFP*] to restrict the ability to [modify] the security attributes [*Enable/Disable the functionality of an unauthorized reset to factory defaults*[17]] to: [*authenticated TOE administrators (excluding SICCT interface)*[18]].

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

[17] i.e its configuration status

[18] i.e. the standard interface to the connector using the SICCT-Protocol

FMT_SMF.1 Specification of Management Functions

6.1.4.2 FMT_MSA.1/Management Management of security attributes for Management SFP

FMT_MSA.1.1/Management	The TSF shall enforce the [*Management SFP*] to restrict the ability to [query, modify, delete, {*no other operations*}] the security attributes [*manageable objects, i.e. all management functions*] to: [*TOE administrators*].
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

6.1.4.3 FMT_MSA.2 Secure security attributes

FMT_MSA.2.1	The TSF shall ensure that only secure values are accepted for [*role(s)*[19]].
Hierarchical to:	No other components.
Dependencies:	[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

6.1.4.4 FMT_MSA.3/Terminal Static attribute initialisation for Terminal SFP

FMT_MSA.3.1/Terminal	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/Terminal	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

6.1.4.5 FMT_MSA.3/Management Static attribute initialisation for management SFP

FMT_MSA.3.1/Management	The TSF shall enforce the [*Management SFP*] to provide [restrictive] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/Management	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
Application Note 19:	*Restrictive* specifically means that remote update functionality for firmware update and remote management functionality are disabled by default.

[19] Role(s) as defined in 6.1.4.7

6.1.4.6 FMT_SMF.1 Specification of Management Functions

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

- *Manage local and remote management login credentials*[20]
- *Perform the pairing process (initial pairing, review of pairing-information and maintenance-pairing) with the connector*
- *Secure deletion of pairing information from all three possible pairing processes*
- *Manage the list of TSP CAs*[21]
- *View/set card terminal name*[22] *for card terminal*
- *Perform a firmware update*
- *Reset the TOE settings to factory defaults*[23]
- *Reset the TOE settings to factory defaults (fallback)*[24]
- *Display the product version number of the TOE*
- *Display the installed firmware group version*
- *Return self-assessment through the user interface of the administration interface*
- *Enable/disable remote management functionality*
- *{Managing network configuration}*
- *{Enable/Disable remote update functionality for firmware update}*
- *{none (not applicable: Enable/Disable the functionality of an unauthorized reset to factory defaults.)}*
- *{none (not applicable: Choose, which reset to factory defaults mechanism (reset the TOE settings to factory defaults or unauthorized reset to factory defaults) to perform.)}*
- *{none (not applicable: Display the MAC-address(es) of the TOEs network interface(s))}*[25]

{no other relevant management functions}].

Hierarchical to: No other components.

Dependencies: No dependencies.

Application Note 20: FDP_ACF.1/Management and FDP_ACC.1/Management further define which management functions are executable for the various user roles. Please note, that relevant data like failure counters for management interfaces and the shared secret shall not be reset when the firmware is updated.

[20] On first start-up the TOE forces the administrator to specify a password for local management.

[21] Management of TSP-CAs includes the update of TSP-CA lists as described in [17] as well as a selection of a particular TSP-CA list to be used in case of multiple TSP-CA lists residing in the firmware (e.g. a separate TSP-CA list for test purposes).

[22] The card terminal name is a unique identifier for the card terminal. Note that the terminal name shall not be set using dhcp.

[23] Note that after a reset to factory defaults the TOE is supposed to be in its initial state, and the administrator's local management credentials have to be set again.

[24] The fallback solution for reset of TOE settings is necessary in case the credentials for management are lost.

[25] Another option would be to attach the MAC-address(es) to the body of the card terminal.

Note that remote update functionality for firmware update may only be implemented as a PUSH service described in [17]. This requires an update component located in the local network of the medical supplier which is under the control of the TOE administrator (see OE.PUSH_SERVER). The administrator approves and releases the firmware update that should be pushed by the update component. The update component logs card terminal identifier, the time of update, the version of the firmware to install, and the result of the update for each single update process.

6.1.4.7 FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the following roles: [*user, TOE administrator, TOE Reset Administrator {no other roles}*].

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

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

6.1.5 Protection of the TSF (FPT)

6.1.5.1 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 in the TSF occur: [*disconnection of connector*[26]*, failure during firmware update, {*

- *failure of any of the self-tests as defined in FPT_TST.1*
- *an alarm condition indicates possible tampering*
- *removal of the tray for the ID-000 smart card interfaces*

}].

Hierarchical to: No other components.

Dependencies: No dependencies

Application Note 21: – deleted –

6.1.5.2 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 22: 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.

[26] When the TLS connection to the connector is lost, the secure state is preserved by resetting all plugged smart cards.

6.1.5.3 FPT_PHP.1 Passive detection of compromise 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.

Application Note 23: FPT_PHP.1 has been augmented by FPT_PHP.2 to require an active protection mechanism against physical manipulation.
The dependency to FMT_MOF.1 required by FPT_PHP.2 has been considered.[27]

6.1.5.4 FPT_PHP.2 Notification of physical attack

FPT_PHP.2.1 The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF.

FPT_PHP.2.2 The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.

FPT_PHP.2.3 For [all TSF devices / elements which protect the

- ID-000 smart card interfaces
- keypad
- I/O circuits necessary for the internal transfer of the TSF-Data and User-Data
- Memory chips for the storage of TSF and TSF-Data

AND

the opening of the housing of the TOE is permanently monitored
], the TSF shall monitor the devices and elements and notify
[all roles] when physical tampering with the TSF's devices or TSF's elements has occurred.

Hierarchical to: FPT_PHP.1

Dependencies: No dependencies.

6.1.5.5 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: {*request of TOE administrators*}] 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: {TSF data}.

FPT_TST.1.3 The TSF shall provide authorised users with the capability to verify the integrity of {TSF}.

[27] Application note has been uniquely refined by the ST author

Hierarchical to:	No other components.
Dependencies:	No dependencies.
Application Note 24:	– deleted –

6.1.6 TOE Access

6.1.6.1 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 25:	In the context of FTA_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 26:	This SFR is used to meet O.STATE. The “secure state” refers to a mode of operation in which all TSPs of this ST are met and no additional value-added module functionality (as allowed by [17]) 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 [17] a TOE could in principle accept unencrypted communications by a third party for applications that are outside the scope of the German Healthcare System. However as long as an unencrypted connection is established the TOE cannot be considered being in a secure state. This SFR is implicitly fulfilled in case the TOE doesn’t provide any additional functionality than the functionality, required by this ST and can’t operate in an insecure state.

6.1.7 Trusted path/channels (FTP)

6.1.7.1 FTP_ITC.1/Connector Inter-TSF trusted channel for connector communication

FTP_ITC.1.1/Connector	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/Connector	The TSF shall permit [*the connector*] to initiate communication via the trusted channel.
FTP_ITC.1.3/Connector	The TSF shall initiate communication via the trusted channel for [*all communications functions used by eHealth applications*].
Hierarchical to:	No other components.
Dependencies:	No dependencies.
Application Note 27:	The SFR covers the authentication of the connector by the TOE using the connector certificate of an already paired connector. The TOE also

verifies that the connector certificate is trusted by the TSP CA using signature verification of FCS_COP.1/SIG.

The trusted channel will only be active when the TOE is in “secure state”. Otherwise it will be dropped.

There is only one connection to one connector at a time.

The TOE authenticates itself with the shared secret and the certificate of the SM-KT. It has to be ensured that no security threat arises when the SM-KT is unplugged (e.g. by dropping the TLS connection).

6.1.7.2 FTP_TRP.1/Management Trusted path for remote management

FTP_TRP.1.1/Management The TSF shall provide a communication path between itself and [remote] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from [modification or disclosure, {*no other types of integrity or confidentiality violation*}].

FTP_TRP.1.2/Management The TSF shall permit [remote users] to initiate communication via the trusted path.

FTP_TRP.1.3/Management The TSF shall require the use of the trusted path for [[*authentication of TOE administrators, remote management*]].

Hierarchical to: No other components.

Dependencies: No dependencies.

6.2 Security Assurance Requirements for the TOE

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

Assurance Class	Assurance components
ADV: Development	ADV_ARC.1 Security architecture description
	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.3 Authorisation controls
	ALC_CMS.3 Implementation representation 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.4 Vulnerability analysis**

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

6.3 Security Requirements Rationale

6.3.1 Security Functional Requirements Rationale

The SFR rationale is literally identical to the respective section of BSI-CC-PP-0032-V2-2015-MA-01.

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/Connector					X		
FCS_CKM.1/Management				X			
FCS_CKM.4				X	X		X
FCS_COP.1/Con_Sym					X		
FCS_COP.1/SIG					X		
FCS_COP.1/Management				X			
FCS_COP.1/SIG_FW				X			
FCS_COP.1/SIG_TSP				X			
FDP_ACC.1/Terminal	X	X		X			
FDP_ACC.1/Management				X			
FDP_ACF.1/Terminal	X	X		X			
FSP_ACF.1/Management				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_AFL.1			X				
FIA_ATD.1			X				
FIA_SOS.1				X			
FIA_UAU.1			X				
FIA_UAU.5			X				
FIA_UAU.7		X					
FIA_UID.1			X				
FMT_MSA.1/Terminal	X			X			
FMT_MSA.1/Management				X			
FMT_MSA.2				X	X		
FMT_MSA.3/Terminal	X			X			
FMT_MSA.3/Management				X			
FMT_SMF.1				X			
FMT_SMR.1			X				
FPT_TST.1							X
FPT_FLS.1							X
FPT_ITT.1							X
FPT_PHP.1							X
FPT_PHP.2							X
FTA_TAB.1/SEC_STATE						X	
FTP_ITC.1/Connector					X		
FTP_TRP.1/Management				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/Terminal, FDP_ACF.1/Terminal, FMT_MSA.1/Terminal* and *FMT_MSA.3/Terminal*. *FDP_ACC.1/Terminal* defines the access control policy for the terminal and *FDP_ACF.1/Terminal* defines the rules for the access control policy. It is specifically defined in *FDP_ACF.1/Terminal* that nobody must be allowed to read out the PIN or private cryptographic keys from the terminal. *FMT_MSA.1/Terminal* defines, who will be allowed to manage the attributes for the access control policy while *FMT_MSA.3/Terminal* 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/Terminal, FDP_ACF.1/Terminal, FDP_IFC.1/PIN, FDP_IFF.1/PIN,* and *FIA_UAU.7.* As part of the access control policy of the terminal *FDP_ACC.1/Terminal* and *FDP_ACF.1/Terminal* define that nobody must be able to read out the PIN from the terminal, which is required by O.PIN_ENTRY. *FDP_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, *FIA_UAU.7* requires that the PIN digits are presented as asterisks on the display.

The Security Objective **O.I&A** is met by a combination of *FIA_AFL.1, FIA_ATD.1, FIA_UAU.1, FIA_UAU.5, FIA_UID.1 and FMT_SMR.1*. *FIA_AFL.1* requires that the password policy is enforced. *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, TOE administrator and TOE Reset 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, in particular for local and remote management.

The Security Objective **O.MANAGEMENT** is met by a combination of *FCS_CKM.1/Management, FCS_CKM.4, FCS_COP.1/Management, FCS_COP.1/SIG_FW, FCS_COP.1/SIG_TSP, FDP_ACC.1/Terminal, FDP_ACF.1/Terminal, FDP_ACC.1/Management, FDP_ACF.1/Management, FIA_SOS.1, FMT_MSA.1/Terminal, FMT_MSA.1/Management, FMT_MSA.2, FMT_MSA.3/Terminal, FMT_MSA.3/Management, FMT_SMF.1, and FTP_TRP.1/Management. FCS_CKM.1/Management* requires that adequate keys are generated for remote management communication. *FCS_CKM.4* requires that keys are adequately destroyed. *FCS_COP.1/Management* requires that remote management shall enforce TLS. *FCS_COP.1/SIG_FW* is used to define the mechanism to check the authenticity of a firmware update. *FCS_COP.1/SIG_TSP* is used to define the mechanism to check the authenticity of a TSP CA list update. The access control policy defined in *FDP_ACC.1/Terminal* and *FDP_ACF.1/Terminal* define the rules under which a firmware update is possible. *FDP_ACC.1/Management* and *FDP_ACF.1/Management* define the access control policy that determines under what circumstance a particular management function is accessible and by whom. *FIA_SOS.1* defines the password policy for management credentials. FMT_MSA.1/Terminal and FMT_MSA.1/Management define, 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. *FMT_MSA.3/Terminal* defines that the terminal has to provide restrictive default values for the terminal access control policy attributes. *FMT_MSA.3/Management* defines that the terminal has to provide restrictive default values for the management access control policy attributes. *FMT_SMF.1* describes the minimum set of management functionality, which has to be available according to the Security Objective. Finally, FTP_TRP.1/Management defines the trusted path between the TOE and the management client.

The Security Objective **O.SECURE_CHANNEL** is met by a combination of the SFR *FCS_CKM.1/Connector, FCS_CKM.4, FCS_COP.1/Con_Sym, FCS_COP.1/SIG, FDP_IFF.1/NET and FDP_IFC.1/NET., FMT_MSA.2*, and *FTP_ITC.1/Connector. FCS_CKM.1/Connector, FCS_COP.1/Con_Sym, and FCS_COP.1/SIG* define the cryptographic operations, which are necessary for this objective. *FCS_CKM.1/Connector* 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_CKM.4* defines the functionality to securely destroy cryptographic keys. 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. *FMT_MSA.2* defines that only secure values shall be used for security attributes. Finally *FTP_ITC.1/Connector* 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_PHP.2, FPT_FLS.1 and FPT_TST.1.*

FCS_CKM.4 defines that cryptographic keys have to be securely deleted when they are no 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 no 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* and *FPT_PHP.2* builds the physical protection for the stored assets. *FPT_TST.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 start-up and during the normal operation of the TOE.

6.3.2 SFR Dependency Rationale

The SFR dependency rationale is literally identical to the respective section of BSI-CC-PP-0032-V2-2015-MA-01.

SFR	Dependencies	Support of the Dependencies
FCS_CKM.1 /Connector	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	Fulfilled by the use of FCS_CKM.4
FCS_CKM.1 /Management	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	Fulfilled by the use of FCS_COP.1/Management and FCS_CKM.4
FCS_CKM.4	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]	Fulfilled by the use of FCS_CKM.1/Connector FCS_CKM.1/Management
FCS_COP.1 /Con_Sym	[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	Fulfilled by the use of FCS_CKM.1/Connector and FCS_CKM.4
FCS_COP.1/SIG	[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	Fulfilled by the use of FCS_CKM.1/Connector and FCS_CKM.4
FCS_COP.1 /Management	[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	Fulfilled by the use of FCS_CKM.1/Management and FCS_CKM.4

SFR	Dependencies	Support of the Dependencies
FCS_COP.1 /SIG_FW	[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	See chapter 6.3.2.1 for FDP_ITC.1 and FCS_CKM.4
FCS_COP.1 /SIG_TSP	[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	See chapter 6.3.2.1 for FDP_ITC.1 and FCS_CKM.4
FDP_ACC.1 /Terminal	FDP_ACF.1 Security attribute based access control	Fulfilled by FDP_ACF.1/Terminal
FDP_ACC.1 /Management	FDP_ACF.1 Security attribute based access control	Fulfilled by FDP_ACF.1/Management
FDP_ACF.1 /Terminal	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization	Fulfilled by FDP_ACC.1/Terminal and FMT_MSA.3/Terminal
FDP_ACF.1 /Management	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization	Fulfilled by FDP_ACC.1/Management and FMT_MSA.3/Management
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 FMT_MSA.3 Static attribute initialisation	Fulfilled by FDP_IFC.1/PIN See chapter 6.3.2.1 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 FMT_MSA.3 Static attribute initialisation	Fulfilled by FDP_IFC.1/NET See chapter 6.3.2.1 for FMT_MSA.3
FDP_RIP.1	No dependencies	-
FIA_AFL.1	FIA_UAU.1 Timing of authentication	Fulfilled by FIA_UAU.1
FIA_ATD.1	No dependencies	-
FIA_SOS.1	No dependencies	-
FIA_UAU.1	FIA_UID.1 Timing of identification	Fulfilled by FIA_UID.1
FIA_UAU.5	No dependencies	-
FIA_UAU.7	FIA_UID.1 Timing of identification	Fulfilled by FIA_UID.1

SFR	Dependencies	Support of the Dependencies
FIA_UID.1	No dependencies	-
FMT_MSA.1 /Terminal	[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	Fulfilled by FDP_ACC.1/Terminal, FMT_SMR.1 and FMT_SMF.1
FMT_MSA.1 /Management	[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	Fulfilled by FDP_ACC.1/Management, FMT_SMR.1 and FMT_SMF.1
FMT_MSA.2	[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	Fulfilled by FDP_ACC.1/Terminal, FDP_ACC.1/Management FDP_IFC.1/PIN, FDP_IFC.1/NET, FMT_MSA.1/Terminal, and FMT_SMR.1
FMT_MSA.3 /Terminal	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	Fulfilled by FMT_MSA.1/Terminal and FMT_SMR.1
FMT_MSA.3 /Management	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	Fulfilled by FMT_MSA.1/Management and FMT_SMR.1
FMT_SMF.1	No dependencies	-
FMT_SMR.1	FIA_UID.1 Timing of identification	Fulfilled by FIA_UID.1
FPT_TST.1	No dependencies	-
FPT_FLS.1	No dependencies	-
FPT_ITT.1	No dependencies	-
FPT_PHP.1	No dependencies	-
FPT_PHP.2	No dependencies	See chapter 6.3.2.1 for the dependency to FMT_MOF.1
FTA_TAB.1 /SEC_STATE	No dependencies	-
FTP_ITC.1 /Connector	No dependencies	-
FTP_TRP.1 /Management	No dependencies	-

Table 12: Dependencies of the SFR for the TOE

6.3.2.1 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 initialisation of their security attributes.

The dependencies FDP_ITC.1 and FMT_MSA.2 of FCS_COP.1/SIG_FW and FCS_COP.1/SIG_TSP 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 ST FCS_COP.1/SIG_FW and FCS_COP.1/SIG_TSP are used for a requirement on signature verification for which no secret key is necessary these dependencies do not need to be considered.

The dependency of FPT_PHP.2 to FMT_MOF.1 was considered to be not applicable because the TSF supporting FPT_PHP.2 does not provide the ability to manage security functions behaviour.

6.3.3 Security Assurance Requirements Rationale

The Evaluation Assurance Level for this Security Target is **EAL 3 augmented by AVA_VAN.4 (and consequently with its dependencies ADV_FSP.4, ADV_IMP.1, ADV_TDS.3 and ALC_TAT.1)**.

The reason for choosing these SARs is the strict conformance to BSI-CC-PP-0032-V2-2015-MA-01.

The main decision about the Evaluation Assurance Level has been taken:

- based on the fact that the TOE described in this Security Target shall serve as a secure PIN entry device (see also OSP.PIN_ENTRY), and
- based on the fact that the TOE is used in a controlled environment but also needs to provide an adequate level of protection for its assets.

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

- AVA_VAN.4

These components have the following direct and indirect dependencies, which have to be satisfied within the evaluation:

- ADV_FSP.4
- ADV_TDS.3
- ADV_IMP.1
- ALC_TAT.1 (required by ADV_IMP.1)

6.3.4 Security Requirements – Mutual Support and Internal Consistency

The core TOE functionality in this ST 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 ST 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). By this way the SFRs in this ST 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 CC part 2 [2] are used to fulfil the security objectives.

7 TOE summary specification

The TOE provides the following security functions (SF).

- Secure update of TSP CA list, firmware list and/or firmware core (SF.SECDOWN)
- Protection against physical manipulation (SF.DRILLSEC)
- Self-tests (SF.SELFTEST)
- User identification and authentication (SF.I&A)
- Residual information protection (SF.CLRMEM)
- Management functions (SF.MNGT)
- Protected PIN entry (SF.PINCMD)
- Protected data exchange between the TOE and a connector (SF.TRUSTCH)
- Protected data exchange between the TOE and a remote TOE administrator (SF.ADMCH)

FMT_MSA.1/Terminal is implicitly fulfilled since the TOE does not provide an unauthorized reset to factory defaults (see FDP_ACF.1/Terminal/Management and FMT_SMF.1).

In accordance with application note 22, FPT_ITT.1 is implicitly fulfilled since the TOE does not comprise physically separated parts.

In accordance with application note 26, FTA_TAB.1/SEC_STATE is implicitly fulfilled since the TOE does not provide any additional functionality than the functionality required by this ST and can't operate in an insecure state.

7.1 SF.SECDOWN

An update of TSP CA list[28], firmware list and/or firmware core can be performed either remotely or locally.

A remote update is initiated by the TOE administrator (SICCT) after successful identification and authentication (SF.I&A) via corresponding SICCT commands [18] presented at the SICCT management interface. These commands are accepted even if no pairing process is established and no valid connector certificate is presented (FDP_IFC.1/NET, FDP_IFF.1/NET). Before actually starting the update process, the following preparative steps are performed:

- Reading the update file and associated signature from the SICCT management interface (non-authentic transmissions or transmission errors are detected).
- Computing the SHA-256 digest of the update file and verifying the RSASSA-PKCS-v1_5 signature (FCS_COP.1/SIG_FW, FCS_COP.1/SIG_TSP) using a pre-installed public 4096 bit RSA key.
- Checking the rules for allowed update operations of firmware list, firmware core or a combination of both (FDP_ACC.1/Terminal, FDP_ACF.1/Terminal, FMT_MSA.3/Terminal).

In case of any failure of one the above steps (including non-authentic transmission or transmission errors), the update process is terminated (FDP_ACF.1/Terminal). Otherwise, the content of the update file is persistently installed by replacing the existing TSP CA list, firmware list and/or firmware core.

[28] A TSP CA list update also contains the cross-certificates for secure messaging.

After termination of the update process, the TOE is restarted. The self-test during startup (SF.SELFTEST) recognizes any incomplete installation and initiates a local update process (see below).

A local update is initiated by the TOE administrator (local) after successful identification and authentication (SF.I&A) via selecting the corresponding management function (SF.MNGT). Before actually starting the update process, the following preparative steps are performed:

- Reading the update file and associated signature from an attached USB storage device to internal volatile storage.
- Computing the SHA-256 digest of the update file and verifying the RSASSA-PKCS-v1_5 signature (FCS_COP.1/SIG_FW, FCS_COP.1/SIG_TSP) using a pre-installed public 4096 bit RSA key.
- Checking the rules for allowed update operations (FDP_ACC.1/Terminal, FDP_ACF.1/Terminal, FMT_MSA.3/Terminal).
- Upon recognizing a firmware downgrade (FDP_ACF.1/Terminal):
  Displaying a warning for the TOE administrator and offering a chance to cancel the installation.

In case of any failure of one the above steps (including cancellation of a firmware downgrade), the update process is terminated (FDP_ACF.1/Terminal). Otherwise, the content of the update file is persistently installed by replacing the existing TSP CA list, firmware list and/or firmware core.

After termination of the update process, the TOE is restarted. The self-test during startup (SF.SELFTEST) recognizes any incomplete installation and initiates another local update process. In such a case, a cancellation of the update is impossible (FPT_FLS.1), i.e. it loops until the update with some update file is successfully completed.

7.2 SF.DRILLSEC

For active protection against tampering the TOE has an alarm function.

Automatic notification in case of physical tampering (FPT_PHP.2) is ensured by permanently monitoring the opening of the housing, drilling membrane, the keypad membrane and the closing detector of both ID-000 card slots when the TOE is in the operation state “powered”. If any of these alarm conditions is recognized (indicating possible tampering), an appropriate message is displayed and the TOE will be put in a secure inoperative state (FPT_FLS.1). Any alarm condition is permanently stored and remains after a restart of the TOE.

The opening of the housing of the TOE is permanently monitored (in both operating states: “powered” and “Power loss”) by a seperated battery powered microcontroller. If the housing is opened this alarm condition will be permanently stored in this microcontroller and will be notificated when the TOE regains the operating state “powered” (see above). Thus the TOE nearly meets the requirements of FPT_PHP.3.

Protection of physical tampering is provided by attaching seals to the housing which will be visibly destroyed on attempts to tamper with the housing (FPT_PHP.1). The checking procedure of the sealing is described in the guidance documentation.

7.3 SF.SELFTEST

During startup a self-test is performed (FPT_TST.1). It is also initiated by request of the TOE administrator.

In a first phase it comprises some tests of the internal memory and the external interfaces (ethernet, chip cards, display/keypad). In case of any failure the TOE is inoperative (FPT_FLS.1).

In a second phase, the function SF.DRILLSEC is started, cryptographic selftests are performed and the completion of a previous update process is checked. Upon detection of an incomplete update, the TOE administrator (local) is identified and authenticated (SF.I&A) and another local update is initiated (SF.SECDOWN).

In a third phase, the integrity of the TSF and the TSF data is verified. In case of failure the TOE is inoperative (FPT_FLS.1).

Finally, the ability to establish a trusted communication channel (SF.TRUSTCH and SF.ADMINCH) is tested. In case of failure the TOE is inoperative (FPT_FLS.1).

7.4 SF.I&A

This function provides identification and authentication of users before any TSF-mediated action, except display the product version number of the TOE and view card terminal name for card terminal (FIA_ATD.1, FIA_UAU.1, FIA_UAU.5, FIA_UID.1). It is based on the interactions at the user interfaces (FMT_MSA.3/Management):

- Keypad/display for local users without authentication, TOE administrators (local) with local PIN authentication or challenge-response authentication (alternative reset mechanism) and TOE reset administrators with local PUK authentication (on first start-up the TOE forces the local TOE administrator to specify a PIN and a PUK for the local management interface);
- SICCT management interface for TOE administrators (SICCT) with SICCT password authentication (on enabling the interface the TOE forces the local TOE administrator to specify a password);
- Remote management interface for TOE administrators (remote) with remote password authentication (on enabling the interface the TOE forces the local TOE administrator to specify a password).

Users are associated with one of the roles ‘user’, ‘TOE administrator’ or ‘TOE Reset Administrator’ (FMT_SMR.1). The role association ensures that the roles ‘TOE administrator’ or ‘TOE Reset Administrator’ are associated only after successful authentication (FMT_MSA.2, FMT_MSA.3/Terminal/Management)

It is ensured that PINs/PUKs/passwords for each user role meet their specification (FIA_SOS.1) during entry/reception, verification and change (SF.MNGT). During PIN/PUK entry (keypad), only asterisks are displayed as feedback to the user (FIA_UAU.7). The entered PIN/PUK is zeroized after processing (SF.CLRMEM). Authentication failures are handled as required by FIA_AFL.1.

7.5 SF.CLRMEM

The information content of specific ressources is made unavailable, i.e. destroyed, when it is deallocated (FCS_CKM.4, FDP_RIP.1):

- PINs/PUKs/passwords that are entered/received for the purposes of SF.I&A or SF.PINCMD are zeroized when their processing is finished.
- Cryptographic keys that are related to SF.TRUSTCH or SF.ADMINCH are zeroized when the communication terminates, i.e. when
    - the communication channel/path is disconnected,
    - physical tampering is recognized (SF.DRILLSEC), or
    - the TOE administrator (local) is successfully authenticated (SF.I&A).
- All information that is received by a card in a slot of the TOE or by the connector (except the shared secret) is zeroized after transmission.
- The shared secret is zeroized upon request by SF.DRILLSEC or SF.MNGT.

7.6 SF.MNGT

This function provides all management functions as required by FMT_SMF.1. With some exceptions, the execution of the functions is restricted to authenticated users (SF.I&A). It is further restricted by the rules of the Management SFP (FDP_ACC.1/Management, FDP_ACF.1/Management, FMT_MSA.1/Management, FMT_MSA.3/Management).

The following management functions are available at the local management interface and are executable by the user role (i.e. without authentication):

- Display the product version number of the TOE
- View card terminal name for card terminal

The following management functions are available at the local management interface and are executable by the authenticated TOE Reset Administrator:

- Reset the TOE settings to factory defaults (fallback – necessary in case the credentials for local/remote management are lost)

The following management functions are available at the local management interface and are executable by the authenticated TOE administrator:

- Display the product version number of the TOE
- View card terminal name for card terminal
- Return self-assessment through the local management interface
- Display the installed firmware group version in the context of the self-assessment
- Manage login credentials of the TOE administrator for local management and SICCT interface
- View/Manage the available network configuration
- Enable/Disable management at the SICCT interface including remote update functionality for firmware update (disabled by default)
- Secure deletion of pairing information from all three possible pairing processes (initial pairing, review of pairing information and maintenance pairing)

- Perform a TSP CA list update[29]
- Perform a firmware update (SF.SECDOWN)
- Reset the TOE settings to factory defaults (after a reset to factory defaults the TOE is in its initial state, and the local TOE administrator login credentials have to be set again)
- Enable/disable the remote management interface (disabled by default)
- Perform the initial pairing process with the connector
- Enable/Disable alternative reset mechanism

If enabled, the following management functions are available at the remote management interface (see SF.ADMINCH) and are executable by the authenticated TOE administrator:

- Display the product version number of the TOE
- View card terminal name for card terminal
- Manage login credentials of the TOE administrator for remote management and SICCT interface
- View/Manage the available network configuration
- Secure deletion of pairing information from all three possible pairing processes (initial pairing, review of pairing information and maintenance pairing)

The following management functions are available at the SICCT interface (see SF.TRUSTCH) and are executable by the authenticated TOE administrator:

- Set card terminal name (a unique identifier) for card terminal
- If enabled, perform a remote SICCT firmware update (SF.SECDOWN)

The following management functions are executed in the context of other functionality, i.e. they are not directly available at any management interface:

- review of pairing information and maintenance pairing (SF.TRUSTCH)

7.7 SF.PINCMD

Upon receiving one of the following PIN commands via SICCT [18] the secure PIN entry mode is initiated:

INS-Byte	Command name	Description	Standard
20	VERIFY	PIN verification	ISO/IEC 7816-4
24	CHANGE REFERENCE DATA	PIN change	ISO/IEC 7816-4
26	DISABLE VERIFICATION REQUIREMENT	PIN deactivation	ISO/IEC 7816-8
28	ENABLE VERIFICATION REQUIREMENT	PIN activation	ISO/IEC 7816-8
2A	PERFORM SECURITY OPERATION	crypto operation	ISO/IEC 7816-8
2C	RESET RETRY COUNTER	PIN reset	ISO/IEC 7816-4

PIN entry and processing of the PIN command is performed in accordance with the PIN SFP (FDP_IFC.1/PIN and FDP_IFF.1/PIN). The secure PIN entry mode and the card/slot to which the PIN command is transmitted are indicated by appropriate display symbols. During PIN entry (keypad), only asterisks are displayed as

[29] A TSP CA list update also contains the cross-certificates for secure messaging.

feedback to the user. The PIN command is encrypted by secure messaging[30] before it is sent to the target card, if applicable. The entered PIN and the PIN command are zeroized after transmission (SF.CLRMEM).

7.8 SF.TRUSTCH

This function provides a protected communication between the TOE and a connector (FTP_ITC.1/Connector) via LAN interface. The channel is established based on TLS 1.2 [11] that is restricted to the following cipher suites:

- TLS_DHE_RSA_WITH_AES_128_CBC_SHA (TLS ID = {0x00, 0x33})
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA (TLS ID = {0x00, 0x39})
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 (TLS ID = {0x00, 0x67})
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 (TLS ID = {0x00, 0x6b})
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (TLS ID = {0xc0, 0x13})
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (TLS ID = {0xc0, 0x14})
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (TLS ID = {0xc0, 0x27})
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (TLS ID = {0xc0, 0x28})

The Diffie-Hellman key exchange (DHE) is realized with parameters of group 14 [14] and 2048 bit modulus length. Key generation (FCS_CKM.1/Connector) and signature of the DH parameters (FCS_COP.1/SIG) is performed by using functionality of the SM-KT in one of the SIM slots, as appropriate. The function verifies the authenticity of the connector by checking the validity of its certificate using the TSP CA list (FCS_COP.1/SIG). It supports the verification of its own authenticity by providing the certificate of the SM-KT and the pairing secret. Further, it supports initial, review and maintenance pairing (FMT_SMF.1). When the TLS connection to the connector is lost, a secure state is preserved by resetting all plugged smart cards (FPT_FLS.1).

After successful key exchange, every transmitted/received message is encrypted/decrypted using the session key according to the above mentioned cipher suites (FCS_COP.1/Con_Sym). In case of an authentic connection (valid certificate) with a paired connector (shared secret) all SICCT commands [18] and changed/extended commands according to [17] are accepted, processed and answered (FDP_IFC.1/NET and FDP_IFF.1/NET). Explicit authorization of some commands is given as described below (FDP_IFF.1/NET).

In case of an authentic connection (valid certificate) with an unpaired connector (no shared secret) the following commands (cf. [17], section 3.7.2) are accepted, processed and answered:

- EHEALTH TERMINAL AUTHENTICATE (initial/review/maintenance pairing)

In case of a non-authentic connection (invalid certificate) with an unpaired connector (no shared secret) the following SICCT commands [18] are accepted, processed and answered:

- SICCT INIT CT SESSION
- SICCT CLOSE CT SESSION
- SICCT GET STATUS

[30] Cross-certificates are required for the implementation of secure messaging. These cross-certificates are handled in the same way as the TSP CA list in all aspects of the TOE.

- SICCT SET STATUS
- SICCT CT DOWNLOAD INIT
- SICCT CT DOWNLOAD DATA
- SICCT CT DOWNLOAD FINISH

Commands to identify the TOE in the network according to Simple Service Discovery Protocol (SSDP) and Internet Control Message Protocol (ICMP) are accepted and processed even without an encrypted or authenticated connection.

As a limitation to [18] the control byte for the bits b2..b1 of the Command-To-Perform Data Object CMD DO shall not contain other values than {b2 = 1, b1 = 0} or {b2 = 1, b1 = 1}, i.e. other values are not accepted.

7.9 SF.ADMINCH

This function provides a protected communication between the TOE and a remote TOE administrator (FTP_TRP.1/Management) via LAN interface. The channel is established based on TLS 1.2 [11] that is restricted to the following cipher suites:

- TLS_DHE_RSA_WITH_AES_128_CBC_SHA (TLS ID = {0x00, 0x33})
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA (TLS ID = {0x00, 0x39})
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (TLS ID = {0xC0, 0x14})
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (TLS ID = {0xC0x, 0x13})
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (TLS ID = {0xC0, 0x30})
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (TLS ID = {0xC0, 0x2F})
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (TLS ID = {0xC0, 0x28})
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (TLS ID = {0xC0, 0x27})
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 (TLS ID = {0x00, 0x67})
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 (TLS ID = {0x00, 0x6B})
- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 (TLS ID = {0x00, 0x9E})
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (TLS ID = {0x00, 0x9F})

The Diffie-Hellman key exchange (DHE) is realized with parameters of group 14 [14] and 2048 bit modulus length. The Ephemeral Elliptic Curve Deffie-Hellman key exchange (ECDHE) uses the elliptic curves P-256 and P-384 [FIPS-186-4]. Key generation (FCS_CKM.1/Management) and signature of the DH and ECDHE parameters (FCS_COP.1/SIG) is performed by using functionality of the SM-KT in one of the SIM slots, as appropriate. The function supports the verification of its own authenticity by providing the certificate of the SM-KT. The authenticity of the remote TOE administrator is verified by SF.I&A.

After successful key exchange, every transmitted/received message, i.e. http request/response, is encrypted/decrypted using the session key according to the above mentioned cipher suites (FCS_COP.1/Management). In case of an authentic connection (SF.I&A) with a remote TOE administrator http requests for specific management functions (SF.MNGT) are accepted, processed and answered.

8 Abbreviations

Abbreviation	Denotation
AES	Advanced Encryption Standard
BSI	Bundesamt für Sicherheit in der Informationstechnik
CA	Certification Authority
DF.KT	Dedicated File Kartenterminal
eHC	Electronic Health Card
gSMC-KT	Gerätespezifisches Security Module Card Type Kartenterminal
HPC	Health Professional Card
KSR	Configuration and Software Repository (Service of the telematic infrastructure)
LAN	Local Area Network
PIN	Personal Identification Number
PP	Protection Profile
PUK	Personal Unblocking Key
SAC	Signature Application Component
SFP	Security Function Policy
SFR	Security Functional Requirement
SICCT	Secure Interoperable ChipCard Terminal
SM-KT	Security Module Kartenterminal
SMC-B	Security Module Card Typ B
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSP	Trust-Service Provider that issues connector certificates
VAM	Value-added module

9 References

Key	Label	Reference
[1]	[CC-Part1]	***Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model*** CCMB-2017-04-001, Version: 3.1, Revision 5, April 2017.
[2]	[CC-Part2]	***Common Criteria for Information Technology Security Evaluation – Part 2: Security functional requirements*** CCMB-2017-04-002, Version: 3.1, Revision 5, April 2017.
[3]	[CC-Part3]	***Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance requirements*** CCMB-2017-04-003, Version: 3.1, Revision 5, April 2017.
[4]	[CEM]	***Common Methodology for Information Technology Security Evaluation – Evaluation methodology*** CCMB-2017-04-004, Version 3.1, Revision 5, April 2017.
[6]	[BSI-TR-03116]	Bundesamt für Sicherheit in der Informationstechnik (BSI) ***Kryptographische Vorgaben für Projekte der Bundesregierung*** BSI TR-03116, Version 3.20, 21.09.2018.
[7]	[BSI-TR-03120]	Bundesamt für Sicherheit in der Informationstechnik (BSI) ***Technische Richtlinie „Sichere Kartenterminalidentität (Betriebskonzept)“*** BSI TR-03120, Version 1.1, 09.07.2010
[9]	[RFC-3268]	***Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security (TLS)*** RFC 3268, June 2002. Obsoleted by RFC 5246. http://www.ietf.org/rfc/rfc3268.txt
[10]	[RFC-4346]	***The Transport Layer Security (TLS) Protocol Version 1.1*** RFC 4346, April 2006. Obsoleted by RFC 5246. http://www.ietf.org/rfc/rfc4346.txt
[11]	[RFC-5246]	***The Transport Layer Security (TLS) Protocol Version 1.2*** RFC 5246, August 2008. http://www.ietf.org/rfc/rfc5246.txt
[12]	[RFC-3447]	***Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1*** RFC 3447, February 2003. http://www.ietf.org/rfc/rfc3447.txt
[13]	[RFC-3546]	***Transport Layer Security (TLS) Extensions*** RFC 3546, June 2003. Obsoleted by RFC 5246. http://www.ietf.org/rfc/rfc3546.txt
[14]	[RFC-3526]	***More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE)*** RFC 3526, May 2003. http://www.ietf.org/rfc/rfc3526.txt

Key	Label	Reference
[15]	[BSI-CC-PP-0082]	Bundesamt für Sicherheit in der Informationstechnik (BSI) ***Common Criteria Protection Profile*** ***Card Operating System Generation 2 (PP COS G2)*** BSI-CC-PP-0082-V3, Version 2.0, 19. Juni 2018. BSI-CC-PP-0082-V2, Version 1.9, November 2014. BSI-CC-PP-0082, Version 1.0, August 2013.
[16]	[BSI-CC-PP-0046]	Bundesamt für Sicherheit in der Informationstechnik (BSI) Common Criteria Protection Profile – Schutzprofil 2: Anforderungen an den Konnektor Online-Rollout Stufe 1 BSI-CC-PP-0046
[17]	[gemSpec_KT]	gematik GmbH ***Spezifikation eHealth-Kartenterminal*** Version: 3.11.0, Stand: 02.10.2019.
[18]	[SICCT]	TeleTrusT Deutschland e.V. ***SICCT – Secure Interoperable ChipCard Terminal*** Version 1.21, 17.12.2010.
[19]	[gemSpec_ gSMC-KT_ObjSys]	gematik GmbH ***Spezifikation der gSMC-KT – Objektsystem*** Version: 3.9.0, Stand: 24.08.2016.
[20]	[gemSpec_OM]	gematik GmbH ***Übergreifende Spezifikation*** ***Operations und Maintenance*** Version: 1.12.0, Stand: 15.05.2019.
[21]	[gemSpec_KSR]	gematik GmbH ***Spezifikation Konfigurationsdienst*** Version: 2.4.0, Stand: 02.10.2019.
[22]	[gemKPT_ Arch_TIP]	gematik GmbH ***Konzept Architektur der TI-Plattform*** Version: 2.9.0, Stand: 02.10.2019
[23]	[gemSpec_Krypt]	Gematik GmbH ***Übergreifende Spezifikation*** ***Verwendung kryptographischer Algorithmen in der Telematikinfrastruktur*** Version: 2.15.0, Stand: 02.10.2019