Document Administration

Recipient

Department	Name

For the attention of

Department	Name

Summary

The following document comprises the ST-Lite for a TOE evaluated according to Common Criteria Version 2.1. The TOE being subject of the evaluation is the product

Tachograph Card Version 1.0 128/64 R1.0

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

Keywords

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

Responsibility for updating the document

Dr. Susanne Pingel

ORGA Kartensysteme GmbH

Tachograph Card Version 1.0 128/64 R1.0

ST-Lite

Document Id:	3TachoEval.CSL.0001
Archive:	3
Product/project/subject:	TachoEval (Evaluierung Tachograph Card gemäß CC EAL4+)
Category of document:	CSL (ST-Lite)
Consecutive number:	0001
Version:	V1.00
Date:	20 August 2003
Author:	Dr. Susanne Pingel
Confidentiality:	

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

OdsDok V2.00

Document Organisation

i Notation

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

ii Official Documents and Standards

See bibliography.

iii Revision History

Version	Type of change	Author / team
V1.00	First edition	Dr. Susanne Pingel

Table of Contents

1 ST Introduction

1.1 ST Identification

This Security Target Lite (ST-Lite) refers to the Smartcard Product “Tachograph Card Version 1.0 128/64 R1.0” (TOE) provided by ORGA Kartensysteme GmbH for a Common Criteria evaluation.

Title:	ST-Lite - Tachograph Card Version 1.0 128/64 R1.0
Document Category:	Security Target - Lite for a CC Evaluation according to AIS 35
Document ID:	3TachoEval.CSL.0001
Version:	V1.00
Publisher:	ORGA Kartensysteme GmbH
Confidentiality:	public
TOE:	“Tachograph Card Version 1.0 128/64 R1.0” (Smartcard Product containing IC with Embedded Software dedicated for the Tachograph Application)
Certification ID:	BSI-DSZ-CC-0205
IT Evaluation Scheme:	German CC Evaluation Scheme
Evaluation Body:	SRC Security Research & Consulting GmbH
Certification Body:	Bundesamt für Sicherheit in der Informationstechnik (BSI)

This ST-Lite has been build in conformance with Common Criteria Version 2.1 (ISO 15408) and AIS 35. The ST-Lite has been derived from the full Security Target 3TachoEval.CST.0001, Version V1.01.

1.2 ST Overview

Target of Evaluation (TOE) and subject of this ST-Lite is the Smartcard Product “Tachograph Card Version 1.0 128/64 R1.0” developed by ORGA Kartensysteme GmbH.

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

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

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

The TOE will be employed within the Tachograph System as a security medium which carries a specific Tachograph Application intended for its use with the recording equipment. A Tachograph Card allows for identification of the identity (or identity group) of the cardholder by the recording equipment and allows for data transfer and storage. A Tachograph Card may be of the type Driver Card, Control Card, Workshop Card or Company Card.

The TOE comprises the following components:

- Integrated Circuit (IC) "Philips P16WX064V0C Secure 16-bit Smart Card Controller with Caernarvon Cryptographic Library on Philips Smart *XA*2 as IC Dedicated Support Software" provided by Philips Semiconductors GmbH
- Smartcard Embedded Software based on a Java Card Platform Version 2.1.1 with a specific Java Card Applet for the Tachograph Application provided by ORGA Kartensysteme GmbH

The Java Card Applet for the Tachograph Application consists of a fix part containing the executable code and another configurable part for the Tachograph Card´s filesystem which depends on the respective card type. In this sense, the TOE will be produced and delivered in four different configurations.

The TOE is developed and constructed in full accordance with the Tachograph Card Specification /TachAn1B/, Annex 1B main body, Appendix 2, Appendix 10 (Tachograph Card Generic Security Target) and Appendix 11. In particular, this implies the conformance of the Tachograph Card with the following standards:

- ISO/IEC 7810 Identification cards – Physical characteristics
- ISO/IEC 7816 Identification cards - Integrated circuits with contacts:
  - Part 1: Physical characteristics
  - Part 2: Dimensions and location of the contacts
  - Part 3: Electronic signals and transmission protocols
  - Part 4: Inter-industry commands for interchange
  - Part 8: Security related inter-industry commands
- ISO/IEC 10373 Identification cards – Test methods

As mentioned, the TOE with all its components complies with the Tachograph Card Specification and its functional and security requirements as specified in /TachAn1B/, Annex 1B main body, Appendix 2, Appendix 10 (Tachograph Card Generic Security Target) and Appendix 11. Particularly, the Security Target for the TOE resp. the present ST-Lite takes into account the “Tachograph Card Generic Security Target” for the Tachograph Card in /TachAn1B/, Appendix 10. In order to achieve the required system security, the Tachograph Card and the corresponding Security Target resp. ST-Lite meet all the security requirements and evaluation conditions defined in the Tachograph card´s “Generic Security Target” under consideration of the interpretations in /JILDigTacho/.

The CC evaluation and certification of the TOE against its Security Target serves for the security certificate in the sense of the Tachograph Card Specification /TachAn1B/, Annex 1B main body, chap. VIII. 2. The CC evaluation and certification of the TOE implies the proof for the compliance of the TOE with the requirements of /TachAn1B/, Appendix 10 (Tachograph Card Generic Security Target).

The main objectives of this ST-Lite are

- to describe the TOE as a smartcard product for the Tachograph System
- to define the limits of the TOE
- to describe the assumptions, threats and security objectives for the TOE
- to describe the security requirements for the TOE
- to define the TOE security functions

1.3 CC Conformance

The CC evaluation of the TOE is based upon

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, Version 2.1, August 1999 (/CCPart1/)
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements, Version 2.1, August 1999 (/CCPart2/)
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements, Version 2.1, August 1999 (/CCPart3/)

For the evaluation the following methodology will be used:

- Common Methodology for Information Technology Security Evaluation, Part 2: Evaluation Methodology, Version 1.0, August 1999 (/CEMPart2/)

The Security Target for the TOE resp. the present ST-Lite is written in accordance with the above mentioned Common Criteria Version 2.1 and claims the following CC conformance:

- Part 2 extended
  (Note: The supplement „extended“ is only relevant for the SFRs of the underlying IC with its IC Dedicated Support Software.)
- Part 3 conformant

Furthermore, the Security Target for the TOE resp. the corresponding ST-Lite will be written in view of the requirements of the „Generic Security Target“ for the Tachograph Cards within the Tachograph Card Specification /TachAn1B/, Appendix 10 (Tachograph Card Generic Security Target) and the JIL interpretations and requirements in /JILDigTacho/. In particular, the Security Target complies with the Protection Profile PP9911 „Smartcard Integrated Circuit with Embedded Software“ (/PP9911/). The IC evaluation in compliance with the Protection Profile PP9806 (/PP9806/) as required in /TachAn1B/, Appendix 10 is replaced by the comparable IC evaluation according to the Protection Profile BSI-PP-0002 (/BSI-PP-0002/). Refer for this to the report of the BSI concerning the comparability of the Protection Profiles PP9806 and BSI-PP-0002 (/CompPP9806-BSIPP0002/).

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

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

In order to avoid redundancy and to minimize the evaluation efforts, the evaluation of the TOE will be conducted as a composite evaluation and will make use of the evaluation results of the CC evaluation of the underlying semiconductor "Philips P16WX064V0C Secure 16-bit Smart Card Controller with Caernarvon Cryptographic Library on Philips Smart *XA*2 as IC Dedicated Support Software" provided by Philips Semiconductors GmbH. The IC (incl. its IC Dedicated Software) is evaluated according to Common Criteria EAL 5 augmented with a minimum strength level for its security functions of SOF-high. The evaluation of the IC is based on the Protection Profile BSI-PP-0002 (/BSI-PP-0002/).

2 TOE Description

2.1 TOE Definition

2.1.1 Overview

The Target of Evaluation (TOE) is the Smartcard Product “Tachograph Card Version 1.0 128/64 R1.0” (Tachograph Card for short in the following) implemented in accordance with the Tachograph Card Specification /TachAn1B/, Annex 1B main body, Appendix 2, Appendix 10 (Tachograph Card Generic Security Target) and Appendix 11.

In view of the operating system the Tachograph Card will be realised on base of a Java Card Platform Version 2.1.1 with an appropriate Java Card Applet.

The Tachograph Card is based on the microcontroller "Philips P16WX064V0C Secure 16-bit Smart Card Controller with Caernarvon Cryptographic Library on Philips Smart *XA*2 as IC Dedicated Support Software" provided by Philips Semiconductors GmbH. The IC (incl. its Dedicated Software) is evaluated according to Common Criteria EAL5 augmented with a minimum strength level for its security functions of SOF-high.

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

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

While the Basic Software consists of the Smartcard Operating System of the TOE (based on a Java Card Platform Version 2.1.1), the Application Software implements the specific Tachograph Application in form of a Java Card Applet whereat the Tachograph Applet depends on the respective card type. This will be considered as configuration of the TOE.

Furthermore, the Tachograph Card itself offers the possibility to check its authenticity. For this purpose, the Tachograph Card contains the private part of a dedicated authentication key pair which depends on the configuration of the Tachograph Applet (for more details see chap. 2.1.4.1.1).

The different components of the TOE will be described in the following sections.

2.1.2 TOE Product Scope

TOE component	Designation	Type	Transfer Form
TOE-IC	Philips P16WX064V0C Secure 16-bit Smart Card Controller (incl. its IC Dedicated Software)	HW / SW	---
TOE-ES/BS	Tachograph Smartcard Embedded Software / Part Basic Software	SW	Source Code (implemented in ROM and EEPROM of the microcontroller)
TOE-ES/AS	Tachograph Smartcard Embedded Software / Part Application Software (depending on the resp. card type)	SW	Source Code (implemented in ROM and EEPROM of the microcontroller)
User Guide Personaliser	User guidance for the Personaliser of the Tachograph Card	DOC	Document in paper / electronic form
User Guide Issuer	User guidance for the Issuer of the Tachograph Card	DOC	Document in paper / electronic form
User Guide VU Developer	User guidance for the Developer of Vehicle Units	DOC	Document in paper / electronic form
Identification Data of the Tachograph Card	Data Sheet with the actual identification data of the Tachograph Card delivered to the customer	DOC	Document in paper / electronic form
Aut-Key of the Tachograph Card	Public part of the authentication key pair relevant for the authenticity of the Tachograph Card	KEY	Document in paper / electronic form
Pers-Key of the Tachograph Card	Public part of the personalisation key pair of the Tachograph Card necessary for the personalisation process at the personaliser	KEY	Document in paper / electronic form
Pers-Key Pair of the Personalisation Unit	Personalisation key pair for the personalisation unit necessary for the personalisation of the Tachograph Card delivered to the personaliser	KEY PAIR	Document in paper / electronic form

2.1.3 Integrated Circuit (IC)

Basis for the TOE´s Smartcard Embedded Software is the microcontroller "Philips P16WX064V0C Secure 16-bit Smart Card Controller" (P16WX064V0C for short in the following) with its IC Dedicated Software. The microcontroller and its Dedicated Software are developed and produced by Philips Semiconductors GmbH (within phase 2 and 3 of the smartcard product life-cycle, see chap. 2.2).

2.1.3.1 IC Hardware

The CPU of the P16WX064V0C has a 16-bit architecture with an instruction set that is extended from the 80C51 family instruction set. The first and in some cases the second byte of an instruction are used for operation encoding.

The P16WX064V0C distinguishes between three modes of operations with different privileges: System Mode, Meta Mode and User Mode. The System Mode provides unlimited ac-

cess to the hardware components. For the Meta Mode all hardware components are accessible except the Code Memory Management Unit. In the User Mode the access is restricted to the CPU and specific Special Function Register.

The on-chip hardware components are controlled by the Smartcard Embedded Software via Special Function Registers. These registers are correlated to the activities of the CPU, the memory management unit, interrupt control, I/O configuration, EEPROM, timers, UART, USB and the two coprocessors. (Note: The USB interface will not be used by the Smartcard Embedded Software of the TOE.)

The communication with the P16WX064V0C can be performed through an UART, the direct usage of a I/O port or the USB interface (Note: USB interface not used by the TOE, see above).

The P16WX064V0C provides four different types of interrupts: (i) exceptions, (ii) software traps, (iii) hardware events and (iv) software interrupts. These interrupts force the jump to specific fixed vector addresses in the ROM. Every different interrupts can therefore be controlled and guided by a specific part of Smartcard Embedded Software. In conjunction with the jump to a specific fixed vector address the hardware always enables the System Mode. Therefore the handling of the interrupts is supported by the separation between the modes of operation.

The device includes ROM (128 kByte User-ROM + 12 kByte Test-ROM), RAM (5152 Byte) and EEPROM (64 kByte) memory.

The access control by the memory management unit for code is related to the ROM and the EEPROM. The access control by the memory management unit for data is related to the RAM. Nevertheless the EEPROM can be accessed as data memory as well as program memory. Smartcard Embedded Software running in the System Mode has unlimited access to the memories. In the Meta Mode the Smartcard Embedded Software is not allowed to make modifications of the configuration of the code memory management unit. Smartcard Embedded Software running in the User Mode can not make configuration changes to the memory management.

The Triple-DES co-processor supports single DES and Triple-DES operations. Only Triple-DES will be used by the Smartcard Embedded Software.

The FameX co-processor supplies basic arithmetic functions to perform asymmetric crypto algorithms (here: RSA) implemented by the Smartcard Embedded Software.

The random generator of the IC provides true random numbers without pseudo random calculation.

The P16WX064V0C operates with a single 3V or 5V nominal power supply (except the power supply for the USB operation that must be nominal 5V). The nominal maximum external clock frequency is 6 MHz. The microcontroller can be operated with the internal clock especially to decrease the calculation time for security algorithms. The microcontroller provides power saving modes with reduced activity: the IDLE Mode and the SLEEP Mode, which includes the CLOCK STOP Mode.

The P16WX064V0C protects the secret data stored in and operated by the IC against physical tampering.

Within the composition of the IC with the Smartcard Embedded Software (Basic Software and Application Software) the security functionality is only partly provided by the IC and causes dependencies between the IC security functions and the functions provided by the operating system or the smartcard application on top.

2.1.3.2 IC Dedicated Software

The IC Dedicated Software (IC firmware) comprises proprietary software embedded in the IC and can be divided into two parts:

- IC Dedicated **Test** Software
- IC Dedicated **Support** Software

The usage of the different parts of the IC Dedicated Software is restricted to certain phases in the life-cycle of the product.

The IC Dedicated Test Software part of the IC firmware is used for test purposes of the IC before its delivery but does not provide any functionality thereafter. The IC Dedicated Test Software is embedded in the Test-ROM of the P16WX064V0C and is used to test the functionality of the chip. The IC Dedicated Test Software includes the test operating system, test routines for the various blocks of the circuitry, control flags for the status of the EEPROM's security area and shutdown functions to ensure that security relevant test operations cannot be executed illegally after phase 3 of the smartcard product life-cycle (see chap. 2.2). The IC Dedicated Test Software is disabled before the operational use of the smartcard.

The IC Dedicated Support Software part of the IC firmware provides specific additional cryptographic services for the Smartcard Embedded Software of the TOE. In particular, the IC Dedicated Support Software covers the „Caernarvon Secure Cryptographic Library" (Crypto Library for short) for the TOE´s cryptographic features. The implementation of the Crypto Library provided by Philips is written specifically for the underlying IC of the TOE and its coding, defences against attacks, et cetera all apply specifically to the IC. The Crypto Library is evaluated together with the IC according to Common Criteria.

The Crypto Library provides a set of core routines for the cryptographic functions of the Smartcard Embedded Software. In particular, routines of the Crypto Library for modular arithmetic, RSA and DES operations, random number generation incl. quality test, hash value generation are used by the Smartcard Embedded Software for its cryptographic features. As applicable, the routines of the Crypto Library used by the Smartcard Embedded Software are secured against SPA (Simple Power Analysis), DPA (Differential Power Analysis), DFA (Differential Fault Analysis) and Timing Attacks.

2.1.3.3 IC Interfaces

In the Application Mode the electrical interface of the P16WX064V0C are the pads to connect the lines power supply, reset input, clock input, ground, UART, USB and I/O2.

The software interface of the P16WX064V0C depends on the IC mode:

- In the Test Mode (used before delivery of the P16WX064V0C after production) the logical interface that is visible on the electrical interface is defined by the IC Dedicated Test Software. This IC Dedicated Test Software comprises the test operating system and the package of test function calls stored in the Test-ROM.
- In the Application Mode (used after delivery of the P16WX064V0C) the software interface is the set of instructions, the bits in the special function registers that are related to the Application Mode and the address map of the CPU including memories. The access to the special function registers as well as to the memories depends on the mode (system, meta or user) configured by the operating system.

The logical interface of the IC that is visible on the electrical interface after delivery of the P16WX064V0C is based on the Smartcard Embedded Software. The identification and authentication of the user for the different modes (system, meta and user) is controlled by the Smartcard Embedded Software.

An external voltage and timing supply as well as a data interface are necessary for the operation of the IC. Beyond the physical behaviour the data interface is defined by the Smartcard Embedded Software.

2.1.4 Smartcard Embedded Software

The Smartcard Embedded Software of the TOE comprises the Smartcard Operating System and the specific Tachograph Application and is therefore divided into two parts with specific contents:

- Basic Software (Smartcard Operating System)
- Application Software (Tachograph Application)

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

2.1.4.1 Basic Software

The Basic Software of the Smartcard Embedded Software comprises the Operating System of the TOE and makes use of the Java Card Technology which combines a subset of the Java programming language with a runtime environment optimized for smartcards and provides facilities for secure interoperability of applications.

2.1.4.1.1 Java Card Platform

The Java Card Platform Version 2.1.1 as implemented for the TOE consists of the three following parts:

- Java Card Virtual Machine (JCVM)

- Java Card Runtime Environment (JCRE)
- Java Card Application Programming Interface (JCAPI).

In the following, the term “Java Card System” (JCS) designates the set made of the JCRE, the JCVM and the JCAPI.

The specification and coding of the TOE´s Operating System with its Java Card Technology is carried out according to the Java Card Platform Version 2.1.1 as specified in /JCVM21/, /JCRE21/, /JCAPI21/.

JCVM

The JCVM is essentially an abstract machine embedded in the smartcard that functions as the Java Card Bytecode Interpreter. The JCVM is the component that enforces separation between applications and enables secure data sharing.

A Java Card applet is usually intended to store highly sensitive information, so the sharing of that information must be carefully limited. In the Java Card Platform applet isolation is achieved through the so-called Firewall mechanism.

The Firewall mechanism in the Java Card Technology is responsible for ensuring applet isolation and object sharing. The firewall prevents an applet in one context from unauthorized access to objects owned by the JCRE or by an applet in another context. The mechanism confines an applet to its own designated memory area, thus each applet is prevented from accessing fields and operations of objects owned by other applets, unless an interface is explicitly provided (by the applet who owns it) for allowing access to that information. However applet isolation cannot entirely be granted by the Firewall mechanism if certain integrity conditions are not satisfied by the applications on the card, those conditions can be statically verified to hold by a Bytecode Verifier.

JCRE

The JCRE stands in direct contact with the JCVM, the JCAPI and its associated native methods. This concerns all those dynamic features that are specific to the execution of a Java program in a smartcard, like applet lifetime, applet isolation and object sharing, transient objects, transaction mechanism, etc.

The JCRE is responsible for

- card resource management
- communication
- applet execution
- on-card system and applet security

The JCRE Entry Points are the gateways through which applets request privileged JCRE system services.

JCAPI

The JCAPI

- provides framework classes and interfaces for the core functionality of a Java Card application
- defines the calling conventions by which an applet may access the JCRE and native services such as I/O management functions, PIN and cryptographic specific management and the Exception mechanism.

The Java Card API is compatible with formal international standards, such as ISO 7816, and industry specific standards, such as EMV (Europay/Master Card/Visa).

2.1.4.1.2 Card Manager

- The Java Card Platform is supplemented with the so-called Card Manager, a special component of the Operating System. The Card Manager is an application with specific rights, which is responsible for the administration of the Java Card.

The Tachograph Card offers the capability to check its authenticity. For this purpose, the Card Manager contains the private part of a dedicated authentication key pair (RSA 1024 Bit) over which by an internal authentication procedure the authenticity of the Tachograph Card can be proved.

2.1.4.1.3 Native Platform

The Native Platform of the TOE´s Operating System serves as an abstraction layer between the Java Card Platform and the IC.

2.1.4.1.4 Initialisation Module

The Initialisation Module contains specific initialisation routines for loading initialisation files and specific test routines for checking the correct functioning of the EEPROM area.

2.1.4.2 Application Software

The Application Software part of the Smartcard Embedded Software of the TOE comprises the Tachograph Application itself.

The Tachograph Application is realised as a Java Card applet written in Java Card language (named Tachograph Applet in the following) and is uniquely identified by its own AID. The execution of the Tachograph Applet residing on the card is performed by the TOE´s Java Card Platform and its on-card bytecode interpreter.

The Tachograph Application is implemented in conformance with the requirements of the Tachograph Card Specification /TachAn1B/, Annex 1B main body, Appendix 2, Appendix 10 (Tachograph Card Generic Security Target) and Appendix 11.

For more details about the behaviour of the Tachograph Application refer to the Tachograph Card Specification /TachAn1B/ and to chap. 2.4 of this document.

2.2 TOE Life-Cycle

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

Phase		Description
Phase 1	**Smartcard Embedded Software Development**	The **Smartcard Embedded Software Developer (ORGA Kartensysteme GmbH)** is in charge of • the Smartcard Embedded Software (Basic Software, Application Software) development and • the specification of IC initialisation and pre-personalisation requirements (though the actual data for IC initialisation and pre-personalisation come from Phase 4, 5 resp. 6). The purpose of the Embedded Software designed during phase 1 is to control and protect the TOE during phases 4 to 7 (product usage).The global security requirements of the TOE are such that it is mandatory during the development phase to anticipate the security threats of the other phases.
Phase 2	**IC Development**	The **IC Designer (Philips Semiconductors GmbH)** • designs the IC, • develops IC Dedicated Software, • provides information, software or tools to the Smartcard Embedded Software Developer, and • receives the Smartcard Embedded Software (only Basic Software) from the developer through trusted delivery and verification procedures. From the IC design, IC Dedicated Software and Smartcard Embedded Software, the **IC Designer (Philips Semiconductors GmbH)** • constructs the smartcard IC database, necessary for the IC photomask fabrication.

Phase 3	**IC Manufacturing and Testing**	The **IC Manufacturer (Philips Semiconductors GmbH)** is responsible for
		• producing the IC through three main steps:
		  - IC manufacturing,
		  - IC testing, and
		  - IC pre-personalisation.
		The **IC Mask Manufacturer (Philips Semiconductors GmbH)**
		• generates the masks for the IC manufacturing based upon an output from the smartcard IC database.

Phase 4	**IC Packaging and Testing**	The **IC Packaging Manufacturer (ORGA Kartensysteme GmbH)** is responsible for • the IC packaging (production of modules) and

		• testing.
Phase 5	**Smartcard Product Finishing Process**	The **Smartcard Product Manufacturer (ORGA Kartensysteme GmbH)** is responsible for • the initialisation of the TOE (in form of initialisation of the modules of phase 4) and • its testing. The smartcard product finishing process comprises the embedding of the initialised modules for the TOE and the card production what is done alternatively by **ORGA Kartensysteme GmbH or by the customer.**
Phase 6	**Smartcard Personalisation**	The **Personaliser** is responsible for • the smartcard personalisation and • final tests.
Phase 7	**Smartcard End-usage**	The **Smartcard Issuer** is responsible for • the smartcard product delivery to the smartcard end-user, and the end of life process.

Appropriate procedures for a secure delivery process of the TOE or parts of the TOE under construction from one development resp. production site to another site within the smartcard product life-cycle are established.

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

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

2.3 TOE Environment

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

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

2.3.1 Development Environment

Phase 1 - Smartcard Embedded Software Development

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

Phase 2 – IC Development

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

2.3.2 Production Environment

Phase 3 - IC Manufacturing and Testing

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

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

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

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

The delivery of the ICs from Philips Semiconductors GmbH to ORGA Kartensysteme GmbH is made in form of wafers whereby nonfunctional ICs are marked on the wafer.

Phase 4 – IC Packaging and Testing

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

Phase 5 - Smartcard Product Finishing Process

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

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

Otherwise, the smart card finishing process is part of the production process at ORGA Kartensysteme GmbH, and the TOE is delivered in form of complete (initialised) cards.

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

2.3.3 Personalisation Environment

Note: The phases from the TOE delivery at the end of phase 5 to phase 7 in the smartcard product life-cycle are not part of theTOE development and production process in the sense of the TOE´s Security Target. Information about the phases 6 and 7 are just included to describe how the TOE is used after its development and production. The development and production of the TOE is done in such a way that the security features of the TOE are independent of the user data loaded during its personalisation and cannot be disabled by the personalisation data in the phases afterwards.

Phase 6 - Smartcard Personalisation

The security of the personalisation process of the TOE is supported by the TOE and its Application Software itself. The TOE allows a personalisation only after a successful preceding mutual authentication between the TOE and the external world (according to the procedure described in the Tachograph Card Specification /TachAn1B/, Appendix 11, chap. 4) and only with secured data transfer using the session key and the send sequence counter agreed during the authentication process. The keys necessary for the authentication procedure are part of the Application Software resp. the Tachograph Applet and are brought onto the card in the framework of the TOE´s production.

The establishment of a secure environment for the personalisation process with adequate personnel, organisational and technical security measures is in the responsibility of the personalisation centre itself. Furthermore, the secure handling of the personalisation data and keys is task of the external world resp. the personalisation centre.

2.3.4 End-User Environment

Phase 7 – Smartcard End-usage

The TOE after its personalisation is destinated for use in the Tachograph System as a security medium and data carrier for different user types which is secured against forgery and tampering. For further details concerning the use of the Tachograph Card refer to chap. 2.4.

The TOE is constructed in such a manner that it implements all security requirements of the Tachograph Card Specification /TachAn1B/. There is no possibility, even in an insecure end-user environment, to disable or to circumvent the security features of the TOE.

2.4 TOE Intended Usage

In this section, the intended usage of the TOE within the end-usage phase of the product life-cycle (phase 7), i.e. the intended usage of the TOE in personalised form will be regarded more detailed.

According to the Tachograph Card Specification /TachAn1B/ and the interpretations in /JILDigTacho/ a Tachograph Card is defined as a smartcard product compliant to the Protection Profiles /PP9806/ resp. /BSI-PP-0002/ and /PP9911/ and carrying a specific application intended for its use with the recording equipment. Tachograph Cards allow for identification of the identity (or identity group) of the cardholder by the recording equipment and allow for data transfer and storage.

A Tachograph Card may be of the following types:

- Driver Card:
  a Tachograph Card issued by the authorities of a Member State to a particular driver; identifies the driver and allows for storage of driver activity data
- Control Card:
  a Tachograph Card issued by the authorities of a Member State to a national competent control authority; identifies the control body and possibly the control officer and allows for getting access to the data stored in the data memory or in the driver cards for reading, printing and/or downloading
- Workshop Card:
  a Tachograph Card issued by the authorities of a Member State to a recording equipment manufacturer, a fitter, a vehicle manufacturer or workshop approved by that Member State; identifies the cardholder and allows for testing, calibration and/or downloading of the recording equipment
- Company Card:
  a Tachograph Card issued by the authorities of a Member State to the owner or holder of vehicles fitted with recording equipment; identifies the company and allows for displaying, downloading and printing of the data stored in the recording equipment which has been locked by this company

The basic functions of the Tachograph Card are the following:

- to store card identification and card holder identification data; these data are used by the vehicle unit to identify the cardholder, provide accordingly functions and data access rights, and ensure cardholder accountability for his activities
- to store cardholder activities data, events and faults data and control activities data related to the cardholder

A Tachograph Card is therefore intended to be used by a card interface device of a vehicle unit. It may also be used by any card reader (e.g. of a personal computer) which shall have full read access right on any user data.

During the end-usage phase of a Tachograph Card (phase 7 of the smartcard product life-cycle), vehicle units only may write user data to the card.

Regarding the security of the Tachograph System, the system security aims at

- protecting integrity and authenticity of data exchanged between the cards and the recording equipment
- protecting the integrity and authenticity of data downloaded from the cards
- allowing certain write operations onto the cards to recording equipment only
- ruling out any possibility of falsification of data stored in the cards
- preventing tampering and detecting any attempt of that kind

Especially the following security mechanisms are relevant for the Tachograph Card:

- mutual authentication between a vehicle unit and a Tachograph Card, including session key agreement

- confidentiality, integrity and authentication of data transferred between a vehicle unit and a Tachograph Card
- integrity and authentication of data downloaded from a Tachograph Card to external storage media

The Tachograph Card offers a classical RSA public-key cryptographic system to provide the following security mechanisms:

- authentication between a vehicle unit and a Tachograph Card
- transport of Triple-DES session keys between a vehicle unit and a Tachograph Card
- digital signature of data downloaded from a Tachograph Card to external media

Furthermore, the Tachograph Card offers a Triple DES symmetric cryptographic system to provide a mechanism for data integrity during user data exchange between a vehicle unit and a Tachograph Card, and to provide, where applicable, confidentiality of data exchange between a vehicle unit and a Tachograph Card.

3 TOE Security Environment

3.1 Assets

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

The confidentiality of the code of the TOE is included in the TOE´s Security Target for several reasons. First, the confidentiality is needed for the protection of intellectual/industrial property on security or effectiveness mechanisms. Second, though protection shall not rely exclusively on code confidentiality, disclosure of the code may weaken the security of the involved application. For instance, knowledge about the implementation of the Operating System or the Tachograph Application itself may benefit an attacker. This also applies to internal data of the TOE, which may similarly provide leads for further attacks.

For a description of the TOE´s assets refer to /TachAn1B/, Appendix 10 (Tachograph Card Generic Security Target), /PP9911/, chap. 3.1, /BSI-PP-0002/, chap. 3.1, /ST-ICPhilips/, chap. 3.1. The assets of the TOE sorted in primary and seconday assets are listed in the tables below:

Primary Assets	
Part of the TOE	**Definition**
IC	---
Smartcard Embedded Software / Basic Software	---

Smartcard Embedded Software / Application Software	- application specific user data (refer to /TachAn1B/, Appendix 10, Tachograph Card Generic Security Target, chap. 2.2) as
	  - identification data (card identification data, cardholder identification data)
	  - activity data (cardholder activities data, events and faults data, control activity data)

Secondary Assets	
Part of the TOE	**Definition**
IC	- logical design data - physical design data - IC Dedicated Software - initialisation data - pre-personalisation data - specific development aids - test and characterisation related data - material for software development support - photomasks - the special functions for the communication with an external interface device - the cryptographic co-processor for Triple-DES - the FameX co-processor for basic arithmetic functions to perform asymmetric cryptographic algorithms - the random number generator - TSF data
Smartcard Embedded Software / Basic Software	- specifications - code - related documentation - system specific data - initialisation data - specific development aids - test and characterisation related data - material for software development support - TSF data
Smartcard Embedded Software / Application Software	- specifications - code - related documentation - system specific data - initialisation data - specific development aids - test and characterisation related data - material for software development support - user data related documentation - TSF data, especially the application specific security data (refer to /TachAn1B/, Appendix 10, Tachograph Card Generic Security Target, chap. 2.2)

3.2 Assumptions

3.2.1 General Assumptions for the TOE

The general assumptions made on the environment of the TOE are defined according to /PP9911/, chap. 3.2 and are suitably supplemented for the TOE. The complete set of assumptions is listed in the table below.

Note:

In the following table, items of /PP9911/ which are common with /PP9806/ are indicated by a "*".

Assumptions for the Environment of the TOE	
Name	**Definition**
Assumptions on Phase 1 to 5	
A.DEV_ORG* (PP9911+supplement)	**Protection of the TOE under Development and Production** Procedures dealing with physical, personnel, organizational, technical measures for the confidentiality and integrity of the Smartcard Embedded Software (e.g. source code and any associated documents) and IC designer proprietary information (tools, software, documentation ...) shall exist and be applied in software development. All authorities involved in the development and production of the TOE shall carry out their development and production activities in a suitable and secure environment. Each party has to ensure that the development and production of the TOE (incl. IC with its Dedicated Software, Smartcard Embedded Software) is secure so that no information is unintentionally made available for the later operational phase of the TOE. In particular, the confidentiality and integrity of design information and test data shall be guaranteed, access to development and test tools, samples and other sensitive material shall be restricted to authorised persons only etc.
Assumptions on the TOE Delivery Process (Phases 4 to 7)	
A.DLV_PROTECT* (PP9911)	**Protection of the TOE under Delivery and Storage** Procedures shall ensure protection of TOE material / information under delivery and storage.
A.DLV_AUDIT* (PP9911)	**Audit of Delivery and Storage** Procedures shall ensure that corrective actions are taken in case of improper operation in the delivery process and storage.

A.DLV_RESP* (PP9911)	**Responsibility within Delivery** Procedures shall ensure that people dealing with the procedure for delivery have got the required skill.
Assumptions on Phases 4 to 6	
A.USE_TEST* (PP9911)	**Testing of the TOE** It is assumed that appropriate functionality testing of the TOE is used in phases 4, 5 and 6.
A.USE_PROD* (PP9911)	**Protection of the TOE under Testing and Manufacturing** It is assumed that security procedures are used during all manufacturing and test operations through phases 4, 5, 6 to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorized use).
Assumptions on Phase 6	
A.PERS	The originator of the personalisation data and the personalisation center responsible for the personalisation of the TOE handles the personalisation data in an adequate secure manner. This concerns especially the security data to be personalised as secret cryptographic keys and PINs. The storage of the personalisation data at the originator and at the personalisation center as well as the transfer of these data between the different sides is conducted with respect to data integrity and confidentiality. Furthermore, the personalisation center treats the data for securing the personalisation process, i.e. the personalisation keys suitably secure. It is in the responsibility of the originator of the personalisation data to garantuee for a sufficient quality of the personalisation data, especially of the cryptographic material to be personalised. The preparation and securing of the personalisation data appropriate to the Tachograph Card´s structure and according to the TOE´s personalisation requirements is as well in the responsibility of the external world and is done with care.
Assumptions on Phase 7	
A.USE_DIAG*	**Secure Communication** It is assumed that secure communication protocols and procedures are used between smartcard and terminal.

3.2.2 Tachograph Card Specific Assumptions for the TOE

There do not exist any Tachograph Card specific assumptions for the environment of the TOE.

3.3 Threats

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

Generally, threats can be split into the following types:

- threats against which a specific protection by the TOE is required
- threats against which a specific protection by the environment is required
- threats against which a specific protection by a combination of the TOE and the environment is required

Before listing the general threats for the TOE, several preliminary remarks about these threats:

Threats on phase 1

During phase 1, three types of threats have to be considered:

- threats on the TOE-ES and its development environment, such as unauthorized disclosure, modification or theft of the TOE-ES and/or initialisation data
- threats on the assets transmitted from the IC designer to the TOE-ES developer during the TOE-ES development
- threats on the TOE-ES and initialisation data transmitted during the delivery process from the TOE-ES software developer to the IC designer

Furthermore, one can consider the threats under the aspect of disclosure, theft, use or modification:

- Unauthorized disclosure of assets:

  This type of threats covers unauthorized disclosure of assets by attackers who may possess a wide range of technical skills, resources and motivation. Such attackers may also have technical awareness of the product.

- Theft or unauthorized use of assets:

  Potential attackers may gain access to the TOE and perform operations for which they are not authorized. For example, such an attacker may personalize, modify or influence the product in order to gain access to the smartcard application system.

- Unauthorized modification of assets:

  The TOE may be subjected to different types of logical or physical attacks which may compromise security. Due to the intended usage of the TOE (the TOE environment may be hostile), the TOE security may be bypassed or compromised reducing the integrity of the TOE security mechanisms and disabling their ability to manage the TOE security. This type of threats includes the implementation of malicious Trojan horses.

Threats on delivery from phase 1 to phases 4 to 6

Threats on data transmitted during the delivery process from the TOE-ES developer to the IC packaging manufacturer, the Finishing process manufacturer or the Personalizer.

Threats on phases 4 to 7

During these phases, the assumed threats could be divided in three types:

- Unauthorized disclosure of assets:

  This type of threats covers unauthorized disclosure of assets by attackers who may possess a wide range of technical skills, resources and motivation. Such attackers may also have technical awareness of the product.

- Theft or unauthorized use of assets:

  Potential attackers may gain access to the TOE and perform operation for which they are not allowed. For example, such attackers may personalize the product in an unauthorized manner, or try to gain fraudulently access to the smartcard system.

- Unauthorized modification of assets:

  The TOE may be subjected to different types of logical or physical attacks which may compromise security. Due to the intended usage of the TOE (the TOE environment may be hostile), the TOE security parts may be bypassed or compromised reducing the integrity of the TOE security mechanisms and disabling their ability to manage the TOE security. This type of threat includes the implementation of malicious Trojan horses, Trapdoors, downloading of viruses or unauthorized programs.

3.3.1 Threats of the IC (TOE-IC)

The table below lists the general threats to the assets of the TOE-IC against which specific protection within the TOE or its environment is required. The listed threats concern only phase 7 of the product life-cycle and follow the details given in /BSI-PP-0002/, chap. 3.3, /ST-ICPhilips/, chap. 3.3 and the Security Target related to the evaluation of the IC incl. its Crypto Library.

Note:

For clarity, within the description of the threats in the following table as given in /BSI-PP-0002/ and /ST-ICPhilips/ the word „TOE“ is replaced by „TOE-IC“ and the term „Smartcard Embedded Software“ is supplemented by „TOE-ES“.

Threats / TOE-IC	
Name	**Definition**
T.Leak-Inherent	**Inherent Information Leakage** An attacker may exploit information which is leaked from theTOE-IC during usage of the smartcard in order to disclose confidential data (user data or TSF data). No direct contact with the smartcard internals is required here. Leakage may occur through emanations, variations in power consumption, I/O characteristics, clock frequency or by changes in processing time requirements. One example is the Differential Power Analysis (DPA). This leakage may be interpreted as a covert channel transmission but is more closely related to measurement of operating parameters, which may be derived either from direct (contact) measurements (/BSI-PP-0002/, Numbers 6 and 7 in Figure 8) or measurement of emanations (/BSI-PP-0002/, Number 5 in Figure 8) and can then be related to the specific operation being performed.
T.Phys-Probing	**Physical Probing** An attacker may perform physical probing of the TOE-IC in order (i) to disclose user data, (ii) to disclose/reconstruct the IC Dedicated Software (TOE-IC) or the Smartcard Embedded Software (TOE-ES) or (iii) to disclose other critical operational information, especially TSF data. Physical probing requires direct interaction with the Smartcard Integrated Circuit internals (/BSI-PP-0002/, Numbers 5 and 6 in Figure 8). Techniques commonly employed in IC failure analysis and IC reverse engineering efforts may be used. Before that hardware security mechanisms and layout characteristics need to be identified (/BSI-PP-0002/, Number 3 in Figure 8). Determination of software design including treatment of user data may also be a prerequisite. This pertains to “measurements” using galvanic contacts or any type of charge interaction whereas manipulations are considered under the threat “Physical Manipulation (T.Phys-Manipulation)”. The threats “Inherent Information Leakage (T.Leak-Inherent)” and “Forced Information Leakage (T.Leak-Forced)“ may use physical probing but require complex signal processing in addition.
T.Malfunction	**Malfunction due to Environmental Stress** An attacker may cause a malfunction of theTSF of the TOE-IC or of the Smartcard Embedded Software (TOE-ES) by applying environmental stress in order to (i) deactivate or modify security features or functions of the TOE-IC or (ii) deactivate or modify security functions of the Smartcard Embedded Software (TOE-ES). This may be achieved by operating the smartcard outside the normal operating conditions (/BSI-PP-0002/, Numbers 1, 2 and 9 in Figure 8). To exploit this an attacker needs information about the functional operation.
T.Phys-Manipulation	**Physical Manipulation** An attacker may physically modify the smartcard in order to (i) modify security features or functions of the TOE-IC, (ii) modify security functions of the Smartcard Embedded Software (TOE-ES) or

	(iii) modify user data. The modification may be achieved through techniques commonly employed in IC failure analysis (/BSI-PP-0002/, Numbers 1, 2 and 4 in Figure 8) and IC reverse engineering efforts (/BSI-PP-0002/, Number 3 in Figure 8). The modification may result in the deactivation of a security function. Before that hardware security mechanisms and layout characteristics need to be identified. Determination of software design including treatment of user data may also be a prerequisite. Changes of circuitry or data can be permanent or temporary. In contrast to malfunctions (refer to T.Malfunction) the attacker requires to gather significant knowledge about the TOE-IC's internal construction here /BSI-PP-0002/, Number 3 in Figure 8).
T.Leak-Forced	**Forced Information Leakage** An attacker may exploit information which is leaked from the TOE-IC during usage of the smartcard in order to disclose confidential data (user data or TSF data) even if the information leakage is not inherent but caused by the attacker. This threat pertains to attacks where methods described in "Malfunction due to Environmental Stress" (refer to T.Malfunction) and/or "Physical Manipulation" (refer to T.Phys-Manipulation) are used to cause leakage from signals (/BSI-PP-0002/, Numbers 5, 6, 7 and 8 in Figure 8) which normally do not contain significant information about secrets.
T.Abuse-Func	**Abuse of Functionality** An attacker may use functions of the TOE-IC which may not be used after TOE-IC Delivery in order to (i) disclose or manipulate user data, (ii) manipulate (explore, bypass, deactivate or change) security features or functions of the TOE-IC or of the Smartcard Embedded Software (TOE-ES) or (iii) enable an attack.
T.RND	**Deficiency of Random Numbers** An attacker may predict or obtain information about random numbers generated by the TOE-IC for instance because of a lack of entropy of the random numbers provided. An attacker may gather information about the produced random numbers which might be a problem because they may be used for instance to generate cryptographic keys. Here the attacker is expected to take advantage of statistical properties of the random numbers generated by the TOE-IC without specific knowledge about the TOE-IC's generator. Malfunctions or premature ageing are also considered which may assist in getting information about random numbers. The preceding points include the hardware RNG of the TOE-IC as well as its software RNG.

3.3.2 General Threats of the Smartcard Embedded Software (TOE-ES)

The table below lists the general threats to the assets of the TOE-ES against which specific protection within the TOE or its environment is required. Several groups of threats are distinguished according to the means used in the attack and to the phases of the TOE that are affected. The threats to the TOE-ES are defined as indicated in /PP9911/, chap. 3.3.

Note:

In the following table, items of /PP9911/ which are common with /PP9806/ are indicated by a "*".

Threats / TOE-ES	
Name	**Definition**
Threats on all Phases	
T.CLON* (PP9911)	**Cloning of the TOE** Unauthorized full or partional functional cloning of the TOE. Note: This threat is derived from specific threats combining unauthorized disclosure, modification or theft of assets at different phases.
Threats on Phase 1	
T.DIS_INFO* (PP9911)	**Disclosure of IC Assets** Unauthorized disclosure of the assets delivered by the IC designer to the Smartcard Embedded Software developer, such as sensitive information on IC specification, design and technology, software and tools if applicable.
T.DIS_DEL* (PP9911)	**Disclosure of the Smartcard Embedded Software / Application Data during Delivery** Unauthorized disclosure of the Smartcard Embedded Software and any additional application data (such as IC Pre-personalization requirements) during the delivery from the Smartcard Embedded Software developer to the IC designer.
T.DIS_ES1 (PP9911)	**Disclosure of the Smartcard Embedded Software / Application Data within the Development Environment** Unauthorized disclosure of the Smartcard Embedded Software (technical or detailed specifications, implementation code) and/or Application Data (such as secrets, or control parameters for protection system, specification and implementation for security mechanisms) within the development environment.
T.DIS_TEST_ES (PP9911)	**Disclosure of Smartcard Embedded Software Test Programs / Information** Unauthorized disclosure of the the Smartcard Embedded Software test programs or

	any related information.
T.T_DEL* (PP9911)	**Theft of the Smartcard Embedded Software / Application Data during Delivery** Theft of the Smartcard Embedded Software and any additional application data (such as pre-personalization requirements) during the delivery process from the Smartcard Embedded Software developer to the IC designer.
T.T_TOOLS (PP9911)	**Theft or Unauthorized Use of the Smartcard Embedded Software Development Tools** Theft or unauthorized use of the Smartcard Embedded Software development tools (such as PC, development software, data bases).
T.T_SAMPLE2 (PP9911)	**Theft or Unauthorized Use of TOE Samples** Theft or unauthorized use of TOE samples (e.g. bond-out chips with the Smartcard Embedded Software).
T_MOD_DEL* (PP9911)	**Modification of the Smartcard Embedded Software / Application Data during Delivery** Unauthorized modification of the Smartcard Embedded Software and any additional application data (such as IC prepersonalization requirements) during the delivery process from the Smartcard Embedded Software developer to the IC designer.
T.MOD (PP9911)	**Modification of the Smartcard Embedded Software / Application Data within the Development Environment** Unauthorized modification of the Smartcard Embedded Software and/or Application Data or any related information (technical specifications) within the development environment.
Threats on Delivery from Phase 1 to Phases 4 / 5 / 6	
T.DIS_DEL1 (PP9911)	**Disclosure of Application Data during Delivery** Unauthorized disclosure of Application Data during delivery from the Smartcard Embedded Software developer to the IC Packaging manufacturer, the Finishing process manufacturer or the Personalizer.
T.DIS_DEL2 (PP9911)	**Disclosure of Delivered Application Data** Unauthorized disclosure of Application Data delivered from the Smartcard Embedded Software developer to the IC Packaging manufacturer, the Finishing process manufacturer or the Personalizer.
T.MOD_DEL1 (PP9911)	**Modification of Application Data during Delivery** Unauthorized modification of Application Data during delivery from the Smartcard Embedded Software developer to the IC Packaging manufacturer, the Finishing process manufacturer or the Personalizer.
T.MOD_DEL2	**Modification of Delivered Application Data**

(PP9911)	Unauthorized modification of Application Data delivered from the Smartcard Embedded Software developer to the IC Packaging manufacturer, the Finishing process manufacturer or the Personalizer.
Threats on Phases 4 to 7	
T.DIS_ES2 (PP9911)	**Disclosure of the Smartcard Embedded Software / Application Data** Unauthorized disclosure of the Smartcard Embedded Software and Application Data (such as data protection systems, memory partitioning, cryptographic programs and keys).
T.T_ES (PP9911)	**Theft or Unauthorized Use of TOE** Theft or unauthorized use of the TOE (e.g. bound out chips with the Smartcard Embedded Software).
T.T_CMD (PP9911)	**Use of TOE Command-Set** Unauthorized use of instructions or commands or sequence of commands sent to the TOE.
T.MOD_LOAD (PP9911)	**Program Loading** Unauthorized loading of programs.
T.MOD_EXE (PP9911)	**Program Execution** Unauthorized execution of programs.
T.MOD_SHARE (PP9911)	**Modification of Program Behavior** Unauthorized modification of program behavior by interaction of different programs.
T.MOD_SOFT* (PP9911)	**Modification of Smartcard Embedded Software / Application Data** Unauthorized modification of the Smartcard Embedded Software and Application Data.

3.3.3 Tachograph Card Specific Threats

The following table lists the specific threats relevant for the Tachograph Application within the TOE-ES. The threats are provided by the Tachograph Card Specification /TachAn1B/, Appendix 10, Tachograph Card Generic Security Target, chap. 3.3 and are supplemented for the TOE´s personalisation.

Threats / TOE-ES (Tachograph Card Specific Threats)	
Name	**Definition**
T.Ident_Data	**Modification of Identification Data** A successful modification of identification data held by the TOE (e.g. the type of card, or the card expiry date or the cardholder identification data) would allow a fraudulent use of the TOE and would be a major threat to the global security objective of the system.
T.Activity_Data	**Modification of Activity Data** A successful modification of activity data stored in the TOE would be a threat to the security of the TOE.
T.Data_ex-change	**Modification of Activity Data during Data Transfer** A successful modification of activity data (addition, deletion, modification) during import or export would be a threat to the security of the TOE.
T.Pers_Data	**Authentication for Personalisation** A successful storage of personalisation data without authorisation would be a threat to the security of the TOE.
T.Pers_ex-change	**Modification or Disclosure of Personalisation Data during Data Transfer** A successful modification or disclosure of personalisation data during data import would be a threat to the security of the TOE.

3.4 Organisational Security Policies of the TOE

The TOE reaches is specific security functionality only by a correct and effective implementation of the underlying IC and its security functionality by the Smartcard Embedded Software (TOE-ES). In particular this means, that the TOE-ES must fulfill the assumptions for the TOE-ES as defined in the Security Target for the TOE-IC.

The relevant assumptions for the TOE-ES as given in the Security Target related to the evaluation of the IC incl. its Crypto Library (refer also to /ST-ICPhilips/, chap. 3.2 and /BSI-PP-0002/, chap. 3.2) are suitably redefined in terms of organisational security policies for the TOE as follows:

Organisational Security Policy for the TOE	
Name	**Definition**
P.Process-Card	**Protection during Packaging, Finishing and Personalisation** Security procedures shall be used after TOE-IC Delivery up to delivery to the end-user to maintain confidentiality and integrity of the TOE-IC and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use).
P.Design-Software	**Design of the Smartcard Embedded Software** To ensure that the TOE-IC is used in a secure manner the Smartcard Embedded Software (TOE-ES) shall be designed so that the requirements from the following documents are met: - hardware data sheet for the TOE-IC, - TOE-IC application notes, - findings of the TOE-IC evaluation reports relevant for the Smartcard Embedded Software (TOE-ES). Security relevant User Data (especially cryptographic keys) are treated by the Smartcard Embedded Software (TOE-ES) as required by the security needs of the specific application context. For example the Smartcard Embedded Software (TOE-ES) will not disclose security relevant user data to unauthorised users or processes when communicating with a terminal. To ensure the receipt of the correct TOE-IC, the Smartcard Embedded Software (TOE-ES) shall provide the capability to check a sufficient part of the pre-personalisation data as identification feature of the TOE-IC. This shall include at least the Fabkey Data that is agreed between the TOE-ES developer and the TOE-IC Manufacturer. Key-dependent functions shall be implemented in the Smartcard Embedded Software in a way that they are not susceptible to leakage attacks.

4 Security Objectives

4.1 Security Objectives for the TOE

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

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

4.1.1 Security Objectives for the TOE-IC

The table below lists the security objectives for the TOE-IC. These security objectives concern only phase 7 of the product life-cycle and follow the details given in /BSI-PP-0002/, chap. 4.1, /ST-ICPhilips/, chap. 4.1 and the Security Target related to the evaluation of the IC incl. its Crypto Library.

Note:

For clarity, within the description of the security objectives in the following table as given in /BSI-PP-0002/ and /ST-ICPhilips/ the word „TOE“ is replaced by „TOE-IC“ and the term „Smartcard Embedded Software“ is supplemented by „TOE-ES“.

Security Objectives / TOE-IC	
Name	**Definition**
O.Leak-Inherent	**Protection against Inherent Information Leakage** The TOE-IC must provide protection against disclosure of confidential data (user data or TSF data) stored and/or processed in the Smartcard IC - by measurement and analysis of the shape and amplitude of signals (e.g. on the power, clock, or I/O lines) and - by measurement and analysis of the time between events found by measuring signals (e.g. on the power, clock, or I/O lines). This objective pertains to measurements with subsequent complex signal processing whereas O.Phys-Probing is about direct measurements on elements on the chip surface.
O.Phys-Probing	**Protection against Physical Probing** The TOE-IC must provide protection against disclosure of user data, against the disclosure/reconstruction of the IC Dedicated Software (TOE-IC) and the Smartcard Embedded Software (TOE-ES) or against the disclosure of other critical operational information. This includes protection against - measuring through galvanic contacts which is direct physical probing on the chips surface except on pads being bonded (using standard tools for meas-

	uring voltage and current) or - measuring not using galvanic contacts but other types of physical interaction between charges (using tools used in solid-state physics research and IC failure analysis) with a prior - reverse-engineering to understand the design and ist properties and functions. The TOE-IC must be designed and fabricated so that it requires a high combination of complex equipment, knowledge, skill, and time to be able to derive detailed design information or other information which could be used to compromise security through such a physical attack.
O.Malfunction	**Protection against Malfunctions** The TOE-IC must ensure its correct operation. The TOE-IC must prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent errors. The environmental conditions may include voltage, clock frequency, temperature, or external energy fields. Remark: A malfunction of the TOE-IC may also be caused using a direct interaction with elements on the chip surface. This is considered as being a manipulation (refer to the objective O.Phys-Manipulation) provided that detailed knowledge about the TOE-IC´s internal construction is required and the attack is performed in a controlled manner.
O.Phys-Manipulation	**Protection against Physical Manipulation** The TOE-IC must provide protection against manipulation of the TOE-IC (including its software and TSF data), the Smartcard Embedded Software (TOE-ES) and the user data. This includes protection against - reverse-engineering (understanding the design and its properties and functions), - manipulation of the hardware and any data, as well as - controlled manipulation of memory contents (user data). The TOE-IC must be designed and fabricated so that it requires a high combination of complex equipment, knowledge, skill, and time to be able to derive detailed design information or other information which could be used to compromise security through such a physical attack.
O.Leak-Forced	**Protection against Forced Information Leakage** The TOE-IC must be protected against disclosure of confidential data (user data or TSF data) processed in the card (using methods as described under O.Leak-Inherent) even if the information leakage is not inherent but caused by the attacker - by forcing a malfunction (refer to “Protection against Malfunction due to Environmental Stress (O.Malfunction)” and/or - by a physical manipulation (refer to “Protection against Physical Manipulation (O.Phys-Manipulation)”. If this is not the case, signals which normally do not contain significant information about secrets could become an information channel for a leakage attack.
O.Abuse-Func	**Protection against Abuse of Functionality**

	The TOE-IC must prevent that functions of the TOE-IC which may not be used after TOE-IC Delivery can be abused in order (i) to disclose critical user data of the IC Dedicated Software (TOE-IC) or of the Smartcard Embedded Software (TOE-ES), (ii) to manipulate critical user data of the IC Dedicated Software (TOE-IC) or of the Smartcard Embedded Software (TOE-ES), (iii) to manipulate the IC Dedicated Software (TOE-IC) or Soft-coded Smartcard Embedded Software (TOE-ES) or (iv) bypass, deactivate, change or explore security features or functions of the TOE-IC.
O.Identification	**TOE Identification** The TOE-IC must provide means to store Initialisation Data and Pre-personalisation Data in its non-volatile memory. The Initialisation Data (or parts of them) are used for TOE-IC identification.
O.RND	**Random Numbers** The TOE-IC will ensure the cryptographic quality of random number generation. For instance random numbers shall not be predictable and shall have a sufficient entropy. The TOE will ensure that no information about the produced random numbers is available to an attacker since they might be used for instance to generate cryptographic keys. The preceding points concern the random numbers generated by the hardware RNG as well as the random numbers generated by the software RNG.
O.HW_DES3	**Triple DES Functionality** The TOE-IC shall provide the cryptographic functionality to calculate a Triple DES encryption and decryption to the Smartcard Embedded Software (TOE-ES). The TOE-IC supports directly the calculation of Triple DES with two keys. Note: The TOE-IC will ensure the confidentiality of the user data (and especially cryptographic keys) during Triple DES operation. This is supported by O.Leak-Inherent.
O.MEM_ACCESS	**Area based Memory Access Control** Access by processor instructions to memory areas is controlled by the TOE-IC. The TOE-IC decides based on the Mode of Operation (System Mode, Meta Mode or User Mode) and the configuration of the Memory Management Units (MMU) if the requested type of access to the memory area addressed by the operands in the instruction is allowed.
O.SFR_ACCESS	**Special Function Register Access Control** The TOE-IC shall provide access control to the Special Function Registers based on the Mode of Operation (System Mode, Meta Mode or User Mode). The access control is used to restrict access to the Memory Management Units and all Specialised Components of the TOE-IC. The administration of the access conditions for ROM and EEPROM shall be restricted to code running in System Mode. The administration of the access conditions for RAM shall be restricted to code executed in System Mode or

	Meta Mode. The access to specialised hardware components of the TOE-IC shall be restricted to code running in System Mode or Meta Mode.
O.RANGE_CHK	**Value Range Check** The TOE-IC shall provide a range check for Special Pointer Registers. The range check comprises checking a lower and an upper bound for the value stored in the register. A violation of the allowed range shall interrupt the running code and allow an exception handling.
O.DES3	**DES3** The TOE-IC includes functionality to provide encryption and decryption facilities of the Triple-DES algorithm, resistant to SPA, DPA, DFA and timing attacks. This uses the hardware DES engine specified in the security objective O.HW_DES3.
O.RSA	**RSA** The TOE-IC includes functionality to provide public key facilities using the RSA algorithm, resistant to SPA, DPA, DFA and timing attacks.
O.SHA	**SHA-1** The TOE-IC includes functionality to provide electronic hashing facilities using the SHA-1 algorithm.
O.REUSE	**Object Reuse** The TOE-IC includes measures to ensure that the memory resources being used by the TOE cannot be disclosed to subsequent users of the same memory resource.

4.1.2 General Security Objectives for the TOE-ES

Nearly all security objectives mentioned in the table below concern the general security objectives for the TOE-ES as defined in /PP9911/, chap. 4.1. These security objectives are supplemented by security objectives drawn from /BSI-PP-0002/, chap. 4.2, /ST-ICPhilips/, chap. 4.2 and the Security Target related to the evaluation of the IC incl. its Crypto Library which will be in the current scope switched from assumptions resp. security objectives for the environment of the IC to security objectives for the TOE-ES. The complete set of security objectives for the TOE-ES is listed in the table below.

Note:

For clarity, within the description of the security objectives in the following table as indicated in /BSI-PP-0002/ and /ST-ICPhilips/ the word „TOE“ is replaced by „TOE-IC“ and the term „Smartcard Embedded Software“ is supplemented by „TOE-ES“.

Note:

In the following table, items of /PP9911/ which are common with /PP9806/ are indicated by a "*".

Security Objectives / TOE-ES	
Name	**Definition**
O.CLON* (PP9911)	**Cloning** The TOE functionality must be protected from cloning.
O.OPERATE* (PP9911)	**Correct Operation** The TOE must ensure continued correct operation of its security functions.
O.FLAW* (PP9911)	**Flaws** The TOE must not contain flaws in design, implementation or operation.
O.DIS_MEMORY* (PP9911)	**Disclosure of Memory Contents** The TOE shall ensure that sensitive information stored in memories is protected against unauthorized disclosure.
O.MOD_MEMORY* (PP9911)	**Modification of Memory Contents** The TOE shall ensure that sensitive information stored in memories is protected against any corruption or unauthorized modification.
O.TAMPER_ES (PP9911)	**Tampering of the Smartcard Embedded Software** The TOE must prevent tampering with its security critical parts. Security mechanisms have especially to prevent the unauthorized change of functional parameters, security attributes and secrets such as the life cycle sequence flags and cryptographic keys. The Smartcard Embedded Software must be designed to avoid interpretations of electrical signals from the hardware part of the TOE.
O.DIS_MECHANISM2 (PP9911)	**Disclosure of Security Mechanisms of the Smartcard Embedded Software** The TOE shall ensure that the Smartcard Embedded Software security mechanisms are protected against unauthorized disclosure.
O.Plat-Appl	**Usage of Hardware Platform** To ensure that the TOE-IC is used in a secure manner the Smartcard Embedded Software (TOE-ES) shall be designed so that the requirements from the following documents are met: - hardware data sheet for the TOE-IC, - TOE-IC application notes, - findings of the TOE-IC evaluation reports relevant for the Smartcard Embedded Software (TOE-ES).
O.Resp-Appl	**Treatment of User Data**

	Security relevant User Data (especially cryptographic keys) are treated by the Smartcard Embedded Software (TOE-ES) as required by the security needs of the specific application context. For example the Smartcard Embedded Software (TOE-ES) will not disclose security relevant user data to unauthorised users or processes when communicating with a terminal.
O.Check-Init	**Check of initialisation data by the Smartcard Embedded Software** To ensure the receipt of the correct TOE-IC, the Smartcard Embedded Software (TOE-ES) shall provide the capability to check a sufficient part of the pre-personalisation data as identification feature of the TOE-IC. This shall include at least the Fabkey Data that is agreed between the TOE-ES developer and the TOE-IC Manufacturer.
O.Key-Function	**Usage of Key-dependent Functions** Key-dependent functions shall be implemented in the Smartcard Embedded Software in a way that they are not susceptible to leakage attacks.

4.1.3 Tachograph Card Specific Security Objectives

The following table lists the specific security objectives relevant for the Tachograph Application. The security objectives are drawn from the Tachograph Card Specification /TachAn1B/, Appendix 10, Tachograph Card Generic Security Target, chap. 3.4 and 3.5 and are supplemented by an additional security objective for the personalisation of the TOE.

Security Objectives / TOE-ES (Tachograph Card Specific Security Objectives)	
Name	**Definition**
O.Card_Identification_Data	**Storage of Identification Data** The TOE must preserve card identification data and cardholder identification data stored during card personalisation process.
O.Card_Activity_Storage	**Storage of Activity Data** The TOE must preserve user data stored in the card by vehicle units.
O.Data_Access	**User Data Write Access** The TOE must limit user data write access rights to authenticated vehicle units.
O.Pers_Access	**Personalisation Data Write Access** The TOE must limit personalisation data write access rights to authenticated personalisation units.
O.Secure_Communications	**Secure Communications** The TOE must be able to support secure communication protocols and

	procedures between the card and the card interface device when required by the application.

4.2 Security Objectives for the Environment

4.2.1 General Security Objectives for the Environment of the TOE

Nearly all general security objectives for the environment of the TOE are defined in /PP9911/, chap. 4.2. These security objectives are supplemented by security objectives drawn from /BSI-PP-0002/, chap. 4.2, /ST-ICPhilips/, chap. 4.2 and the Security Target related to the evaluation of the IC incl. its Crypto Library, and a further specific security objective for the TOE´s personalisation.

All of these security objectives have to be fulfilled by organisational measures, thus they are security objectives for the Non-IT-Environment of the TOE. Security objectives for the IT-Environment of the TOE are not present.

The complete set of security objectives for the environment is listed in the table below.

Note:

For clarity, within the description of the security objectives in the following table as given in /BSI-PP-0002/ and /ST-ICPhilips/ the word „TOE“ is replaced by „TOE-IC“ and the term „Smartcard Embedded Software“ is supplemented by „TOE-ES“.

Note:

In the following table, items of /PP9911/ which are common with /PP9806/ are indicated by a “*”.

Security Objectives for the Environment of the TOE	
Name	**Definition**
Objectives on Phase 1	
O.DEV_TOOLS* (PP9911)	**Development Tools for the Smartcard Embedded Software** The Smartcard Embedded Software shall be designed in a secure manner, by using exclusively software development tools (compilers assemblers, linkers, simulators, etc.) and software-hardware integration testing tools (emulators) that will result in the integrity of program and data.
O.DEV_DIS_ES (PP9911)	**Development of the Smartcard Embedded Software** The Smartcard Embedded Software developer shall use established procedures to control storage and usage of the classified development tools and documentation, suitable to maintain the integrity and the confidentiality of the assets of the TOE.

	It must be ensured that tools are only delivered and accessible to the parties authorized personnel. It must be ensured that confidential information on defined assets are only delivered to the parties authorized personnel on a need to know basis.
O.SOFT_DLV* (PP9911)	**Protection of the Delivery of the Smartcard Embedded Software** The Smartcard Embedded Software must be delivered from the Smartcard Embedded Software developer (Phase 1) to the IC designer through a trusted delivery and verification procedure that shall be able to maintain the integrity of the software and its confidentiality, if applicable.
O.INIT_ACS (PP9911)	**Access to Initialisation Data** Initialisation Data shall be accessible only by authorized personnel (physical, personnel, organizational, technical procedures).
O.SAMPLE_ACS (PP9911)	**Access to Samples** Samples used to run tests shall be accessible only by authorized personnel.
Objectives on the TOE Delivery Process (Phases 4 to 7)	

O.DLV_PROTECT* (PP9911)	**Protection of the Delivery of TOE Material / Information**
	
	Procedures shall ensure protection of TOE material / information under delivery including the following objectives:
	- non-disclosure of any security relevant information
	- identification of the element under delivery
	- meet confidentiality rules (confidentiality level, transmittal form, reception acknowledgement)
	- physical protection to prevent external damage
	- secure storage and handling procedures (including rejected TOE's)
	- traceability of TOE during delivery including the following parameters:
	  - origin and shipment details
	  - reception, reception acknowledgement
	  - location material/information

O.DLV_AUDIT* (PP9911)	**Audit of Delivery** Procedures shall ensure that corrective actions are taken in case of improper operation in the delivery process (including if applicable any non-conformance to the confidentiality convention) and highlight all non-conformance to this process.
O.DLV_RESP* (PP9911)	**Responsibility** Procedures shall ensure that people (shipping department, carrier, reception department) dealing with the procedure for delivery have got the required skill, training and knowledge to meet the procedure requirements and be able to act fully in accordance with the above expectations.
Objectives on Deliv-	

ery from Phase 1 to Phases 4, 5 and 6	
O.DLV_DATA (PP9911)	**Delivery of Application Data** The Application Data must be delivered from the Smartcard Embedded Software developer (phase 1) either to the IC Packaging manufacturer, the Finishing Process manufacturer or the Personalizer through a trusted delivery and verification procedure that shall be able to maintain the integrity and confidentiality of the Application Data.
Objectives on Phases 4 to 6	
O.TEST_OPERATE* (PP9911)	**Testing of the TOE** Appropriate functionality testing of the TOE shall be used in phases 4 to 6. During all manufacturing and test operations, security procedures shall be used through phases 4, 5 and 6 to maintain confidentiality and integrity of the TOE and its manufacturing and test data.
O.Process-Card	**Protection during Packaging, Finishing and Personalisation** Security procedures shall be used after TOE-IC Delivery up to delivery to the end-user to maintain confidentiality and integrity of the TOE-IC and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use).
Objectives on Phase 6	
O.PERS	**Maintaining of Personalisation Data** The originator of the personalisation data and the personalisation center responsible for the personalisation of the TOE shall handle the personalisation data in an adequate secure manner. This concerns especially the security data to be personalised as secret cryptographic keys and PINs. The storage of the personalisation data at the originator and at the personalisation center as well as the transfer of these data between the different sides shall be conducted with respect to data integrity and confidentiality. Furthermore, the personalisation center shall treat the data for securing the personalisation process, i.e. the personalisation keys suitably secure. It is in the responsibility of the originator of the personalisation data to garantuee for a sufficient quality of the personalisation data, especially of the cryptographic material to be personalised. The preparation and securing of the personalisation data appropriate to the Tachograph Card´s structure and according to the TOE´s personalisation requirements is as well in the responsibility of the external world and shall be done with care.
Objectives on Phase 7	
O.USE_DIAG* (PP9911)	**Secure Communication** Secure communication protocols and procedures shall be used between the

	smartcard and the terminal.

5 IT Security Requirements

5.1 TOE Security Requirements

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

5.1.1 TOE Security Functional Requirements

The TOE security functional requirements (SFRs) define the functional requirements for the TOE using functional requirement components drawn from /CCPart2/, functional requirement components of /CCPart2/ with extension as well as self-defined functional requirement components (only for the IC with its IC Dedicated Software). This chapter contains the SFRs concerning the IC (TOE-IC) as well as the SFRs concerning the Smartcard Embedded Software (TOE-ES).

Note:

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

For the SFRs of the TOE-IC, the name convention for the SFRs follows the calling in the document /ST-ICPhilips/, chap. 5.1.1 and the Security Target related to the evaluation of the IC incl. its Crypto Library.

For the SFRs of the TOE-ES, the SFRs are numbered by taking the original name of the SFRs resp. its elements and adding “-x” for the x-th iteration.

5.1.1.1 TOE Security Functional Requirements for the IC (TOE-IC)

The following two sections give a survey of the SFRs of the TOE-IC as defined in /BSI-PP-0002/, chap. 5.1.1, 8.4, 8.5, 8.6, /ST-ICPhilips/, chap. 5.1.1 and the Security Target related to the evaluation of the IC incl. its Crypto Library.

Note:

For clarity, within the listings of the SFRs in the following tables the word „TOE“ as given in /BSI-PP-0002/ and /ST-ICPhilips/ is replaced by „TOE-IC“.

5.1.1.1.1 SFRs of the TOE-IC according to the IC Protection Profile

FAU **Security Audit**	
FAU_SAS **Audit Data Storage**	
FAU_SAS.1 **Audit Storage**	
FAU_SAS.1.1 The TSF shall provide [assignment: *authorized users*] with the capability to store [assignement: *list of audit information*] in the audit records. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: ---	**FAU_SAS.1:** **FAU_SAS.1.1** The TSF shall provide [**test personnel before TOE-IC Delivery**] with the capability to store [**the Initialisation Data and/or Prepersonalisation Data and/or supplements of the Smartcard Embedded Software**] in the audit records. Hierarchical to: No other components Dependencies: No dependencies Management: ---

FCS **Cryptographic Support**	
FCS_RND **Generation of Random Numbers**	Refer to Appendix.
FCS_RND.1 **Quality Metric for Random Numbers**	
FCS_RND.1.1 The TSF shall provide a mechanism to generate random numbers that meet [assignment: *a defined quality metric*]. Hierarchical to: No other components Dependencies: No dependencies Management:	**FCS_RND.1:** **FCS_RND.1.1** The TSF shall provide a mechanism to generate random numbers that meet [**the requirement to provide an entropy of at least 7 bit in each byte**]. Note: The entropy of the random number is measured by the Shannon-Entropy as follows: $E = -\sum_{i=0}^{255} p_i \cdot \log_2 p_i$, where p_i is the probability that the

--- Audit: ---	byte (b_7, b_6, ..., b_0) is equal to *i* as binary number. Here term “bit” means measure of the Shannon-Entropy. Hierarchical to: No other components Dependencies: No dependencies Management: ---

FDP **User Data Protection**	
FDP_IFC **Information Flow Control Policy**	
FDP_IFC.1 **Subset Information Flow Control**	
FDP_IFC.1.1 The TSF shall enforce the [assignment: *information flow control SFP*] on [assignment: *list of subjects*, *information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP*]. Hierarchical to: No other components Dependencies: - FDP_IFF.1 Simple security attributes Management: --- Audit: ---	**FDP_IFC.1:** **FDP_IFC.1.1** The TSF shall enforce the [**Data Processing Policy**] on [**all confidential data when they are processed or transferred by the TOE-IC or by the Smartcard Embedded Software**]. **Data Processing Policy:** User Data and TSF data shall not be accessible from the TOE-IC except when the Smartcard Embedded Software decides to communicate the User Data via an external interface. The protection shall be applied to confidential data only but without the distinction of attributes controlled by the Smartcard Embedded Software. Hierarchical to: No other components Dependencies: - FDP_IFF.1 Simple security attributes Management: ---
FDP_ITT **Internal TOE Transfer**	
FDP_ITT.1 **Basic Internal Transfer Protection**	

FDP_ITT.1.1 The TSF shall enforce the [assignment: *access control SFP(s) and/or information flow control SFP(s)*] to prevent the [selection: *disclosure, modification, loss of use*] of user data when it is transmitted between physically-separated parts of the TOE. Hierarchical to: No other components Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] Management: a) If the TSF provides multiple methods to protect user data during transmission between physically separated parts of the TOE, the TSF could provide a pre-defined role with the ability to select the method that will be used Audit: a) Minimal: Successful transfers of user data, including identification of the protection method used b) Basic: All attempts to transfer user data, including the protection method used and any errors that occurred	**FDP_ITT.1:** **FDP_ITT.1.1** The TSF shall enforce the [**Data Processing Policy**] to prevent the [**disclosure**] of user data when it is transmitted between physically-separated parts of the TOE-IC. **Refinement** The different memories, the CPU and other functional units of the TOE-IC (e.g. a cryptographic co-processor) are seen as physically-separated parts of the TOE-IC. Hierarchical to: No other components Dependencies: - [FDP_IFC.1 Subset information flow control] Management: a) If the TSF provides multiple methods to protect user data during transmission between physically separated parts of the TOE, the TSF could provide a pre-defined role with the ability to select the method that will be used

FMT **Security Management**	
FMT_LIM **Limited Capabilities and Availability**	
FMT_LIM.1 **Limited Capabilities**	
FMT_LIM.1.1 The TSF shall be designed in a manner that limits their capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced [assignment: Limited capability and availability policy]. Hierarchical to: No other components Dependencies: - FMT_LIM.2 Limited availability Management: ---	**FMT_LIM.1:** **FMT_LIM.1.1** The TSF shall be designed in a manner that limits their capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced: [**Deploying Test Features after TOE-IC Delivery does not allow User Data to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks**]. Hierarchical to: No other components

Audit: ---	Dependencies: - FMT_LIM.2 Limited availability Management: ---
FMT_LIM.2 **Limited Availability**	
FMT_LIM.2.1 The TSF shall be designed in a manner that limits their availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced [assignment: Limited capability and availability policy]. Hierarchical to: No other components Dependencies: - FMT_LIM.1 Limited capabilities Management: --- Audit: ---	**FMT_LIM.2:** **FMT_LIM.2.1** The TSF shall be designed in a manner that limits their availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced: [**Deploying Test Features after TOE-IC Delivery does not allow User Data to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks**]. Hierarchical to: No other components Dependencies: - FMT_LIM.1 Limited capabilities Management: ---

FPT **Protection of the TSF**	
FPT_FLS **Fail Secure**	
FPT_FLS.1 **Failure with Preservation of Secure State**	
FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [assignment: *list of types of failures in the TSF*]. Hierarchical to: No other components Dependencies: - ADV_SPM.1 Informal TOE security policy model Management:	**FPT_FLS.1:** **FPT_FLS.1.1** The TSF shall preserve a secure state when the following types of failures occur: [**exposure to operating conditions which may not be tolerated according to the requirement Limited fault tolerance (FRU_FLT.2) and where therefore a malfunction could occur**]. **Refinement** The term "failure" above also covers "circumstances".

--- Audit: a) Basic: Failure of the TSF	The TOE-IC prevents failures for the “circumstances” defined above. Hierarchical to: No other components Dependencies: - ADV_SPM.1 Informal TOE-IC security policy model Management: ---
FPT_ITT **Internal TOE TSF Data Transfer**	
FPT_ITT.1 **Basic Internal TSF Data Transfer Protection**	
FPT_ITT.1 The TSF shall protect TSF data from [selection: *disclosure, modification*] when it is transmitted between separate parts of the TOE. Hierarchical to: No other components Dependencies: No dependencies Management: a) management of the types of modification against which the TSF should protect b) management of the mechanism used to provide the protection of the data in transit between different parts of the TSF. Audit: ---	**FPT_ITT.1:** **FPT_ITT.1.1** The TSF shall protect TSF data from [**disclosure**] when it is transmitted between separate parts of the TOE-IC. **Refinement** The different memories, the CPU and other functional units of the TOE-IC (e.g. a cryptographic co-processor) are seen as separated parts of the TOE-IC. This requirement is equivalent to FDP_ITT.1 above but refers to TSF data instead of User Data. Therefore, it should be understood as to refer to the same *Data Processing Policy* defined under FDP_IFC.1 below. Hierarchical to: No other components Dependencies: No dependencies Management: a) management of the types of modification against which the TSF should protect b) management of the mechanism used to provide the protection of the data in transit between different parts of the TSF.
FPT_PHP **Physical Protection**	
FPT_PHP.3 **Resistance to Physical Attack**	
FPT_PHP.3.1	**FPT_PHP.3:**

The TSF shall resist [assignment: *physical tampering scenarios*] to the [assignment: *list of TSF devices / elements*] by responding automatically such that the TSP is not violated. Hierarchical to: No other components Dependencies: No dependencies Management: a) management of the automatic responses to physical tampering Audit: ---	**FPT_PHP.3.1** The TSF shall resist [**physical manipulation and physical probing**] to the [**TSF**] by responding automatically such that the TSP is not violated. **Refinement** The TOE-IC will implement appropriate measures to continuously counter physical manipulation and physical probing. Due to the nature of these attacks (especially manipulation) the TOE-IC can by no means detect attacks on all of its elements. Therefore, permanent protection against these attacks is required ensuring that the TSP could not be violated at any time. Hence, “automatic response” means here (i) assuming that there might be an attack at any time and (ii) countermeasures are provided at any time. Hierarchical to: No other components Dependencies: No dependencies Management: a) management of the automatic responses to physical tampering
FPT_SEP **Domain Separation**	
FPT_SEP.1 **TSF Domain Separation**	
FPT_SEP.1.1 The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects. **FPT_SEP.1.2** The TSF shall enforce separation between the security domains of subjects in the TSC. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: ---	**FPT_SEP.1:** **FPT_SEP.1.1** The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects. **FPT_SEP.1.2** The TSF shall enforce separation between the security domains of subjects in the TSC. **Refinement** Those parts of the TOE-IC which support the security functional requirements “Limited fault tolerance (FRU_FLT.2)” and “Failure with preservation of secure state (FPT_FLS.1)” shall be protected from interference of the Smartcard Embedded Software. Hierarchical to: No other components Dependencies: No dependencies

	Management: ---

FRU **Resource Utilisation**	
FRU_FLT **Fault Tolerance**	
FRU_FLT.2 **Limited Fault Tolerance**	
FRU_FLT.2.1 The TSF shall ensure the operation of all the TOE's capabilities when the following failures occur: [assignment: *list of type of failures*]. Hierarchical to: FRU_FLT.1 Dependencies: - FPT_FLS.1 Failure with preservation of secure state Management: --- Audit: a) Minimal: Any failure detected by the TSF	**FRU_FLT.2:** **FRU_FLT.2.1** The TSF shall ensure the operation of all the TOE-IC's capabilities when the following failures occur: [**exposure to operating conditions which are not detected according to the requirement failure with preservation of secure state (FPT_FLS.1)**]. **Refinement** The term "failure" above means "circumstances". The TOE-IC prevents failures for the "circumstances" defined above. Hierarchical to: FRU_FLT.1 Dependencies: - FPT_FLS.1 Failure with preservation of secure state Management: ---

5.1.1.1.2 Additional SFRs of the TOE-IC (Hardware)

The TOE-IC uses a single Security Function Policy as defined as follows:

Access Control Policy / TOE-IC

The hardware shall provide different modes of operation to a Smartcard Embedded Software. The management of access to code and data as well as the configuration of the hardware shall be performed in a dedicated mode. The hardware shall provide a mode that ensures the separation between different applications running on the TOE-IC. An application shall not be able to access Specialised Components directly to support the separation of

applications. The functions used by the IC Dedicated Test Software to test the chip shall not be available to the Smartcard Embedded Software.

The Security Function Policy (SFP) **Access Control Policy** uses the following definitions:

The subjects are

- **Smartcard Embedded Software** i.e. data in the memories of the TOE-IC executed as instructions by the CPU

The objects are

- the **memories** (ROM, EEPROM and RAM) consisting of
  - the **data in the Code Memory Areas** defined by the Code Memory Management Unit (Code MMU) in ROM and EEPROM
  - the **data in the Data Memory Areas** defined by the Data Memory Management Unit (Data MMU) in RAM
- the **Special Function Registers** consisting of
  - Special Function Registers to configure the Code MMU
  - Special Function Registers to configure the Data MMU
  - Special Function Registers related to Specialised Components
  - Special Function Registers related to General CPU Functions

The memory operations are

- **read** data from the memory,
- **write** data into the memory and
- **execute** data in the memory.

The Special Function Register operations are

- **read** data from a Special Function Register and
- **write** data into a Special Function Register.

(The read and/or write access to a Special Function Register may be not allowed depending on the function of the register, on the mode of operation or on the TOE-IC mode to enforce the access control policy or ensure a secure operation.)

The security attributes are:

- **Mode of Operation**: There are three different mode of operation based on the configuration of the Special Function Register “Program Status Word” defining whether the instruction is executed in the System Mode, Meta Mode and User Mode.
- **TOE-IC mode**: The TOE-IC mode depends on the life cycle phase of the TOE-IC. For the production test the Test Mode is used. After the production test within the usage phase the Application Mode is used. It is not possible to switch back from the Application Mode into the Test Mode. Both modes provide the three different modes System Mode, Meta Mode and User Mode in the same way.

- **Special Function Registers to configure the Code MMU**: Configuration of the Code MMU comprising access rights (read, write, execute and enabled/disabled), the virtual code memory base address of the first and last valid block, and the relocation offset to the physical memory location for each of 12 possible Code Memory Areas.
- **Special Function Registers to configure the Data MMU**: Configuration of the Data MMU comprising access rights (read, write and enabled/disabled), the virtual data memory base address of the first and last valid block, and the relocation offset to the physical memory location for each of 4 possible Data Memory Areas.

The operation “enabled/disabled” of the Code MMU and Data MMU does not define a new operation beyond read, write and execute, but is an implementation detail that allows faster context switching. In “enabled” Code/Data Memory Areas, the MMU uses the values of all other attributes to allow or to deny access. In “disabled” Code/Data Memory Areas, the MMU only denies any access regardless of the values of all other attributes.

Note: A Code/Data Memory Area will be disabled for use if the virtual code/data memory base address of the last valid block is lower than the address of first valid block.

In the following further SFRs of the TOE-IC are listed:

FCS **Cryptographic Support**	
FCS_COP **Cryptographic Operation**	
FCS_COP.1 **Cryptographic Operation**	
FCS_COP.1.1 The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: --- Audit:	**FCS_COP.1:** **FCS_COP.1.1** The TSF shall perform [**encryption and decryption**] in accordance with a specified cryptographic algorithm [**Triple Data Encryption Algorithm (TDEA)**] and cryptographic key sizes [**of 112 bit**] that meet the following [**list of standards**: **FIPS PUB 46-3 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION DATA ENCRYPTION STANDARD (DES) Reaffirmed 1999 October 25, keying option 2**] Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes

a) Minimal: Success and failure, and the type of cryptographic operation b) Basic: Any applicable cryptographic mode(s) of operation, subject attributes and object attributes	Management: ---

FDP **User Data Protection**	
FDP_ACC **Access Control Policy**	
FDP_ACC.1 **Subset Access Control**	
FDP_ACC.1.1 The TSF shall enforce the [assignment: *access control SFP*] on [assignment: *list of subjects*, *objects, and operations among subjects and objects covered by the SFP*]. Hierarchical to: No other components Dependencies: - FDP_ACF.1 Security attribute based access control Management: --- Audit: ---	**FDP_ACC.1[MEM]:** **FDP_ACC.1.1[MEM]** The TSF shall enforce the [**Access Control Policy**] on [**all code running on the TOE-IC, all memories and all memory operations**]. **Application Note** The Access Control Policy shall be enforced by implementing a Code MMU and a Data MMU, which map virtual addresses to physical addresses. The CPU always uses virtual addresses, which are mapped to physical addresses by the MMU. Prior to accessing the respective memory at the physical address, the respective MMU checks if the access is allowed. Hierarchical to: No other components Dependencies: - FDP_ACF.1[MEM] Security attribute based access control Management: ---
	FDP_ACC.1[SFR]: **FDP_ACC.1.1[SFR]** The TSF shall enforce the [**Access Control Policy**] on [**all code running on the TOE-IC, all Special Function Registers, and all SFR operations**]. Hierarchical to: No other components Dependencies: - FDP_ACF.1[SFR] Security attribute based access control

	Management: ---
FDP_ACF **Access Control Functions**	
FDP_ACF.1 **Security Attribute Based Access Control**	
FDP_ACF.1.1 The TSF shall enforce the [assignment: *access control SFP*] to objects based on [assignment: *security attributes*, *named groups of security attributes*]. **FDP_ACF.1.2** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]. **FDP_ACF.1.3** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]. **FDP_ACF.1.4** The TSF shall explicitly deny access of subjects to objects based on the [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]. Hierarchical to: No other components Dependencies: - FDP_ACC.1 Subset access control - FMT_MSA.3 Static attribute initialisation Management: a) Managing the attributes used to make explicit access or denial based decisions Audit: a) Minimal: Successful requests to perform an operation on an object covered by the SFP b) Basic: All requests to perform an operation on an object covered by the SFP c) Detailed: The specific security attributes used in making an access check	**FDP_ACF.1[MEM]:** **FDP_ACF.1.1[MEM]** The TSF shall enforce the [**Access Control Policy**] to objects based on [**the Mode of Operation, the Special Function Registers to configure the Code MMU and the Special Function Registers to configure the Data MMU**]. **FDP_ACF.1.2[MEM]** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**Code executed in the System Mode** - **has read and execute access to all code/data in User-ROM,** - **has read, write and execute access to all code/data in EEPROM,** - **read and write access to all data in RAM** **Code executed in the Meta Mode** - **read and/or execute access to code/data in the User-ROM controlled by the Code MMU,** - **has read, write and/or execute access to code/data in the EEPROM controlled by the Code MMU,** - **has read and write access to all data in RAM controlled by the Data MMU** **Code executed in the User Mode** - **has read and/or execute access to code/data in the User-ROM controlled by the Code MMU,** - **has read and/or write and/or execute access to code/data in the EEPROM controlled by the Code MMU,** - **has read and/or write access to data in RAM controlled by the Data MMU**]. **FDP_ACF.1.3[MEM]** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**Code running in System Mode has unrestricted access to all memories**]. **FDP_ACF.1.4[MEM]** The TSF shall explicitly deny access of subjects to objects based on the rules: [**disabled Code/Data area**].

	Application Note The explicitly authorised access according to FDP_ACF.1.3 shall be implemented in System Mode by switching the Code and the Data MMU to a transparent behaviour, which means that the virtual address is mapped one-to-one to the physical address without controlling access to the address. Hierarchical to: No other components Dependencies: - FDP_ACC.1[MEM] Subset access control - FMT_MSA.3[MEM] Static attribute initialisation Management: a) Managing the attributes used to make explicit access or denial based decisions
	FDP_ACF.1[SFR]: **FDP_ACF.1.1[SFR]** The TSF shall enforce the [**Access Control Policy**] to objects based on [**the Mode of Operation and the TOE-IC mode**]. **FDP_ACF.1.2[SFR]** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**(i) The code executed in System Mode is allowed to access all Special Function Registers.** **(ii) The code executed in the Meta Mode is allowed to access all Special Function Registers except the Special Function Registers to configure the Code MMU where only read access is allowed.** **(iii) The code executed in the User Mode is only allowed to access the Special Function Registers related to General CPU Functions**]**.** **FDP_ACF.1.3[SFR]** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**Within the Test Mode the IC Dedicated Test Software running in the System Mode or in the Meta Mode is allowed to read and write additional SFR for test purposes**]. **FDP_ACF.1.4[SFR]** The TSF shall explicitly deny access of subjects to objects based on the rules: [**Smartcard Embedded Software executed in the Meta Mode is not allowed to write the SCR Register. Smartcard Embedded Software executed in the System Mode or Meta Mode is not allowed to directly modify the bits 6 and 7 of the PSWH Register. Within the Application Mode the Smartcard Embedded Software executed in any mode of operation is not allowed**

	to read and write additional SFR for test purposes]. **Application Note** The access control is enforced regardless of the access as 16-bit or 8-bit Special Function Register. Modifying bits 6 and 7 of the PSWH Register with an 8-bit access is equivalent to modifying bits 14 and 15 of the PSW Registers with an 16-bit access. Hierarchical to: No other components Dependencies: - FDP_ACC.1[SFR] Subset access control - FMT_MSA.3[SFR] Static attribute initialisation Management: a) Managing the attributes used to make explicit access or denial based decisions

FMT **Security Management**	
FMT_MSA **Management of Security Attributes**	
FMT_MSA.1 **Management of Security Attributes**	
FMT_MSA.1.1 The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] to restrict the ability to [selection: *change_default, query, modify, delete,* [assignment: *other operations*]] the security attributes [assignment: *list of security attributes*] to [assignment: *the authorised identified roles*]. 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 Management: a) managing the group of roles that can interact with the security attributes Audit: a) Basic: All modifications of the values of security attributes	**FMT_MSA.1[MEM]:** **FMT_MSA.1.1[MEM]** The TSF shall enforce the [**Access Control Policy**] to restrict the ability to [**modify**] the security attributes [**Special Function Registers to configure the Code MMU and Special Function Registers to configure the Data MMU**] to [**code executed in the System Mode or Meta Mode**]. **Application Note** Code executed in the System Mode is able to modify the Special Function Registers to configure the Code MMU and the Special Function Registers to configure the Data MMU, code executed in the Meta Mode is only able to modify the Special Function Registers to configure the Data MMU. Hierarchical to: No other components Dependencies: - [FDP_ACC.1[MEM] Subset access control] - FMT_SMR.1 Security roles

	- **FMT_SMF.1 Specification of management functions** Management: a) managing the group of roles that can interact with the security attributes
	FMT_MSA.1[SFR]: **FMT_MSA.1.1[SFR]** The TSF shall enforce the [**Access Control Policy**] to restrict the ability to [**modify**] the security attributes [**Mode of Operation**] to [**the hardware executed on behalf of an exception or interrupt**]. **Application Note** The Mode of Operation is coded in the register Program Status Word.The relevant bits 6 and 7 of the PSWH can only be changed directly by the hardware based on an exception or interrupt. Hierarchical to: No other components Dependencies: - [FDP_ACC.1[SFR] Subset access control] - FMT_SMR.1 Security roles - **FMT_SMF.1 Specification of management functions** Management: a) managing the group of roles that can interact with the security attributes
FMT_MSA.3 **Static Attribute Initialisation**	
FMT_MSA.3.1 The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] to provide [selection: *restrictive, permissive, other property*] default values for security attributes that are used to enforce the *SFP*. **FMT_MSA.3.2** The TSF shall allow the [assignment: *the authorised identified roles*] to specify alternative initial values to override the default values when an object or information is created. Hierarchical to: No other components Dependencies: - FMT_MSA.1 Management of security attributes - FMT_SMR.1 Security roles Management:	**FMT_MSA.3[MEM]:** **FMT_MSA.3.1[MEM]** The TSF shall enforce the [**Access Control Policy**] to provide [**permissive**] default values for security attributes that are used to enforce the SFP. **FMT_MSA.3.2[MEM]** The TSF shall allow [**the Smartcard Embedded Software**] to specify alternative initial values to override the default values when an object or information is created. **Application Note** Permissive means here that the reset values of the Special Function Register do not provide any restrictions. The SFR of the Code/Data memory management units must be configured after reset by the Smartcard Embedded Software. In addition the Smartcard Embedded Software must define and maintain security attributes for all objects generated

a) managing the group of roles that can specify initial values b) managing the permissive or restrictive setting of default values for a given access control SFP Audit: a) Basic: Modifications of the default setting of permissive or restrictive rules b) Basic: All modifications of the initial values of security attributes	by it. Hierarchical to: No other components Dependencies: - FMT_MSA.1[MEM] Management of security attributes - FMT_SMR.1 Security roles Management: a) managing the group of roles that can specify initial values b) managing the permissive or restrictive setting of default values for a given access control SFP
	FMT_MSA.3[SFR]: **FMT_MSA.3.1[SFR]** The TSF shall enforce the [**Access Control Policy**] to provide [**restrictive**] default values for security attributes that are used to enforce the SFP. **FMT_MSA.3.2[SFR]** The TSF shall allow [**no subject**] to specify alternative initial values to override the default values when an object or information is created. **Application Note** Here restrictive means that all exceptions including the reset are set up by the hardware in System Mode with disabled MMU for Code and data. Thereby the selection of the dedicated entry in the vector table and the complete control of the TOE-IC is ensured. Nevertheless the developer of the Smartcard Embedded Software is able to run the assigned exception routine in any Mode of Operation as configured in the vector table. Hierarchical to: No other components Dependencies: - FMT_MSA.1[SFR] Management of security attributes - FMT_SMR.1 Security roles Management: a) managing the group of roles that can specify initial values b) managing the permissive or restrictive setting of default values for a given access control SFP
FMT_SMF **Specification of Management Functions**	
FMT_SMF.1 **Specification of Management Functions**	

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions:[*list of security management functions to be provided by the TSF*]. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: a) Minimal: Use of the management functions	**FMT_SMF.1:** **FMT_SMF.1.1** The TSF shall be capable of performing the following security management functions:[- **Change of the Mode of Operation by invoking an exception or interrupt** - **Change of the Mode of Operation by finishing an interrupt (with a RETI instruction executed in System Mode or Meta Mode)** - **Modification of the Code Memory Management Unit Attributes** - **Modification of the Data Memory Management Unit Attributes** - **Modification of the clock settings and power configuration**] **Application Note** A separation of the Specification of Management Functions based on the iterations used for FMT_MSA.1 that requires this security functional requirement is not needed because all management functions rely on the same features implemented in the hardware. The Mode of Operation is changed at the time the interrupt is invoked by loading a new value for the PSW Register from the interrupt vector table. Similarly, at the time the interrupt is finished by executing the RETI instruction, a previously saved value for the PSW Register is loaded from the stack. Hierarchical to: No other components Dependencies: No dependencies Management: ---

FRU **Resource Utilisation**	
FRU_VRC **Value Range Checking**	
FRU_VRC.1 **Simple Value Range Check**	
FRU_VRC.1.1 The TSF shall enforce a range checking for the value of the following resources: [assignment: *controlled*	**FRU_VRC.1:** **FRU_VRC.1.1**

resources]. **FRU_VRC.1.2** The TSF shall notify [assignment: *the authorised identified roles]* that the value is out of the defined range. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: ---	The TSF shall enforce a range checking for the value of the following resources: [**R15/AP1 CPU register and CSFVAL Special Function Register**]. **FRU_VRC.1.2** The TSF shall notify [**the related exception**] that the value is out of the defined range. Hierarchical to: No other components Dependencies: No dependencies Management: ---

5.1.1.1.3 Additional SFRs of the TOE-IC (IC Dedicated Support Software)

FCS **Cryptographic Support**	
FCS_COP **Cryptographic Operation**	
FCS_COP.1 **Cryptographic Operation**	
FCS_COP.1.1 The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: ---	**FCS_COP.1+1:** **FCS_COP.1.1+1** The TSF shall perform [**encryption and decryption**] in accordance with the specified cryptographic algorithm [**Triple-DES with "outer" CBC mode or ECB mode**] and cryptographic key sizes [**double-length (112 bit) or triplelength (168 bit)**] that meet the following: [**standard ANSI X9.52-1998**]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management:

Audit: a) Minimal: Success and failure, and the type of cryptographic operation b) Basic: Any applicable cryptographic mode(s) of operation, subject attributes and object attributes	---
	FCS_COP.1+2: **FCS_COP.1.1+2** The TSF shall perform [**encryption and decryption**] in accordance with the specified cryptographic algorithm [**RSA**] and cryptographic key sizes [**1024 bits to 2048 bits**] that meet the following: [**as described in B. Schneier, Angewandte Kryptographie, page 468, or Menezes, van Oorshot and Vanstone, Handbook of Applied Cryptography, section 8.2, and also mentioned in the standard ISO/IEC 9796, Annex A, section A.4**]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: ---
	FCS_COP.1+4: **FCS_COP.1.1+4** The TSF shall perform [**cryptographic checksum generation**] in accordance with the specified cryptographic algorithm [**SHA-1**] and cryptographic key size [**none**] that meet the following: [**standard FIPS 180-1**]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: ---
FCS_RND **Generation of Random Numbers**	Refer to Appendix.

FCS_RND.2 **Random Number Generation**	
FCS_RND.2.1 The TSF shall provide a mechanism to generate random numbers that meet the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: ---	**FCS_RND.2:** **FCS_RND.2.1** The TSF shall provide a mechanism to generate random numbers that meet the following: [**standard FIPS 186-2 with Change Notice 1**]. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: ---

FDP **User Data Protection**	
FDP_IFC **Subset Information Flow Policy**	
FDP_IFC.1 **Subset Information Flow Control**	
	FDP_IFC.1 as defined in chap. 5.1.1.1.1 (with extension to resistance leakage attacks)
FDP_ITT **Internal TOE Transfer**	
FDP_ITT.1 **Basic Internal Transfer Protection**	
	FDP_ITT.1 as defined in chap. 5.1.1.1.1 (with extension to resistance leakage attacks)
FDP_RIP **Residual Information Protection**	
FDP_RIP.1 **Subset Residual Information Protection**	
FDP_RIP.1.1 The TSF shall ensure that any previous information	**FDP_RIP.1:**

content of a resource is made unavailable upon the [selection: *allocation of the resource to*, *deallocation of the resource from*] the following objects: [assignment: *list of objects*]. Hierarchical to: No other components Dependencies: No dependencies Management: a) The choice of when to perform residual information protection (i.e. upon allocation or deallocation) could be made configurable within the TOE Audit: ---	**FDP_RIP.1.1** 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: [**all objects used by the Crypto Library as specified in the user guidance documentation**]. **Note** Memory areas, the content of which is explicitly documented and described in the user guidance, need not be cleared by the crypto library. The TSF ensures that, upon exit from each function, with the exception of input parameters, return values or locations where it is explicitly documented that values remain at specific addresses, any memory resources used by that function that contained temporary or secret values are cleared. Hierarchical to: No other components Dependencies: No dependencies Management: Not applicable

FPT **Protection of the TSF**	
FPT_FLS **Fail Secure**	
FPT_FLS.1 **Failure with Preservation of Secure State**	
FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [assignment: *list of types of failures in the TSF*]. Hierarchical to: No other components Dependencies: - ADV_SPM.1 Informal TOE security policy model Management: --- Audit: b) Basic: Failure of the TSF	**FPT_FLS.1:** (compare with FPT_FLS.1 of chap. 5.1.1.1.1) **FPT_FLS.1.1** The TSF shall preserve a secure state when the following types of failures occur: [**(i) exposure to operating conditions which may not be tolerated according to the requirement Limited fault tolerance (FRU_FLT.2) and where therefore a malfunction could occur, (ii) DFA attacks on RSA and DES**]. **Refinement** The term “failure” above also covers “circumstances”. The TOE prevents failures for the “circumstances” defined above. This refinement should be understood with the following implementation details in mind: The TOE contains both hardware sensors (implemented in the chip card hardware) and software sensors (im-

	plemented in the Crypto Library software). The software sensors detect DFA attacks in RSA and DES computations (by verifying RSA private key calculations and by double computation of DES calculations) and lead to a secure state (no computation results are output) in case such an attack occurs. Hierarchical to: No other components Dependencies: - ADV_SPM.1 Informal TOE-IC security policy model Management: ---
FPT_ITT **Internal TOE TSF Data Transfer**	
FPT_ITT.1 **Basic Internal TSF Data Transfer Protection**	
	FPT_ITT.1 as defined in chap. 5.1.1.1.1 (with extension to resistance leakage attacks)
FPT_TST **TSF Self Test**	Refer to Appendix.
FPT_TST.2 **Subset TOE Security Testing**	
FPT_TST.2.1 The TSF shall run a suite of self tests [selection: *during initial start-up, periodically during normal operation, at the request of the authorised user, and/or at the conditions* [assignment: *conditions under which self test should occur*]] to demonstrate the correct operation of [assigment: *functions and/or mechanisms*]. Hierarchical to: No other components Dependencies: - FPT_AMT.1 Abstract machine testing Management: a) management of the conditions under which TSF self testing occurs, such as during initial start-up, regular interval, or under specified conditions b) management of the time interval if appropriate Audit: a) Basic: Execution of the TSF self tests and the results of the tests	**FPT_TST.2:** **FPT_TST.2.1** The TSF shall run a suite of self tests [**at the request of the authorised user**] to demonstrate the correct operation of [**the hardware RNG (F.RNG)**]. **Refinement** The *authorized user* is the technical user using the Crypto Library (typically this will be the smart card operating system). The (assigned) *mechanism* to be tested here is the hardware RNG (F.RNG). The hardware RNG is used to seed the software RNG (F.RNG_Access), and therefore has to be tested in advance: Since it is absolutely necessary to guarantee the quality of the seed, a suitable online test has to be performed before the seeding, i.e. the *suite of self tests* is an appropriate **online test**. Since the Crypto Library is not invoked automatically at start-up, the operating system has to ensure that the test routine is called before seed from the hardware RNG is taken for the software RNG, i.e. before the software RNG is initialized. This is what is intended by "*at the request of the authorized user*". In addition to the online test mentioned above, the Crypto Library may

	implement other test(s), the use of which depends on the intended application. For example, if the hardware RNG is to be used for re-seeding, a more simple test may be sufficient to ensure correct operation of the RNG. **Note** It is assumed that the hardware RNG is not used for other purposes than seeding and possibly re-seeding the software RNG. The user of the Crypto Library (the operating system) is assumed to use random bits produced by the software RNG whenever he needs random bits. However, the Crypto Library cannot prevent the operating system from accessing the hardware RNG. If the hardware RNG is to be used by the operating system directly, it has to be decided based on the operating system's and the application's security needs, what kind of test has to be performed first and what requirements will have to be applied for this test. In this case the Guidance, Delivery and Operation Manual for the Philips P16WX064V0C Secure 16-bit Smart Card Controller (Smart*XA*2) should be read carefully. Hierarchical to: No other components Dependencies: - FPT_AMT.1 Abstract machine testing Management: Not applicable

5.1.1.2 TOE Security Functional Requirements for the Smartcard Embedded Software (TOE-ES)

The following section gives a survey of the SFRs concerning the TOE´s Smartcard Embedded Software as required in the Tachograph Card Specification /TachAn1B/, Appendix 10 (Tachograph Card Generic Security Target) and the JIL interpretation /JILDigTacho/, Annex B.

5.1.1.2.1 Security Function Policies

The Tachograph Card distinguishes between two different phases, more precise between the personalisation phase and the end-usage (operational) phase, each of it with its own security functional policy (SFP). The SFPs for these different phases of the Tachograph Card will be described in detail in the following.

For a **non-personalised** Tachograph Card, the TOE-ES maintains a Security Function Policy as defined as follows:

SFP Personalisation Access Control (PERS-AC_SFP)

The SFP PERS-AC_SFP is only relevant for the personalisation phase of the Tachograph Card, i.e. after the initialisation of the card has been completed and no personalisation has been conducted.

Subjects:

- personalisation unit
- other card interface devices

Security attributes for subjects:

- USER_GROUP
  (PERSO_UNIT, NON_PERSO_UNIT)

Objects:

- data fields for user data as:
  - identification data (card identification data, cardholder identification data)
  - activity data (cardholder activities data, events and faults data, control activity data)
- data fields for security data as:
  - card´s private signature key (within the Tachograph Applet)
  - public keys (in form of certificates or other forms)
  - PIN (only workshop card)
- security data (loaded during initialisation or negotiated during personalisation):
  - card´s private personalisation key (within the Tachograph Applet)
  - personalisation unit´s public personalisation key (within the Tachograph Applet)
  - session keys
  - card´s private authentication key (within the Card Manager Applet)
- TOE software code
- TOE file system (incl. file structure, additional internal structures, access conditions)
- identification data of the TOE concerning the IC and the Smartcard Embedded Software (within the Card Manager Applet)
- data field for identification data of the TOE´s personalisation concerning the date and time of the personalisation (within the Card Manager Applet)

Security attributes for objects:

Access Rules for data fields for user data (see below).

Operations (Access Modes):

- data fields for user data as:
  - identification data: selecting (command Select File), writing (command Update Binary)
  - activity data: selecting (command Select File), writing (command Update Binary)
- data fields for security data as:
  - card´s private signature key: Loading (command Put Key)
  - public keys (in form of certificates or other forms): Loading (command Put Key)
  - PIN (only workshop card): Loading (command Put Key)
- security data:
  - card´s private personalisation key: internal authentication (command Internal Authenticate), external authentication (command External Authenticate)
  - personalisation unit´s public personalisation key: referencing over a MSE-command (for further use within cryptographic operations as authentication)
  - session keys: securing of commands with Secure Messaging
  - card´s private authentication key: internal authentication (command Internal Authenticate)
- TOE software code: ---
- TOE file system (incl. file structure, additional internal structures, access conditions): ---
- identification data of the TOE: reading (command Get Data)
- data field for identification data of the TOE´s personalisation (date and time of personalisation): writing (command Store Data), reading (command Get Data)

The SFP PERS-AC_SFP controls the access of subjects to **security data,** which are loaded during initialisation or are negotiated during personalisation data, by an implicit connection with the respective command. There are no further restrictions for the execution of the above mentioned access modes concerning secret data.

The SFP PERS-AC_SFP controls the access of subjects to the **data fields for security data** for personalisation purposes as follows: The loading of the secrets is only possible with Secure Messaging whereby this requires a preceding mutual authentication process with the card´s and personalisation unit´s personalisation keys which leads to session key negotiation (incl. send sequence counter). Hereby, the securing of the command is done with encryption and MAC-securing. Furthermore, the loading of the card´s private signature key is connected in an atomar process with the change of the Tachograph Card´s status from „initialised statuts“ to „operational status“. Afterwards, the personalisation commands are no longer available, and from now on only the SFP Access Control (AC_SFP) as loaded in the framework of the initialisation is relevant.

The SFP PERS-AC_SFP controls the access of subjects to **identification data of the TOE** over the corresponding command (Get Data). There are no further restrictions for the access to this data area.

The SFP PERS-AC_SFP controls the access of subjects to the **data field for identification data of the TOE´s personalisation** over the corresponding command (Store Data). There are no further restrictions for the access to this data area.

The SFP PERS-AC_SFP controls the access of subjects to the **data fields for user data** on the basis of security attributes.

For user data, the TOE maintains the following **type of security attributes**:

- Access Rule (AR) consisting of one or more Access Modes (AM) and a single Access Condition (AC)

The AM indicates the command type for accessing the object. The AC defines the conditions under which a command executed by a subject is allowed to access the object.

The access modes to objects of type user data have been defined above. Further, the TOE maintains the following **types of elementary ACs**:

- **AUT (Key based user authentication)**
  The right corresponding to a successful external key based authentication must be opened up (done by the command External Authenticate) before the command can be executed.
- **PRO SM (Secure Messaging providing data integrity and authenticity)**
  The command must be secured with a cryptographic checksum using secure messaging as defined in the Tachograph Card Specification (/TachAn1B/, Appendix 11, chap. 5, Appendix 2, chap. 3.6.2.2, 3.6.3.2).
- **ENC SM (Secure Messaging providing data confidentiality)**
  The command must be secured with an encryption using secure messaging as defined in the Tachograph Card Specification (/TachAn1B/, Appendix 11, chap. 5, Appendix 2, chap. 3.6.2.2, 3.6.3.2).

The above mentioned elementary AC elements can be combined according to the rules defined in ISO/IEC 7816-9 (ACs in expanded format). The object's AC assigned to an AM comprises then a logical expression of elementary AC elements using logical AND.

For each type of Tachograph Card the access rules for the different data fields for the user data and access modes are implemented as follows: The personalisation of the Tachograph Card´s data fields for the user data by using the access modes selecting and writing is only possible with Secure Messaging whereby this requires a preceding mutual authentication process with the card´s and the personalisation unit´s personalisation keys which leads to session key negotiation (incl. send sequence counter). More precise, each personalisation command is combined with the elementary AC elements AUT, PRO SM and ENC SM (logical AND), thus personalisation is only possible in a secured mode with encryption and MAC-securing using the negotiated session key and send sequence counter.

Generally, an object of type user data can only be accessed if an access mode exists and an access rule has been attached to the object (during its creation).

Additionally, for rule decisions the PERS-AC_SFP uses the actual security status set in the card as reference value.

The PERS-AC_SFP explicitly authorises access of subjects to objects based on the following rules:

- The TSF allows access to an object for a defined access mode, if the object's access condition is valid for this access mode.
- The TSF evaluates within an AC the logical expression of elementary AC elements (boolean expression) according to the following rules:
  - AC element AUT is set to "true", if AUT complies with the actual security status (preceding external authentication has been conducted successfully)
  - AC element SM PRO is set to "true", if SM PRO complies with the user indication for SM PRO and SM PRO complies with the actual security status (preceding internal-external authentication has been conducted successfully).
  - AC element SM ENC is set to "true", if SM ENC complies with the user indication for SM ENC and SM ENC complies with the actual security status (preceding internal-external authentication has been conducted successfully).

For a **personalised** Tachograph Card, the TOE-ES maintains a Security Function Policy as defined as follows:

SFP Access Control (AC_SFP)

The SFP AC_SFP is only relevant for the end-usage phase of the Tachograph Card, i.e. after the personalisation of the card has been completed and the Tachograph Application is in the „operational status“.

Subjects:

- vehicle units (in sense of the Tachograph Card specification)
- other card interface devices (non-vehicle units)

Security attributes for subjects:

- USER_GROUP
  (VEHICLE_UNIT, NON_VEHICLE_UNIT)
- USER_ID
  (Vehicle Registration Number (VRN) and Registering Member State Code (MSC), where USER_ID is only known to USER_GROUP = VEHICLE_UNIT)

Objects:

- user data:

  - identification data (card identification data, cardholder identification data)
  - activity data (cardholder activities data, events and faults data, control activity data)
- security data:
  - card´s private signature key (within the Tachograph Applet)
  - public keys (stored permanently on the card, imported into the card in form of certificates)
  - session keys
  - PIN (only workshop card)
  - card´s private authentication key (within the Card Manager Applet)
- TOE software code
- TOE file system (incl. file structure, additional internal structures, access conditions)
- identification data of the TOE concerning the IC and the Smartcard Embedded Software (within the Card Manager Applet)
- identification data of the TOE´s personalisation concerning the date and time of the personalisation (within the Card Manager Applet)

Security attributes for objects:

Access Rules for user data (see below).

Operations (Access Modes):

- user data:
  - identification data: selecting (command Select File), reading (command Read Binary), download function (command PSO Compute Digital Signature)
  - activity data: selecting (command Select File), reading (command Read Binary), writing / modification (command Update Binary), download function (command PSO Compute Digital Signature)
- security data:
  - card´s private signature key: generation of a digital signature (command PSO Compute Digital Signature), internal authentication (command Internal Authenticate), external authentication (command External Authenticate)
  - public keys: referencing over a MSE-command (for further use within cryptographic operations as authentication, verification of a digital signature etc.)
  - session keys: securing of commands with Secure Messaging
  - PIN (only workshop card): verification (command Verify)
  - card´s private authentication key: internal authentication (command Internal Authenticate)
- TOE software code: ---
- TOE file system (incl. file structure, additional internal structures, access conditions): ---

- identification data of the TOE: reading (command Get Data)
- identification data of the TOE´s personalisation (date and time of personalisation): reading (command Get Data)

The SFP AC_SFP controls the access of subjects to **security data** by an implicit connection with the respective command. There are no further restrictions for the execution of the above mentioned access modes concerning secret data.

The SFP AC_SFP controls the access of subjects to **identification data of the TOE itself resp. of the TOE´s personalisation** over the corresponding command (Get Data). There are no further restrictions for the access to these data areas.

The SFP AC_SFP controls the access of subjects to **user data** on the basis of security attributes.

For user data, the TOE maintains the following **type of security attributes**:

- Access Rule (AR) consisting of one or more Access Modes (AM) and a single Access Condition (AC)

The AM indicates the command type for accessing the object. The AC defines the conditions under which a command executed by a subject is allowed to access the object.

The access modes to objects of type user data have been defined above. Further, the TOE maintains the following **types of elementary ACs**:

- **ALW (Always)**
  The command can always be executed without any restriction.
- **NEV (Never)**
  The command can never be executed.
- **AUT (Key based user authentication)**
  The right corresponding to a successful external key based authentication must be opened up (done by the command External Authenticate) before the command can be executed.
- **PWD (Password based user authentication)**
  The right corresponding to a successful password based authentication must be opened up (done by the command Verify) before the command can be executed.
- **PRO SM (Secure Messaging providing data integrity and authenticity)**
  The command must be secured with a cryptographic checksum using secure messaging as defined in the Tachograph Card Specification (/TachAn1B/, Appendix 11, chap. 5, Appendix 2, chap. 3.6.2.2, 3.6.3.2).
- **ENC SM (Secure Messaging providing data confidentiality)**
  The command must be secured with an encryption using secure messaging as defined in the Tachograph Card Specification (/TachAn1B/, Appendix 11, chap. 5, Appendix 2, chap. 3.6.2.2, 3.6.3.2).

The above mentioned elementary AC elements can be combined according to the rules defined in ISO/IEC 7816-9 (ACs in expanded format). The object's AC assigned to an AM comprises then a logical expression of elementary AC elements using logical AND.

For each type of Tachograph Card the access rules for the different objects and access modes are implemented according to the requirements in the Tachograph Card Specification /TachAn1B/, Appendix 2, chap. 4.

The AC element PWD is only relevant for the Tachograph Card type Worshop Card. For a Workshop Card the actual security status reached by the AC element PWD will be evaluated. A mutual authentication process between the card and the external world is only possible if a successful preceding verification process with the PIN of the card has been taken place.

Generally, an object of type user data can only be accessed if an access mode exists and an access rule has been attached to the object (during its creation).

Additionally, for rule decisions the AC_SFP uses the actual security status set in the card as reference value.

The AC_SFP explicitly authorises access of subjects to objects based on the following rules:

- The TSF allows access to an object for a defined access mode, if the object's access condition is valid for this access mode.
- The TSF evaluates within an AC the logical expression of elementary AC elements (boolean expression) according to the following rules:
  - AC element ALW is set to "true"
  - AC element NEV is set to "false"
  - AC element AUT is set to "true", if AUT complies with the actual security status (preceding external authentication has been conducted successfully)
  - AC element PWD is set to "true", if PWD complies with the actual security status (preceding PIN verification has been conducted successfully)
  - AC element SM PRO is set to "true", if SM PRO complies with the user indication for SM PRO and SM PRO complies with the actual security status (preceding internal-external authentication has been conducted successfully).
  - AC element SM ENC is set to "true", if SM ENC complies with the user indication for SM ENC and SM ENC complies with the actual security status (preceding internal-external authentication has been conducted successfully).
- For the command Read Binary, the following special rules hold:
  - The TSF allows read access to an object as well in that case, that there does not exist an SM PRO element in the object's AC, but SM PRO is indicated by the user. (The command will then be secured accordingly with Secure Messaging.)

5.1.1.2.2 Security Functional Requirements

FAU **Security Audit**	
FAU_SAA **Security Audit Analysis**	
FAU_SAA.1 **Potential Violation Analysis**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.5
FAU_SAA.1.1 The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the TSP. **FAU_SAA.1.2** The TSF shall enforce the following rules for monitoring audited events: a) Accumulation or combination of [assignment: *subset of defined auditable events*] known to indicate a potential security violation; b) [assignment: *any other rules*]. Hierarchical to: No other components Dependencies: - FAU_GEN.1 Audit data generation Management: a) maintenance of the rules by (adding, modifying, deletion) of rules from the set of rules Audit: a) Minimal: Enabling and disabling of any of the analysis mechanisms b) Minimal: Automated responses performed by the tool	**FAU_SAA.1-1:** **FAU_SAA.1.1-1** The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the TSP. **FAU_SAA.1.2-1** The TSF shall enforce the following rules for monitoring audited events: a) Accumulation or combination of [- **cardholder authentication failure (5 consecutive unsuccessful PIN checks),** - **self test error,** - **stored data integrity error,** - **activity data input integrity error** - **error in the framework of securing of data exchange (concerning data integrity and / or data confidentiality)** - **software / hardware failure**] known to indicate a potential security violation; b) [**none**]. Hierarchical to: No other components Dependencies: Not applicable Management: Not applicable

FCO **Communication**	
FCO_NRO **Non-Repudiation of Origin**	

FCO_NRO.1 **Selective Proof of Origin**	TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.8.2
FCO_NRO.1.1 The TSF shall be able to generate evidence of origin for transmitted [assignment: *list of information types*] at the request of the [selection: *originator, recipient,* [assignment: *list of third parties*]]. **FCO_NRO.1.2** The TSF shall be able to relate the [assignment: *list of attributes*] of the originator of the information, and the [assignment: *list of information fields*] of the information to which the evidence applies. **FCO_NRO.1.3** The TSF shall provide a capability to verify the evidence of origin of information to [selection: *originator, recipient,* [assignment: *list of third parties*]] given [assignment: *limitations on the evidence of origin*]. Hierarchical to: No other components Dependencies: - FIA_UID.1 Timing of identification Management: a) The management of changes to information types, fields, originator attributes and recipients of evidence. Audit: a) Minimal: The identity of the user who requested that evidence of origin would be generated. b) Minimal: The invocation of the non-repudiation service. c) Basic: Identification of the information, the destination, and a copy of the evidence provided. d) Detailed: The identity of the user who requested a verification of the evidence.	**FCO_NRO.1-1:** **FCO_NRO.1.1-1** The TSF shall be able to generate evidence of origin for transmitted [**user data (download function)**] at the request of the [**recipient**]. **Refinement** DEX_304: The TOE shall be able to generate an evidence of origin for data downloaded to external media. **FCO_NRO.1.2-1** The TSF shall be able to relate the [**card identity given by the card´s specific private signature key**] of the originator of the information, and the [**hash value of the data area of the currently selected transparent elementary file**] of the information to which the evidence applies. **Refinement** DEX_306: The TOE shall be able to download data to external storage media with associated security attributes such that downloaded data integrity can be verified. **FCO_NRO.1.3-1** The TSF shall provide a capability to verify the evidence of origin of information to [**the recipient**] given [**no limitation**]. **Refinement** DEX_305: The TOE shall be able to provide a capability to verify the evidence of origin of downloaded data to the recipient. Hierarchical to: No other components Dependencies: - FIA_UID.1-1 Timing of identification Management: Not applicable

FCS **Cryptographic Support**	
FCS_CKM **Cryptographic Key Management**	

FCS_CKM.1 **Cryptographic Key Generation**	TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.9
FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *cryptographic key generation algorithm*] and specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: a) the management of changes to cryptographic key attributes. Examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption) Audit: a) Minimal: Success and failure of the activity b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	**FCS_CKM.1-1:** **FCS_CKM.1.1-1** The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [**generation of a 3-DES session key**] and specified cryptographic key sizes [**of double length (128 bits with 112 bits entropy, no parity bits set)**] that meet the following: [- **ISO/IEC 9798-3 Information Technology – Security Techniques – Entity Authentication Mechanisms – Part 2: Entity Authentication Using a Public Key Algorithm, Second Edition 1998** - **Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 3.1.3 (CSM_012), 3.2 (CSM_015), 4 (CSM_020)**]. **Refinement** CSP_301: If the TSF generates cryptographic keys, it shall be in accordance with specified cryptographic key generation algorithms and specified cryptographic key sizes. (...) Hierarchical to: No other components Dependencies: - [FCS_CKM.2-1 Cryptographic key distribution] - FCS_CKM.4-1 Cryptographic key destruction Management: Not applicable
FCS_CKM.2 **Cryptographic Key Distribution**	TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.9
FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [assignment: *cryptographic key distribution method*] that meets the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction	**FCS_CKM.2-1:** **FCS_CKM.2.1-1** The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [**3-DES session key agreement (with send sequence counter) by an internal-external authentication mechanism**] that meets the following: [- **ISO/IEC 9798-3 Information Technology – Security Techniques – Entity Authentication Mechanisms – Part 2: Entity Authentication Using a Public Key Algorithm, Second Edition 1998** - **Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 3.1.3 (CSM_012), 4**

- FMT_MSA.2 Secure security attributes Management: a) the management of changes to cryptographic key attributes. Examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption) Audit: a) Minimal: Success and failure of the activity b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	**(CSM_020), Appendix 2, chap. 3.6.8, 3.6.9**]. **Refinement** CSP_302: If the TSF distributes cryptographic keys, it shall be in accordance with specified cryptographic key distribution methods. Hierarchical to: No other components Dependencies: - [FCS_CKM.1-1 Cryptographic key generation] - FCS_CKM.4-1 Cryptographic key destruction Management: Not applicable
	FCS_CKM.2-2: **FCS_CKM.2.1-2** The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [**import of public RSA-keys by certificates (non self-descriptive card verifiable certificates in conformance with ISO/IEC 7816-8)**] that meets the following: [- **Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 3.3, esp. 3.3.1 (CSM_017), 3.3.2 (CSM_018) and 3.3.3 (CSM_019), Appendix 2, chap. 3.6.7 (esp. TCS_346)**]. **Refinement** CSP_302: If the TSF distributes cryptographic keys, it shall be in accordance with specified cryptographic key distribution methods. Hierarchical to: No other components Dependencies: - [FDP_ITC.1-1 Import of user data without security attributes] - FCS_CKM.4-2 Cryptographic key destruction Management: Not applicable
FCS_CKM.3 **Cryptographic Key Access**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.9
FCS_CKM.3.1 The TSF shall perform [assignment: *type of cryptographic key access*] in accordance with a specified cryptographic key access method [assignment: *cryptographic key access method*] that meets the	**FCS_CKM.3-1:** **FCS_CKM.3.1-1** The TSF shall perform [**the access to a private RSA-key for the generation of a digital signature**] in

following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: a) the management of changes to cryptographic key attributes. Examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption) Audit: a) Minimal: Success and failure of the activity b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	accordance with a specified cryptographic key access method [**access to the key by its implicitly known reference within the execution of the command PSO Compute Digital Signature resp. the command Internal Authenticate**] that meets the following: [- **Tachograph Card specification,** /TachAn1B/ **Appendix 2, chap. 3.6.13 (TCS_373), 3.6.8 (TCS_350)**]. Hierarchical to: No other components Dependencies: Not applicable Management: Not applicable
	FCS_CKM.3-2: **FCS_CKM.3.1-2** The TSF shall perform [**the access to a public RSA-key for the verification of a digital signature**] in accordance with a specified cryptographic key access method [**access to the key by its reference explicitly set before within the execution of the command PSO Verify Digital Signature resp. the command External Authenticate resp. the command PSO Verify Certificate**] that meets the following: [- **Tachograph Card specification** /TachAn1B/**, Appendix 2, chap. 3.6.14 (TCS_377), 3.6.9 (TCS_355), 3.6.7 (TCS_347)**]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1-1 Import of user data without security attributes] - FCS_CKM.4-2 Cryptographic key destruction Management: Not applicable
	FCS_CKM.3-3: **FCS_CKM.3.1-3** The TSF shall perform [**the access to a private RSA-key for the decryption operation**] in accordance

	with a specified cryptographic key access method [**access to the key by its implicitly known reference within the execution of the command External Authenticate**] that meets the following: [- **Tachograph Card specification** /TachAn1B/**, Appendix 2, 3.6.9 (TCS_355)**]. Hierarchical to: No other components Dependencies: - Not applicable Management: Not applicable
	FCS_CKM.3-4: **FCS_CKM.3.1-4** The TSF shall perform [**the access to a public RSA-key for the encryption operation**] in accordance with a specified cryptographic key access method [**access to the key by its reference explicitly set before within the execution of the command Internal Authenticate**] that meets the following: [- **Tachograph Card specification** /TachAn1B/**, Appendix 2, chap. 3.6.8 (TCS_350)**]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1-1 Import of user data without security attributes] - FCS_CKM.4-2 Cryptographic key destruction Management: Not applicable
	FCS_CKM.3-5: **FCS_CKM.3.1-5** The TSF shall perform [**the encryption, decryption, MAC generation and MAC verification operations with a 3-DES session key for secure messaging**] in accordance with a specified cryptographic key access method [**access to the session key by its reference implicit set by the card before within the execution of the command Read Binary resp. the command Update Binary, if Secure Messaging is required**] that meets the following: [- **Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 3.1.3 (CSM_013), Appendix 2, chap. 3.6.2.2, 3.6.3.2**

	]. **Refinement** CSP_301: (...) Generated cryptographic session keys shall have a limited (TBD by manufacturer and not more than 240) number of possible use. Hierarchical to: No other components Dependencies: - [FCS_CKM.1-1 Cryptographic key generation] - FCS_CKM.4-1 Cryptographic key destruction Management: Not applicable
FCS_CKM.4 **Cryptographic Key Destruction**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.9
FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: *cryptographic key destruction method*] that meets the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FMT_MSA.2 Secure security attributes Management: a) the management of changes to cryptographic key attributes. Examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption) Audit: a) Minimal: Success and failure of the activity b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	**FCS_CKM.4-1:** **FCS_CKM.4.1-1** The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [**erasing of 3-DES session keys**] that meets the following: [- **Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 3.1.3 (CSM_013), Appendix 2, chap. 3.6.8 (TCS_353)**]. Hierarchical to: No other components Dependencies: - [FCS_CKM.1-1 Cryptographic key generation] Management: Not applicable
	FCS_CKM.4-2: **FCS_CKM.4.1-2** The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [**erasing of imported public RSA-keys and references to public RSA-keys**] that meets the following:

	[- **Tachograph Card specification** /TachAn1B/**, Appendix 2, chap. 3.6.10 (TCS_363)**]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1-1 Import of user data without security attributes] Management: Not applicable
FCS_COP **Cryptographic Operation**	
FCS_COP.1 **Cryptographic Operation**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.9
FCS_COP.1.1 The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] - FCS_CKM.4 Cryptographic key destruction - FMT_MSA.2 Secure security attributes Management: --- Audit: a) Minimal: Success and failure, and the type of cryptographic operation b) Basic: Any applicable cryptographic mode(s) of operation, subject attributes and object attributes	**FCS_COP.1-1:** **FCS_COP.1.1-1** The TSF shall perform [**the explicit signature generation and verification (commands PSO Compute Digital Signature and PSO Verify Digital Signature)**] in accordance with a specified cryptographic algorithm [**RSA**] and cryptographic key sizes [**of 1024 bits**] that meet the following: [- **PKCS#1 (with SHA-1) signature generation / verification scheme, RSA Encryption Standard Version 2.0, October 1998** - **SHA-1, FIPS Pub. 180-1, NIST, April 1995** - **Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 2.2.1 (CSM_003), 2.2.2 (CSM_004), 6.1 (CSM_034) and 6.2 (CSM_035)**]. Hierarchical to: No other components Dependencies: Not applicable Management: ---
	FCS_COP.1-2: **FCS_COP.1.1-2** The TSF shall perform [**the implicit signature generation and verification (commands Internal Authenticate, External Authenticate and PSO Verify Certificate)**] in accordance with a specified cryptographic algorithm [**RSA**] and cryptographic key sizes [**of 1024 bits**] that meet the following:

	[- **ISO/IEC 9796-2 Information Technology – Security Techniques – Digital Signature Schemes Giving Message Recovery – Part 2: Mechanisms Using a Hash Function, First Edition 1997** - **SHA-1, FIPS Pub. 180-1, NIST, April 1995** - **Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 2.2.1 (CSM_003), 2.2.2 (CSM_004), 4 (CSM_020), 3.3.2, 3.3.3**]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1-1 Import of user data without security attributes] - FCS_CKM.4-2 Cryptographic key destruction Management: ---
	FCS_COP.1-3: **FCS_COP.1.1-3** The TSF shall perform [**the implicit encryption and decryption operations concerning asymmetric cryptography (commands Internal Authenticate and External Authenticate)**] in accordance with a specified cryptographic algorithm [**RSA**] and cryptographic key sizes [**of 1024 bits**] that meet the following: [- **PKCS#1 (with SHA-1) encryption / decryption primitive, RSA Encryption Standard Version 2.0, October 1998** - **Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 2.2.1 (CSM_003), 4**]. Hierarchical to: No other components Dependencies: - [FDP_ITC.1-1 Import of user data without security attributes] - FCS_CKM.4-2 Cryptographic key destruction Management: ---
	FCS_COP.1-4: **FCS_COP.1.1-4** The TSF shall perform [**the encryption and decryption operations concerning symmetric cryptography**] in accordance with a specified cryptographic algorithm [**3-DES in CBC mode with ICV = 0**] and

	cryptographic key sizes [**of 128 bits (112 bits entropy, no parity bits set)**] that meet the following: [- **Data Encryption Standard, FIPS Pub. 46-3, NIST, Draft 1999** - **ANSI X9.52 Triple Data Encryption Algorithm Modes of Operations 1998** - **Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 2.2.3 (CSM_005) and 5.4 (CSM_031)**]. Hierarchical to: No other components Dependencies: - [FCS_CKM.1-1 Cryptographic key generation] - FCS_CKM.4-1 Cryptographic key destruction Management: ---
	FCS_COP.1-5: **FCS_COP.1.1-5** The TSF shall perform [**the MAC generation and the MAC verification concerning symmetric cryptography**] in accordance with a specified cryptographic algorithm [**DES Retail-MAC (with consideration of the send sequence counter)**] and cryptographic key sizes [**of 128 bits (112 bits entropy, no parity bits set)**] that meet the following: [- **ANSI X9.19 Financial Institution Retail Message Authentication 1986** - **Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 2.2.3 (CSM_005) and 5.3 (CSM_028))**]. Hierarchical to: No other components Dependencies: - [FCS_CKM.1-1 Cryptographic key generation] - FCS_CKM.4-1 Cryptographic key destruction Management: ---

FDP **User Data Protection**	
FDP_ACC	

Access Control Policy	
FDP_ACC.2 **Complete Access Control**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.3.1, 4.4

FDP_ACC.2.1
The TSF shall enforce the [assignment: *access control SFP*] on [assignment: *list of subjects and objects*] and all operations among subjects and objects covered by the SFP.
 	
FDP_ACC.2.2
The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.
 	
Hierarchical to:
FDP_ACC.1
 	
Dependencies:
- FDP_ACF.1 Security attribute based access control
 	
Management:
--- 	
 	
Audit:
--- 	
	FDP_ACC.2-1:
	
	FDP_ACC.2.1-1
	The TSF shall enforce the [**AC_SFP**] on
	[
	subjects:
	- **vehicle units (in the sense of the Tachograph Card specification)**
	- **other card interface devices (non-vehicle units)**
	
	objects:
	- **user data:**
	  - **identification data (card identification data, cardholder identification data)**
	  - **activity data (cardholder activities data, events and faults data, control activity data)**
	- **security data:**
	  - **card´s private signature key**
	  - **public keys**
	  - **session keys**
	  - **PIN (only workshop card)**
	  - **card´s private authentication key**
	- **TOE software code**
	- **TOE file system (incl. file structure, add. internal structures, access conditions)**
	- **identification data of the TOE (-IC, -ES)**
	- **identification data of the TOE´s personalisation**
	]
	and all operations among subjects and objects covered by the SFP.
	
	FDP_ACC.2.2-1
	The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.
	
	Hierarchical to:
	FDP_ACC.1
	
	Dependencies:
	- FDP_ACF.1-1 Security attribute based access control
	
	Management:
	--- 	

	FDP_ACC.2-2: **FDP_ACC.2.1-2** The TSF shall enforce the [**PERS-AC_SFP**] on

	[
	subjects:
	- **personalisation units**
	- **other card interface devices (non-personalisation units)**
	
	objects:
	- **data fields for user data as:**
	  - **identification data (card identification data, cardholder identification data)**
	  - **activity data (cardholder activities data, events and faults data, control activity data)**
	- **data fields for security data as:**
	  - **card´s private signature key**
	  - **public keys**
	  - **PIN (only workshop card)**
	- **security data:**
	  - **card´s private personalisation key**
	  - **personalisation unit´s public personalisation key**
	  - **session keys**
	  - **card´s private authentication key**
	- **TOE software code**
	- **TOE file system (incl. file structure, add. internal structures, access conditions)**
	- **identification data of the TOE (-IC, -ES)**
	- **data field for identification data of the TOE´s personalisation**
	]
	and all operations among subjects and objects covered by the SFP.
	
	FDP_ACC.2.2-2
	The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.
	
	Hierarchical to:
	FDP_ACC.1
	
	Dependencies:
	- FDP_ACF.1-2 Security attribute based access control
	
	Management:
	--- 	

FDP_ACF **Access Control Functions**	
FDP_ACF.1 **Security Attribute Based Access Control**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.3.2, 4.4 / JILDig-Tacho, chap. 2.6
FDP_ACF.1.1 The TSF shall enforce the [assignment: *access control SFP*] to objects based on [assignment: *security*	**FDP_ACF.1-1:** **FDP_ACF.1.1-1**

attributes, *named groups of security attributes*].
 	
FDP_ACF.1.2
The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*].
 	
FDP_ACF.1.3
The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*].
 	
FDP_ACF.1.4
The TSF shall explicitly deny access of subjects to objects based on the [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*].
 	
Hierarchical to:
No other components
 	
Dependencies:
- FDP_ACC.1 Subset access control
- FMT_MSA.3 Static attribute initialisation
 	
Management:
a) Managing the attributes used to make explicit access or denial based decisions
 	
Audit:
a) Minimal: Successful requests to perform an operation on an object covered by the SFP
b) Basic: All requests to perform an operation on an object covered by the SFP
c) Detailed: The specific security attributes used in making an access check
	The TSF shall enforce the [**AC_SFP**] to objects based on [**USER_GROUP**].
	
	FDP_ACF.1.2-1
	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:
	[
	- **GENERAL_READ:**
	  - **driver card, workshop card: user data may be read from the TOE by any user**
	  - **control card, company card: user data may be read from the TOE by any user, except cardholder identification data which may be read by VEHICLE_UNIT only;**
	- **IDENTIF_WRITE:**
	all card types: identification data may only be written once and before the end of phase 6 of card's life-cycle; no user may write or modify identification data during end-usage phase of card's life-cycle;
	- **ACTIVITY_WRITE:**
	all card types: activity data may be written to the TOE by VEHICLE_UNIT only;
	- **SOFT_UPGRADE:**
	all card types: no user may upgrade TOE's software;
	- **FILE_STRUCTURE:**
	all card types: files structure and access conditions shall be created before end of phase 5 of TOE's life-cycle and then locked from any future modification or deletion by any user
	- **IDENTIF_TOE_READ:**
	all card types: identification data of the TOE and identification data of the TOE´s personalisation may be read from the TOE by any user;
	- **IDENTIF_TOE_WRITE:**
	all card types: identification data of the TOE may only be written once and before the end of phase 5 of card's life-cycle; no user may write or modify these identification data during phase 6 or end-usage phase of card's life-cycle;
	- **IDENTIF_ TOE_ PERS_WRITE:**
	all card types: identification data of the TOE´s personalisation may only be written once and within phase 6 of card's life-cycle; no user may write or modify these identification data during end-usage phase of card's life-cycle
	- **SECDATA_ACCESS:**
	access to session keys or other secret data stored in the framework of the initialisation or personalisation of the TOE is done by an implicit connection with the respective command; hereby, the access to the card´s private signature key for an external authentication, to session keys or to the PIN is only successful for VEHICLE_UNIT, for all other secrets any user will succeed

	]. **Note** /TachAn1B/, Appendix 10, Tachograph Card Generic Security Target, chap. 4.2 and 4.3.2 (FDP_ACF.1.2 GENERAL_READ) say that only control cards may have an authentication process before exporting cardholder identification data, but /TachAn1B/, Appendix 2 TCS_415 says that authentication is mandatory for exporting cardholder identification data. Furthermore, there are no TSF mediated actions defined in FIA_UAU.1.1 for the company card. Agreed interpretation in /JILDigTacho/, chap. 2.6: The allowed actions for a company card seems to be missing in the specification of FIA_UAU.1 in the generic security target of /TachAn1B/, Appendix 10. From the context it is clear that a company card should allow the actions as specified by /TachAn1B/, Appendix 2 (which are the same for a control card). Therefore, the specification of the TOE SFRs in /TachAn1B/, Appendix 10, Tachograph Card Generic Security Target should be read as follows: - GENERAL_READ: User data may be read from the TOE by any user, except cardholder identification data which may be read from control cards **or company cards** by VEHICLE_UNIT only. **Refinements** ACT_301: The TOE shall hold permanent identification data. ACT_302: There shall be an indication of the time and date of the TOE's personalisation. This indication shall remain unalterable. **FDP_ACF.1.3-1** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**none**]. **FDP_ACF.1.4-1** The TSF shall explicitly deny access of subjects to objects based on the [**none**]. Hierarchical to: No other components Dependencies: - FDP_ACC.2-1 Subset access control Management: Not applicable
	FDP_ACF.1-2: **FDP_ACF.1.1-2** The TSF shall enforce the [**PERS-AC _SFP**] to ob-

	jects based on [**USER_GROUP**]. **FDP_ACF.1.2-2** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [- **IDENTIF_WRITE:** **all card types: identification data may only be written before the end of phase 6 of card's life-cycle; within phase 6, identification data may be written by PERSO_UNIT only** - **ACTIVITY_WRITE:** **all card types: activity data may only be written during the end-usage phase of card's life-cycle** - **SECDATA_WRITE:** **all card types: security data (except session keys) may only be written resp. loaded before the end of phase 6 of card's life-cycle; within phase 6, security data may be loaded by PERSO_UNIT only** - **SECDATA_ACCESS:** **access to session keys or other secret data stored in the framework of the initialisation of the TOE is done by an implicit connection with the respective command; hereby, the access to the card´s private personalisation key for an external authentication or to session keys is only successful for PERSO_UNIT, for all other secrets any user will succeed** - **SOFT_UPGRADE:** **all card types: no user may upgrade TOE's software;** - **FILE_STRUCTURE:** **all card types: files structure and access conditions shall be created before end of phase 5 of TOE's life-cycle and then locked from any future modification or deletion by any user** - **IDENTIF_TOE_READ:** **all card types: identification data of the TOE or of the TOE´s personalisation may be read from the TOE by any user;** - **IDENTIF_TOE_WRITE:** **all card types: identification data of the TOE may only be written once and before the end of phase 5 of card's life-cycle; no user may write or modify these identification data during phase 6 phase of card's life-cycle or later;** - **IDENTIF_ TOE_ PERS_WRITE:** **all card types: identification data of the TOE´s personalisation may only be written within phase 6 of card's life-cycle; no user may write or modify these identification data during end-usage phase of card's life-cycle**]. **Refinements** ACT_301: The TOE shall hold permanent identifica-

	tion data. ACT_302: There shall be an indication of the time and date of the TOE's personalisation. This indication shall remain unalterable. **FDP_ACF.1.3-2** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**none**]. **FDP_ACF.1.4-2** The TSF shall explicitly deny access of subjects to objects based on the [**none**]. Hierarchical to: No other components Dependencies: - FDP_ACC.2-2 Subset access control Management: Not applicable
FDP_DAU **Data Authentication**	
FDP_DAU.1 **Basic Data Authentication**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.6.2
FDP_DAU.1.1 The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of [assignment: *list of objects or information types*]. **FDP_DAU.1.2** The TSF shall provide [assignment: *list of subjects*] with the ability to verify evidence of the validity of the indicated information. Hierarchical to: No other components Dependencies: No dependencies Management: a) The assignment or modification of the objects for which data authentication may apply could be configurable in the system Audit: a) Minimal: Successful generation of validity evidence b) Basic: Unsuccessful generation of validity evidence c) Detailed: The identity of the subject that requested the evidence	**FDP_DAU.1-1:** **FDP_DAU.1.1-1** The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of [**activity data**]. **FDP_DAU.1.2-1** The TSF shall provide [**any subject (i.e. vehicle units and other card interface devices (non-vehicle units))**] with the ability to verify evidence of the validity of the indicated information. Hierarchical to: No other components Dependencies: No dependencies Management: Not applicable

FDP_ETC **Export to Outside TSF Control**	
FDP_ETC.1 **Export of User Data without Security Attributes**	PP9911
FDP_ETC.1.1 The TSF shall enforce the [assignment: *access control SFP(s) and/or information flow control SFP(s)*] when exporting user data, controlled under the SFP(s), outside of the TSC. **FDP_ETC.1.2** The TSF shall export the user data without the user data's associated security attributes. Hierarchical to: No other components Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] Management: --- Audit: a) Minimal: Successful export of information b) Basic: All attempts to export information	**FDP_ETC.1-1:** **FDP_ETC.1.1-1** The TSF shall enforce the [**for phase 6 of the product´s life-cycle: PERS-AC_SFP; for phase 7 of the product´s life-cycle: AC_SFP**] when exporting user data, controlled under the SFP(s), outside of the TSC. **FDP_ETC.1.2-1** The TSF shall export the user data, without the user data's associated security attributes. Hierarchical to: No other components Dependencies: - [FDP_ACC.2-1 Subset access control] - [FDP_ACC.2-2 Subset access control] Management: ---
FDP_ETC.2 **Export of User Data with Security Attributes**	TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.8.2
FDP_ETC.2.1 The TSF shall enforce the [assignment: *access control SFP(s) and/or information flow control SFP(s)*] when exporting user data, controlled under the SFP(s), outside of the TSC. **FDP_ETC.2.2** The TSF shall export the user data with the user data's associated security attributes. **FDP_ETC.2.3** The TSF shall ensure that the security attributes, when exported outside the TSC, are unambiguously associated with the exported user data. **FDP_ETC.2.4** The TSF shall enforce the following rules when user data is exported from the TSC: [assignment: *additional exportation control rules*]. Hierarchical to: No other components	**FDP_ETC.2-1:** **FDP_ETC.2.1-1** The TSF shall enforce the [**AC_SFP**] when exporting user data **within the card data download function**, controlled under the SFP(s), outside of the TSC. **FDP_ETC.2.2-1** The TSF shall export the user data with the user data's associated security attributes. **Refinement** DEX_306: The TOE shall be able to download data to external storage media with associated security attributes such that downloaded data integrity can be verified. **FDP_ETC.2.3-1** The TSF shall ensure that the security attributes, when exported outside the TSC, are unambiguously associated with the exported user data. **FDP_ETC.2.4-1**

Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] Management: a) The additional exportation control rules could be configurable by a user in a defined role. Audit: a) Minimal: Successful export of information b) Basic: All attempts to export information	The TSF shall enforce the following rules when user data is exported from the TSC: [**none**] Hierarchical to: No other components Dependencies: - [FDP_ACC.2-1 Subset access control] Management: Not applicable
FDP_ITC **Import from Outside TSF Control**	
FDP_ITC.1 **Import of User Data without Security Attributes**	PP9911
FDP_ITC.1.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] when importing user data, controlled under the SFP, from outside of the TSC. **FDP_ITC.1.2** The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC. **FDP_ITC.1.3** The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: [assignment: *additional importation control rules*]. Hierarchical to: No other components Dependencies: - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] - FMT_MSA.3 Static attribute initialisation Management: a) The modification of the additional control rules used for import Audit: a) Minimal: Successful import of user data, including any security attributes b) Basic: All attempts to import user data, including any security attributes c) Detailed: The specification of security attributes for imported user data supplied by an authorised user	**FDP_ITC.1-1:** **FDP_ITC.1.1-1** The TSF shall enforce the [**for phase 6 of the product´s life-cycle: PERS-AC_SFP; for phase 7 of the product´s life-cycle: AC_SFP**] when importing user data, controlled under the SFP, from outside of the TSC. **FDP_ITC.1.2-1** The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC. **FDP_ITC.1.3-1** The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: [**none**]. Hierarchical to: No other components Dependencies: - [FDP_ACC.2-1 Subset access control] - [FDP_ACC.2-2 Subset access control] Management: Not applicable

FDP_RIP **Residual Information Protection**	
FDP_RIP.1 **Subset Residual Information Protection**	PP9911
FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection: *allocation of the resource to*, *deallocation of the resource from*] the following objects: [assignment: *list of objects*]. Hierarchical to: No other components Dependencies: No dependencies Management: a) The choice of when to perform residual information protection (i.e. upon allocation or deallocation) could be made configurable within the TOE Audit: ---	**FDP_RIP.1-1:** **FDP_RIP.1.1-1** The TSF shall ensure that any previous information content of a resource is made unavailable upon the [**allocation of the resource from**] the following objects: [**security relevant material (e.g. cryptographic KEYs, PINs, ...)**]. Hierarchical to: No other components Dependencies: No dependencies Management: Not applicable
FDP_SDI **Stored Data Integrity**	
FDP_SDI.2 **Stored Data Integrity Monitoring and Action**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.6.1
FDP_SDI.2.1 The TSF shall monitor user data stored within the TSC for [assignment: *integrity errors*] on all objects, based on the following attributes: [assignment: *user data attributes*]. **FDP_SDI.2.2** Upon detection of a data integrity error, the TSF shall [assignment: *action to be taken*]. Hierarchical to: FDP_SDI.1 Dependencies: No dependencies Management: a) The actions to be taken upon the detection of an integrity error could be configurable Audit: a) Minimal: Successful attempts to check the integrity of user data, including an indication of the results of the check b) Basic: All attempts to check the integrity of user	**FDP_SDI.2-1:** **FDP_SDI.2.1-1** The TSF shall monitor user data **(incl. stored secrets)** stored within the TSC for [**integrity error before access and processing**] on all objects, based on the following attributes: [**user data value, user data object**]. **FDP_SDI.2.2-1** Upon detection of a data integrity error, the TSF shall [**warn the entity connected**]. Hierarchical to: FDP_SDI.1 Dependencies: No dependencies Management: Not applicable

data, including an indication of the results of the check, if performed c) Detailed: The type of integrity error that occurred d) Detailed: The action taken upon detection of an integrity error	

FIA **Identification and Authentication**	
FIA_AFL **Authentication Failures**	
FIA_AFL.1 **Authentication Failure Handling**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.2.3
FIA_AFL.1.1 The TSF shall detect when [assignment: *number*] unsuccessful authentication attempts occur related to [assignment: *list of authentication events*]. **FIA_AFL.1.2** When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [assignment: *list of actions*]. Hierarchical to: No other components Dependencies: - FIA_UAU.1 Timing of authentication Management: a) management of the threshold for unsuccessful authentication attempts b) management of actions to be taken in the event of an authentication failure Audit: a) Minimal: the reaching of the threshold for the unsuccesful authentication attempts and the actions (e.g. disabling of a terminal) taken and the subsequent, if appropriate, restoration to the normal state (e.g. re-enabling of a terminal)	For all card types: Card reaction for each single user authentication failure: **FIA_AFL.1-1:** **FIA_AFL.1.1-1** The TSF shall detect when [**1**] unsuccessful authentication attempt occurs related to [**authentication of a card interface device**]. **FIA_AFL.1.2-1** When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [**warn the entity connected, assume the user as NON_VEHICLE_UNIT**]. Hierarchical to: No other components Dependencies: - FIA_UAU.1-1 Timing of authentication Management: Not applicable
	For workshop cards only: Card reaction in the case of a failure of the additional PIN-authentication mechanism: **FIA_AFL.1-2:** **FIA_AFL.1.1-2** The TSF shall detect when [**5**] unsuccessful authentication attempts occur related to [**PIN check (workshop card)**].

	FIA_AFL.1.2-2 When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [**warn the entity connected, block the PIN check procedure such that any subsequent PIN check attempt will fail, be able to indicate to subsequent users the reason of the blocking**]. **Note** Agreed interpretation in /JILDigTacho/, chap. 2.6: To ensure that the Tachograph Card takes care of unsuccessful authentication events, the sentence "The following assignments describe the card reaction in the case of failure of the additional authentication mechanism required in UIA_302." (/TachAn1B/, Appendix 10, Tachograph Card Generic Security Target, chap. 4.2.3) should be read as follows: "**Additionally** the following assignments describe the card reaction in the case of failure of the additional authentication mechanism required in UIA_302." This should ensure that the Tachograph Card (here only the workshop card) only allows a mutual authentication with the Vehicle Unit after a successful PIN verification of a human user. Hierarchical to: No other components Dependencies: - FIA_UAU.1-1 Timing of authentication Management: Not applicable
FIA_ATD **User Attribute Definition**	
FIA_ATD.1 **User Attribute Definition**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.2.1
FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: [assignment: *list of security attributes*]. Hierarchical to: No other components Dependencies: No dependencies Management: a) if so indicated in the assignment, the authorised administrator might be able to define additional security attributes for users Audit:	**FIA_ATD.1-1:** **FIA_ATD.1.1-1** The TSF shall maintain the following list of security attributes belonging to individual users: [**phase 6 of the product´s life-cycle:** - **USER_GROUP (PERSO_UNIT, NON_PERSO_UNIT)** **phase 7 of the product´s life-cycle:** - **USER_GROUP (VEHICLE_UNIT, NON_VEHICLE_UNIT)** - **USER_ID (VRN and Reg. MSC, where USER_ID is only known to USER_GROUP = VEHICLE_UNIT)**

---	] Hierarchical to: No other components Dependencies: No dependencies Management: Not applicable
FIA_UAU **User Authentication**	
FIA_UAU.1 **Timing of Authentication**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.2.2 / JILDigTacho, chap. 2.6
FIA_UAU.1.1 The TSF shall allow [assignment: *list of TSF mediated actions*] on behalf of the user to be performed before the user is authenticated. **FIA_UAU.1.2** The TSF shall require each user to be successfully authenticated before allowing any other TSF- mediated actions on behalf of that user. Hierarchical to: No other components Dependencies: - FIA_UID.1 Timing of identification Management: a) management of the authentication data by an administrator b) management of the authentication data by the associated user c) managing the list of actions that can be taken before the user is authenticated Audit: a) Minimal: Unsuccessful use of the authentication mechanism b) Basic: All use of the authentication mechanism c) Detailed: All TSF mediated actions performed before authentication of the user	**FIA_UAU.1-1:** **FIA_UAU.1.1-1** The TSF shall allow [**driver card, workshop card: export of user data with security attributes (card data download function),** **control card, company card: export of user data without security attributes except export of cardholder identification data**] on behalf of the user to be performed before the user is authenticated. **Note** /TachAn1B/, Appendix 10, Tachograph Card Generic Security Target, chap. 4.2 and 4.3.2 (FDP_ACF.1.2 GENERAL_READ) say that only control cards may have an authentication process before exporting cardholder identification data, but /TachAn1B/, Appendix 2 TCS_415 says that authentication is mandatory for exporting cardholder identification data. Furthermore, there are no TSF mediated actions defined in FIA_UAU.1.1 for the company card. Agreed interpretation in /JILDigTacho/, chap. 2.6: The allowed actions for a company card seems to be missing in the specification of FIA_UAU.1 in the generic security target of /TachAn1B/, Appendix 10. From the context it is clear that a company card should allow the actions as specified by /TachAn1B/, Appendix 2 (which are the same for a control card). Therefore, the specification of the TOE SFRs in /TachAn1B/, Appendix 10, Tachograph Card Generic Security Target should be read as follows: - Control **and company** card**s**: Export of user data without security attributes except cardholder identification data

	FIA_UAU.1.2-1 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user. **Refinements** UIA_301: Authentication of a vehicle unit shall be performed by means of proving that it possesses security data that only the system could distribute. UIA_302: The workshop card shall provide an additional authentication mechanism by checking a PIN code (This mechanism is intended for the vehicle unit to ensure the identity of the cardholder, it is not intended to protect workshop card content). Hierarchical to: No other components Dependencies: - FIA_UID.1-1 Timing of identification Management: Not applicable
FIA_UAU.3 **Unforgeable Authentication**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.2.2
FIA_UAU.3.1 The TSF shall [selection: *detect, prevent*] use of authentication data that has been forged by any user of the TSF. **FIA_UAU.3.2** The TSF shall [selection: *detect, prevent*] use of authentication data that has been copied from any other user of the TSF. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: a) Minimal: Detection of fraudulent authentication data b) Basic: All immediate measures taken and results of checks on the fraudulent data	**FIA_UAU.3-1:** **FIA_UAU.3.1-1** The TSF shall [**prevent**] use of authentication data that has been forged by any user of the TSF. **FIA_UAU.3.2-1** The TSF shall [**prevent**] use of authentication data that has been copied from any other user of the TSF. Hierarchical to: No other components Dependencies: No dependencies Management: ---
FIA_UAU.4 **Single-use Authentication Mechanisms**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.2.2
FIA_UAU.4.1	**FIA_UAU.4-1:**

The TSF shall prevent reuse of authentication data related to [assignment: *identified authentication mechanism(*s)]. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: a) Minimal: Attempts to reuse authentication data	**FIA_UAU.4.1-1** - The TSF shall prevent reuse of authentication data related to [**key based authentication mechanisms**]. Hierarchical to: No other components Dependencies: No dependencies Management: ---
FIA_UID **User Identification**	
FIA_UID.1 **Timing of Identification**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.2.1 / JILDigTacho, chap. 2.6
FIA_UID.1.1 The TSF shall allow [assignment: *list of TSF-mediated actions*] on behalf of the user to be performed before the user is identified. **FIA_UID.1.2** The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user. Hierarchical to: No other components Dependencies: No dependencies Management: a) the management of the user identities b) if an authorised administrator can change the actions allowed before identification, the managing of the action lists Audit: a) Minimal: Unsuccessful use of the user identification mechanism, including the user identity provided b) Basic: All use of the user identification mechanism, including the user identity provided	**FIA_UID.1-1:** **FIA_UID.1.1-1** The TSF shall allow [**none of the TSF-mediated actions**] on behalf of the user to be performed before the user is identified. **Note** In /TachAn1B/, Appendix 10, Tachograph Card Generic Security Target, FIA_UID.1.1(TSF mediated actions) states that the card shall allow no operations before the identification of the user, and, FDP_ACF.1.2 (GENERAL_READ) states “User data may be read from the TOE by any user, ...”. However, /TachAn1B/, Appendix 11 defines a process to identify and authenticate a VEHICLE_UNIT, but no process is defined to identify other users. Agreed interpretation in /JILDigTacho/, chap. 2.6: In /TachAn1B/, Appendix 10, Tachograph Card Generic Security Target the following types of users are identified:VEHICLE_UNIT and NON_VEHICLE_UNIT. The user NON_VEHICLE_UNIT is identified by the Tachograph Card by just putting it into a card reading device (which could be a Vehicle Unit). After a successful mutual authentication between Tachograph Card and Vehicle Unit, the Tachograph Card assumes the user VEHICLE_UNIT to be identified. **FIA_UID.1.2-1** The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user. Hierarchical to:

	No other components Dependencies: No dependencies Management: Not applicable
FIA_USB **User-Subject Binding**	
FIA_USB.1 **User-Subject Binding**	PP9911
FIA_USB.1.1 The TSF shall associate the appropriate user security attributes with subjects acting on behalf of that user. Hierarchical to: No other components Dependencies: - FIA_ATD.1 User attribute definition Management: a) an authorised administrator can define default subject security attributes Audit: a) Minimal: Unsuccessful binding of user security attributes to a subject (e.g. creation of a subject) b) Basic: Success and failure of binding of user security attributes to a subject (e.g. success and failure to create a subject)	**FIA_USB.1-1:** **FIA_USB.1.1-1** The TSF shall associate the appropriate user security attributes with subjects acting on behalf of that user. Hierarchical to: No other components Dependencies: - FIA_ATD.1-1 User attribute definition Management: Not applicable

FMT **Security Management**	
FMT_MOF **Management of Functions in TSF**	
FMT_MOF.1 **Management of Security Functions Behaviour**	PP9911
FMT_MOF.1.1 The TSF shall restrict the ability to [selection: *determine the behaviour of, disable, enable, modify the behaviour of*] the functions [assignment: *list of functions*] to [assignment: *the authorised identified roles*]. Hierarchical to: No other components Dependencies:	Not applicable

- FMT_SMR.1 Security roles Management: a) managing the group of roles that can interact with the functions in the TSF Audit: a) Basic: All modifications in the behaviour of the functions in the TSF	
FMT_MSA **Management of Security Attributes**	
FMT_MSA.1 **Management of Security Attributes**	PP9911
FMT_MSA.1.1 The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] to restrict the ability to [selection: *change_default, query, modify, delete,* [assignment: *other operations*]] the security attributes [assignment: *list of security attributes*] to [assignment: *the authorised identified roles*]. 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 Management: a) managing the group of roles that can interact with the security attributes Audit: a) Basic: All modifications of the values of security attributes	Not applicable
FMT_MSA.2 **Secure Security Attributes**	PP9911
FMT_MSA.2.1 The TSF shall ensure that only secure values are accepted for security attributes. Hierarchical to: No other components Dependencies: - ADV_SPM.1 Informal TOE security policy model - [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] - FMT_MSA.1 Management of security attributes	Not applicable

- FMT_SMR.1 Security roles Management: --- Audit: a) Minimal: All offered and rejected values for a security attribute b) Detailed: All offered and accepted secure values for a security attribute	
FMT_MSA.3 **Static Attribute Initialisation**	PP9911
FMT_MSA.3.1 The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] to provide [selection: *restrictive, permissive, other property*] default values for security attributes that are used to enforce the *SFP*. **FMT_MSA.3.2** The TSF shall allow the [assignment: *the authorised identified roles*] to specify alternative initial values to override the default values when an object or information is created. Hierarchical to: No other components Dependencies: - FMT_MSA.1 Management of security attributes - FMT_SMR.1 Security roles Management: a) managing the group of roles that can specify initial values b) managing the permissive or restrictive setting of default values for a given access control SFP Audit: a) Basic: Modifications of the default setting of permissive or restrictive rules b) Basic: All modifications of the initial values of security attributes	Not applicable
FMT_MTD **Management of TSF Data**	
FMT_MTD.1 **Management of TSF Data**	PP9911
FMT_MTD.1.1 The TSF shall restrict the ability to [selection: *change_default, query, modify, delete, clear*, [assignment: *other operations*]] the [assignment: *list of TSF data*] to [assignment: *the authorised identified*	Not applicable

roles]. Hierarchical to: No other components Dependencies: - FMT_SMR.1 Security roles Management: a) managing the group of roles that can interact with the TSF data Audit: a) Basic: All modifications to the values of TSF data	
FMT_SMR **Security Management Roles**	
FMT_SMR.1 **Security Roles**	PP9911
FMT_SMR.1.1 The TSF shall maintain the roles [assignment: *the authorised identified roles*]. **FMT_SMR.1.2** The TSF shall be able to associate users with roles. Hierarchical to: No other components Dependencies: - FIA_UID.1 Timing of identification Management: a) managing the group of users that are part of a role Audit: a) Minimal: modifications to the group of users that are part of a role b) Detailed: every use of the rights of a role	Not applicable

FPR **Privacy**	
FPR_UNO **Unobservability**	
FPR_UNO.1 **Unobservability**	PP9911
FPR_UNO.1.1 The TSF shall ensure that [assignment: *list of users*	**FPR_UNO.1-1:**

and/or subjects] are unable to observe the operation [assignment: *list of operations*] on [assignment: *list of objects*] by [assignment: *list of protected users and/or subjects*]. Hierarchical to: No other components Dependencies: No dependencies Management: a) the management of the behaviour of the unobservability function Audit: a) Minimal: The invocation of the unobservability mechanism	**FPR_UNO.1.1-1** The TSF shall ensure that [**within phase 6 of the product´s life cycle: non-personalisation units,** **within phase 7 of the product´s life-cycle: non-vehicle units**] are unable to observe the operation [**mutual authentication (for the agreement of session keys and send sequence counters)**] on [**authentication tokens**] by [**within phase 6 of the product´s life cycle: a personalisation unit,** **within phase 7 of the product´s life-cycle: a vehicle unit**]. Hierarchical to: No other components Dependencies: No dependencies Management: Not applicable
	FPR_UNO.1-2: **FPR_UNO.1.1-2** The TSF shall ensure that [**within phase 6 of the product´s life cycle: non-personalisation units,** **within phase 7 of the product´s life-cycle: non-vehicle units**], **if required,** are unable to observe the operation [**import function of user data, export function of user data**] on [**user data**] by [**within phase 6 of the product´s life cycle: a personalisation unit,** **within phase 7 of the product´s life-cycle: a vehicle unit**]. Hierarchical to: No other components Dependencies: No dependencies Management: Not applicable

FPT **Protection of the TSF**	
FPT_FLS **Fail Secure**	
FPT_FLS.1 **Failure with Preservation of Secure State**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.7.3, 4.7.4
FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [assignment: *list of types of failures in the TSF*]. Hierarchical to: No other components Dependencies: - ADV_SPM.1 Informal TOE security policy model Management: --- Audit: a) Basic: Failure of the TSF	**FPT_FLS.1-1:** **FPT_FLS.1.1-1** The TSF shall preserve a secure state when the following types of failures occur: [- **reset** - **power supply cut-off** - **power supply variations** - **unexpected abortion of the execution of the TSF due to external or internal events (esp. break of a transaction before completion)** - **system breakdown** - **internal Hardware- or Software failure** - **card life cycle corruption** - **application life cycle corruption**]. **Refinements** RLB_306: The TOE shall preserve a secure state during power supply cut-off or variations. RLB_307: If power is cut (or if power variations occur) from the TOE, or if a transaction is stopped before completion, or on any other reset conditions, the TOE shall be reset cleanly. Hierarchical to: No other components Dependencies: - ADV_SPM.1 Informal TOE security policy model Management: ---
FPT_PHP **Physical Protection**	
FPT_PHP.3 **Resistance to Physical Attack**	PP9911
FPT_PHP.3.1 The TSF shall resist [assignment: *physical tampering scenarios*] to the [assignment: *list of TSF devices*	**FPT_PHP.3-1:** **FPT_PHP.3.1-1**

/elements] by responding automatically such that the TSP is not violated. Hierarchical to: No other components Dependencies: No dependencies Management: a) management of the automatic responses to physical tampering Audit: ---	The TSF shall resist [**side channel attacks like SPA-attacks, DPA-attacks, DFA-attacks and timing attacks concerning all critical cryptographic operations**] to the [**TSF interfaces**] by responding automatically such that the TSP is not violated. Hierarchical to: No other components Dependencies: No dependencies Management: Not applicable
FPT_SEP **Domain Separation**	
FPT_SEP.1 **TSF Domain Separation**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.7.2
FPT_SEP.1.1 The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects. **FPT_SEP.1.2** The TSF shall enforce separation between the security domains of subjects in the TSC. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: ---	**FPT_SEP.1-1:** **FPT_SEP.1.1-1** The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects. **FPT_SEP.1.2-1** The TSF shall enforce separation between the security domains of subjects in the TSC. **Refinements** RLB_304: There shall be no way to analyse, debug or modify TOE's software in the field. RLB_305: Inputs from external sources shall not be accepted as executable code. Hierarchical to: No other components Dependencies: No dependencies Management: ---
FPT_TDC **Inter-TSF TSF Data Consistency**	
FPT_TDC.1 **Inter-TSF Basic TSF Data Consistency**	PP9911
FPT_TDC.1.1 The TSF shall provide the capability to consistently	**FPT_TDC.1-1:**

interpret [assignment: *list of TSF data types*] when shared between the TSF and another trusted IT product. **FPT_TDC.1.2** The TSF shall use [assignment: *list of interpretation rules to be applied by the TSF*] when interpreting the TSF data from another trusted IT product. Hierarchical to: No other components Dependencies: No dependencies Management: --- Audit: a) Minimal: Successful use of TSF data consistency mechanisms b) Basic: Use of the TSF data consistency mechanisms c) Basic: Identification of which TSF data have been interpreted d) Basic: Detection of modified TSF data	**FPT_TDC.1.1-1** The TSF shall provide the capability to consistently interpret [- **authentication tokens with their input data for session keys and send sequence counters** - **session keys and send sequence counters themselves** - **PINs and their formats** - **imported certificates, their format and their included signature** - **imported signatures for verification**] when shared between the TSF and another trusted IT product. **FPT_TDC.1.2-1** The TSF shall use [- **rules for the interpretation of the input data for session keys and send sequence counters within authentication tokens for the creation of session keys and send sequence counters: Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 4, 3.2, Appendix 2, chap. 3.6.8, 3.6.9** - **rules for the interpretation of session keys and send sequence counters within Secure Messaging: Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 5, 3.2, Appendix 2, chap. 3.6.2.2, 3.6.3.2** - **rules for the interpretation of imported PINs: Tachograph Card specification** /TachAn1B/**, Appendix 2, chap. 3.6.5** - **rules for the interpretation of imported certificates, their format and their included signature: Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 3.3** - **rules for the interpretation of imported signatures for verification: Tachograph Card specification** /TachAn1B/**, Appendix 11, chap. 6.2**] when interpreting the TSF data from another trusted IT product. Hierarchical to: No other components Dependencies: No dependencies Management: ---
FPT_TST	

TSF Self Test	
FPT_TST.1 **TSF Testing**	PP9911 / TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.7.1
FPT_TST.1.1 The TSF shall run a suite of self tests [selection: *during initial start-up, periodically during normal operation, at the request of the authorised user, at the conditions* [assignment: *conditions under which self test should occur*]] to demonstrate the correct operation of 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 stored TSF executable code. Hierarchical to: No other components Dependencies: - FPT_AMT.1 Abstract machine testing Management: a) management of the conditions under which TSF self testing occurs, such as during initial start-up, regular interval, or under specified conditions b) management of the time interval if appropriate Audit: a) Basic: Execution of the TSF self tests and the results of the tests	**FPT_TST.1-1:** **FPT_TST.1.1-1** The TSF shall run a suite of self tests [**during initial start-up, periodically during normal operation**] to demonstrate the correct operation of the TSF. **Note** During initial start-up means before code is executed. **Refinements** RLB_301: The TOE's self tests shall include the verification of the integrity of any software code not stored in ROM. RLB_302: Upon detection of a self test error the TSF shall warn the entity connected. RLB_303: After operating system testing is completed, all testing-specific commands and actions shall be disabled or removed. It shall not be possible to override these controls and restore them for use. Command associated exclusively with one life cycle state shall never be accessed during another state. The term “periodically during normal operation” is understood as follows: It is assumed that the TOE performs at least one reset-operation each day, so that the self test at each initial start-up suffices the requirement of performing the self test periodically during normal operation. **FPT_TST.1.2-1** The TSF shall provide authorised users with the capability to verify the integrity of TSF data. **Refinement** In this framework, the Smartcard Embedded Software of the TOE (TOE-ES) itself is understood as „authorised user“. **FPT_TST.1.3-1** The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code. **Refinement** This requirement concerns only the production phase, more precise the initialisation phase of the TOE (phase 5 of the product´s life cycle). Prior to the initialisation of the TOE, the ROM-code of the TOE shall be verifiable by the Smartcard Embedded Software developer. The integrity of the EEPROM-code shall be provable by the TOE during the initialisation process.

	Hierarchical to: No other components Dependencies: Not applicable Management: Not applicable

FTP **Trusted Path/Channels**	
FTP_ITC **Inter-TSF Trusted Channel**	
FTP_ITC.1 **Inter-TSF Trusted Channel**	TachAn1B, Appendix 10, Tachograph Card Generic Security Target, chap. 4.8.1
FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. **FTP_ITC.1.2** The TSF shall permit [selection: *the TSF, the remote trusted IT product*] to initiate communication via the trusted channel. **FTP_ITC.1.3** The TSF shall initiate communication via the trusted channel for [assignment: *list of functions for which a trusted channel is required*]. Hierarchical to: No other components Dependencies: No dependencies Management: a) Configuring the actions that require trusted channel, if supported Audit: a) Minimal: Failure of the trusted channel functions b) Minimal: Identification of the initiator and target of failed trusted channel functions c) Basic: All attempted uses of the trusted channel functions d) Basic: Identification of the initiator and target of all	**FTP_ITC.1-1:** **FTP_ITC.1.1-1** The TSF shall provide a communication channel between itself and a remote trusted IT product **(vehicle unit, personalisation unit)** 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. **Refinements** DEX_301: The TOE shall verify the integrity and authenticity of data imported from a vehicle unit. DEX_302: Upon detection of an imported data integrity error, the TOE shall: - warn the entity sending the data, - not use the data. DEX_303: The TOE shall export user data to the vehicle unit with associated security attributes, such that the vehicle unit will be able to verify the integrity and authenticity of data received. **Note** The integrity and authenticity resp. the confidentiality, if required, of the data transfer between the Tachograph Card and the remote trusted IT product (vehicle unit, personalisation unit) shall be conducted with Secure Messaging in accordance with ISO/IEC 7816-4. **FTP_ITC.1.2-1** The TSF shall permit [**the remote trusted IT product**] to initiate communication via the trusted channel.

trusted channel functions	**FTP_ITC.1.3-1** The TSF shall initiate communication via the trusted channel for [**user data import from a remote trusted IT product, user data export to a remote trusted IT product**]. Hierarchical to: No other components Dependencies: No dependencies Management: Not applicable

5.1.2 SOF Claim for TOE Security Functional Requirements

According to the requirements in the Tachograph Card specification /TachAn1B/, Annex 1B main body and Appendix 10 (Tachograph Card Generic Security Target), and to the JIL interpretations /JILDigTacho/, the required level for the Strength of Function of the TOE security functional requirements listed in the preceding chap. 5.1.1 is “SOF-high”. This correlates to the claimed assurance level with its augmentation by the assurance component AVA_VLA.4 (refer to the following chap. 5.1.3).

5.1.3 TOE Security Assurance Requirements

The evaluation of the Tachograph Card according to ITSEC E3 high as required in the Tachograph Card Specification /TachAn1B/, Appendix 10 (Tachograph Card Generic Security Target) will be replaced by a comparable evaluation according to Common Criteria, whereat the requirements in the JIL interpretations /JILDigTacho/, Annex A have to be considered. The TOE security assurance level is fixed as

EAL4 augmented by ADO_IGS.2, ADV_IMP.2, ATE_DPT.2 and AVA_VLA.4,

thus the CC evaluation of the TOE matches the evaluation assurance requirements stated in the Tachograph Card Specification /TachAn1B/, Appendix 10 (Tachograph Card Generic Security Target).

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

SAR	
Class ACM **Configuration Management**	ACM_AUT.1 Partial CM Automation

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

Class AVA Vulnerability Assessment	AVA_MSU.2 Validation of Analysis
	AVA_SOF.1 Strength of TOE Security Function Evaluation
	AVA_VLA.**4** Highly Resistant

5.1.4 Refinements of the TOE Security Assurance Requirements

All assurance components given in the table of chap. 5.1.3 are used as defined in /CCPart3/ and /CEMPart2/. Additionally, according to /JILDigTacho/, Annex A.3, Note 2 and 9 the following refinements resp. interpretations are taken into account:

ADO_IGS.2

ADO_IGS.2 is interpreted resp. refined according to ITSEC E3.32 and ITSEC-JIL, Section 16.2 as follows:

- The term “generation” is always interpreted as “installation”.
- "While installing the TOE, any configuration options and/or changes shall be audited in such a way that it is subsequently possible to reconstruct exactly how the TOE was initially configured and when the TOE was installed."

AVA_MSU.2

ITSEC 3.33 additionally requires evaluator tests where necessary. This testing, can be part of the penetration testing under AVA_VLA. It is decided on a case by case basis if the evaluator performs misuse-testing as additional part of penetration testing to confirm or disprove the misuse analysis. Specifically, if high attack potential is assumed, such independent misuse-testing is performed.

5.2 Security Requirements for the Environment of the TOE

5.2.1 Security Requirements for the IT-Environment

There are no security requirements for the IT-Environment of the TOE defined.

5.2.2 Security Requirements for the Non-IT-Environment

There are no security requirements for the Non-IT-Environment of the TOE defined.

6 TOE Summary Specification

6.1 TOE Security Functions

6.1.1 TOE Security Functions / TOE-IC

The following section gives a survey of the TSFs of the IC as defined in the corresponding document /ST-ICPhilips/, chap. 6.1 and the Security Target related to the evaluation of the IC incl. its Crypto Library.

Note:

For clarity, within the description of the TSFs in the following table the word „TOE“ as given in /ST-ICPhilips/ is replaced by „TOE-IC“.

TOE Security Functions / TOE-IC	
F.RNG	**Random Number Generator**
	The random number generator continuously produces random numbers with a length of one byte. The TOE-IC implements the F.RNG by means of a physical hardware random number generator working stable within the limits guaranteed by F.OPC (operational conditions). According to AIS31 the random number generator claims the fulfilment of the requirements of functionality class P2. This means that the random number generator is suitable for generation of signature key pairs, generation of session keys for symmetric encryption mechanisms, random padding bits, zero-knowledge proofs and generation of seeds for DRNGs.
F.HW_DEA	**Triple-DES Co-processor**
	The TOE-IC provides the Triple Data Encryption Algorithm (TDEA) according to the Data Encryption Standard (DES). F.HW_DEA is a modular basic cryptographic function which provides the TDEA algorithm as defined by FIPS PUB 46 by means of a hardware co-processor and supports the 2-key Triple DEA algorithm according to keying option 2 in FIPS PUB 46-3. The two 56 bit keys (112 bit) for the 2-key Triple DES algorithm shall be provided by the Smartcard Embedded Software. For encryption the Smartcard Embedded Software provides 8 bytes of the plain text and F.HW_DEA calculates 8 bytes cipher text. The calculation output is read by the Smartcard Embedded Software. For decryption the Smartcard Embedded Software also provides 8 bytes of cipher text and F.HW_DEA calculates 8 bytes plain text. The calculation output is read by the Smartcard Embedded Software.
F.OPC	**Control of Operating Conditions**
	The function F.OPC ensures the correct operation of the TOE-IC (functions offered by the microcontroller including the standard CPU as well as the Triple-DES co-processor, the arithmetic coprocessor, the memories, registers, I/O interfaces and the other system peripherals)

	during the execution of IC Dedicated Support Software and Smartcard Embedded Software. This includes all specific security features of the TOE-IC which are able to provide an active response. The TOE-IC ensures its correct operation and prevents any malfunction using the following subfunctions: filtering of power supply and clock input as well as monitoring of power supply, the frequency of the clock and the temperature of the chip by means of sensors. The thresholds allowed for these parameters are defined within the range where the TOE-IC ensures its correct operation. If one of the monitored parameters is out of the specified range a reset is forced and the actual running program is aborted. All components of the TOE-IC are initialised with their reset values. Before TOE-IC delivery the TOE-IC mode is set to Application Mode. In Application Mode the TOE-IC enables the sensors automatically when operated. Furthermore it prevents that the Smartcard Embedded Software disables the sensors. In addition, the TOE-IC controls the specified range of the System Stack Pointer and the User Stack Pointer. In case the specified limits are reached an exception is generated. Beside the sensors the security function comprises an additional sensor to check the high voltage for the write process to the EEPROM during every write sequence. The result of this sensor must be read from a Special Function Register and does not force an automatic event (e.g. reset).
F.PHY	**Protection against Physical Manipulation**
	The function F.PHY protects the TOE-IC against manipulation of (i) the hardware, (ii) the IC Dedicated Test Software in the ROM, (iii) the Smartcard Embedded Software in the ROM and the EEPROM, (iv) the application data in the EEPROM and RAM, (v) the configuration data in the security row, (vi) the control of the TOE-IC mode and (vii) the OTP-area. It also protects User Data or TSF data against disclosure by physical probing when stored or while being processed by the TOE-IC. The protection of the TOE-IC comprises different features within the design and construction which make reverse-engineering and tamper attacks more difficult. These feature comprise dedicated shielding techniques and different scrambling features for the memory blocks. The security function F.PHY supports the efficiency of other security functions.
F.LOG	**Logical Protection**
	The function F.LOG implements measures to limit or eliminate the information that might be contained in the shape and amplitude of signals or in the time between events found by measuring such signals. This comprises the power consumption and signals on the other pads that are not intended by the terminal or the Smartcard Embedded Software. Thereby this security function prevents the disclosure of User Data or TSF data stored and/or processed in the Smartcard IC through the measurement of the power consumption and subsequent complex signal processing. The protection of the TOE-IC comprises different features within the design that support the other security functions. The Triple-DES co-processor includes special features to prevent SPA/DPA analysis of shape and amplitude of the power consumption and ensures the same calculation time for all operations. The FameX co-processor provides measures to prevent timing attacks on basic modular func-

	tion. The calculation time of one modular operation depends on the lengths of the operands but not on the value of the operands. In addition special features are included to provide limitations of the capability for the analysis of shape and amplitude of the power consumption. Of course the FameX does not realise an algorithm on its own and algorithm-specific leakage countermeasures have to be added for the FameX. Additional features that can be configured by the Smartcard Embedded Software comprise (i) the secure DCDC-converter that can be used to smooth the power consumption and (ii) several clock configurations that can be used to prevent the possibility to synchronise the internal operation with the external clock or to synchronise with the characteristics of the power consumption that can be used as trigger signal to support leakage attacks (DPA or timing attacks). Specific features as described for the function F.PHY (e.g. the scrambling features) and for the function F.OPC (e.g. the filter feature) support the logical protection. The TSF F.LOG includes software protection against attacks by externally measuring the power consumption of the Smart*XA*2 processor (Simple Power Attack (SPA) or Differential Power Attack (DPA)) or measuring the execution time (as required by FDP_ITT.1, FPT_ITT.1 and FDP_IFC.1): - The DES and Triple-DES algorithm is power and timing attack resistant. - The RSA algorithm is power and timing attack resistant. - The TSF F.LOG includes software protection against DFA attacks (as required by FPT_FLS.1) for the Triple-DES and the RSA algorithm.
F.COMP	**Protection of Mode Control**
	The function F.COMP provides a control of the TOE-IC mode for (i) Test Mode and (ii) Application Mode. This includes the protection of electronic fuses stored in a protected memory area. In addition F.COMP provides a write once memory area. All bits in this area can only be set once. The control of the TOE-IC mode prevents the use of features implemented in the TOE-IC that are used during production/test and that are disabled before the delivery of the TOE-IC. The initial TOE-IC mode is the Test Mode. F.COMP limits the capabilities of the test functions and provides test personnel during phase 3 with the capability to store the identification and/or pre-personalisation data and/or supplements of the Smartcard Embedded Software in the EEPROM. The implemented control of the TOE-IC mode ensures that in the Test Mode the TOE-IC (i) allows to execute the IC Dedicated Test Software and (ii) prevents to execute the Smartcard Embedded Software. In the Application Mode the TOE-IC (i) allows to execute the Smartcard Embedded Software and (ii) prevents to execute the IC Dedicated Test Software. The protection of electronic fuses ensures the secure storage of configuration- and calibration data stored in the Test Mode. The TOE-IC allows to change the TOE-IC mode only one time from the Test Mode into the Application Mode. The TOE-IC prevents to change the TOE-IC mode from the Application Mode into the Test Mode. The write once memory area is erased during the Test Mode to ensure a well defined content. After the switch to the Application Mode the Smartcard Embedded Software is able to read this memory area and to set every bit in this memory area once. The access to the OTP user memory area is only possible in the system mode or in the meta mode. The OTP user memory area is designed to store the identification of a dedicated smartcard or a sequence of events over the life cycle that can be coded by an increasing number of bits set to "one".

	The security function F.COMP maintains the security domain for its own execution that protects it from interference and tampering by untrusted subjects both in the Test Mode and in the Application Mode. It also enforces the separation between the security domains of subjects within each mode of operation. The OTP memory is maintained in the same way to ensure the settings stored in that memory area.
F.MEM_ACC	**Memory Access Control**
	F.MEM_ACC controls access of any subject (program code comprising processor instructions) to the memory of the TOE-IC through the Code Memory Management Unit and the Data Memory Management Unit. Thereby the code is also stored in a dedicated memory area. Based on the value of the Special Function Register “Program Status Word” the processor is assigned to (i) System Mode, (ii) Meta Mode or (iii) User Mode. In the System Mode both the Code MMU and Data MMU are transparent. In the Meta Mode both MMU’s are enabled. Memory access is based on virtual addresses that are mapped to physical addresses. The CPU uses virtual addresses, physical addresses are used to access the memories. The Memory Management Units perform the translation from virtual to physical addresses. The access control is performed by the definition of memory areas with related access rights. It is possible to define several code memory areas at the same time with the access rights read, write, execute and enable/disable. It is possible to define several data memory areas at the same time with the access rights read, write and enable/disable. A disabled MMU is transparent and performs no address translation and no access control. Instead, the virtual addresses are mapped one-to-one to physical addresses. An enabled MMU performs both tasks. In addition the memory management units permanently check whether the selected addresses are within the boundary of the physical implemented memory range. Access violations and accesses outside the boundary of the physical implemented memory range are notified by raising an exception.
F.SFR_ACC	**Special Function Register Access Control**
	The function F.SFR_ACC controls access to the Special Function Registers and the switch between the System Mode, the Meta Mode and the User Mode. Based on the value of the Special Function Register “Program Status Word” the processor is assigned to (i) System Mode, (ii) Meta Mode or (iii) User Mode. The access control is as follows: - In System Mode, all Special Function Registers are accessible. - In Meta Mode, all Special Function Registers are accessible except the Special Function Registers to configure the Code MMU and the SCR Register that are only readable. - In User Mode, only the Special Function Registers related to General CPU Functions are accessible. This implies that the security functions Random Number Generator and Triple-DES Coprocessor can only be used and the security function Logical Protection can only be configured in System Mode or Meta Mode. In addition also the FameX co-processor and all Special Function Register of Specialised Components are only accessible in the System Mode or the Meta Mode. Only the Special Function Register related to General CPU Functions including most registers of the Value Range Check are directly accessible in the User Mode. For the Memory Access Control and the Value Range Check refer to the definition of the related security function. Based on the function of the register, on the mode of operation or on the TOE-IC mode, the read and/or write access for a specific SFR is not allowed (e.g. read access to DES key reg-

	ister or write access to the output register of the random number generator). There are two different implementations to prevent an access. The first method provides an exception and the second method will ignore any operation on the SFR. Ignored means that the write access has no influence and/or that the read access always provides a fixed return value independent of the content of the SFR. One of these two methods is implemented for the affected register. Regardless of the Mode of Operation the respective bits in the “Program Status Word” which store the Mode of Operation can not be changed by direct access. Instead, the bits can only be changed by invoking an exception or an interrupt or returning from the respective routine. When an interrupt is invoked, the actual state of the PSW register is stored on the stack and a new value is set from the interrupt vector table. When an interrupt is finished, the previously saved value of the PSW is restored.
F.RANGE_-CHK	**Value Range Check**
	The function F.RANGE_CHK provides range checking for dedicated registers of the TOE-IC. Range checking comprises checking against lower and upper bounds. Any violation of the allowed range is notified via an interrupt. Range checking is performed on the following registers: - R15/AP1 register accessible in System, Meta and User Mode - CSFVAL Special Function Register write only in System, Meta and User Mode Note: Although this security function (as a hard-wired functionality) can not be disabled, the range checking can be stopped by setting the lower and upper bound to the minimum and maximum values that the register is able to store. The reset values are 0000h for the lower limit and FFFFh for the upper limit. Therefore the security function must be enabled by loading more restrict values.
F.DES	**DES Operation**
	The TOE-IC uses the Smart*XA*2 DES hardware coprocessor to provide a DES encryption and decryption facility using 56-bit keys, and to provide Triple-DES encryption and decryption. Note that only the Triple-DES encryption and decryption is within the scope of the evaluation of the TOE-IC. This functionality is required by the security functional component FCS_COP.1 taken from the Common Criteria Part 2. The Triple-DES function uses double-length or triple-length keys with sizes of 112 or 168 bits respectively. The supported modes are ECB and “outer” CBC. F.DES is a modular basic cryptographic function which provides the DES algorithm as defined by the standard FIPS 46-3, and supports the 2-key and 3-key Triple-DES algorithm according to ANSI Standard X9.52 - 1998. Note that, for the evaluated TOE-IC, it is permitted to use only the Triple-DES configuration of the TSF for encryption or decryption. SPA/DPA, timing and DFA attack resistance for F.DES is discussed under the TSF F.LOG.
F.RSA	**RSA Operation**
	The Crypto Library provide functions that implement the RSA algorithm as described in Schneier, Angewandte Kryptographie, page 468 or Menezes, van Oorshot and Vanstone, Handbook of Applied Cryptography, section 8.2, and also mentioned in the standard ISO/IEC 9796, Annex A, section A.4. This functionality is required by the security functional component FCS_COP.1 taken from the Common Criteria Part 2. This routine supports various key lengths

	from 128 bits to 2048 bits. Note that, for the evaluated TOE-IC, RSA keys must have a key length in the range 1024 to 2048 bits. The TOE-IC contains modular exponentiation functions, which, together with other functions in the TOE-IC, perform the operations required for RSA encryption or decryption. Two different RSA algorithms are supported by the TOE-IC, namely the “Simple Straight Forward Method” and the “Chinese Remainder Theorem”. SPA/DPA, timing and DFA attack resistance for F.RSA is discussed under the TSF F.LOG.
F.SHA-1	**SHA-1 Computation**
	The TOE-IC implements functions to compute the Secure Hash Algorithm SHA-1 according to the standard FIPS 180-1. This functionality is required by the security functional component FCS_COP.1 taken from the Common Criteria Part 2. The SHA-1 algorithm generates an output of length 160 bits.
F.RNG_-Access	**Generation of Random Numbers**
	The TOE-IC contains both a hardware Random Number Generator (RNG) and a software RNG. For the hardware RNG see TSF F.RNG. The TSF F.RNG_Access consists of the implementation of the software RNG (FCS_RND.2) and of appropriate online tests (FPT_TST.2): The Crypto Library implements a software (pseudo) RNG that can be used as a general purpose random source. This software RNG has to be seeded by random numbers taken from the hardware RNG contained in the Smart*XA*2 processor, after appropritate online tests - which are also implemented within the Crypto Library - have been carried out. These tests are required by SFR FPT_TST.2. The software RNG is implemented according to FIPS 186-2 with Change Notice 1, as specified in the SFR FCS_RND.2.
F.Object_-Reuse	**Reuse of Objects**
	The TOE-IC provides internal security measures which include clearing of relevant parts of the memory involved in security operations after usage. This functionality is required by the security functional component FDP_RIP.1 taken from the Common Criteria Part 2. The TOE-IC also provides a function whereby the TOE-IC environment can explicitly clear the Fame*X* RAM on request. These measures ensure that a subsequent process may not gain access to cryptographic assets stored temporarily in memory used by the TOE-IC. Note: This TSF does not cover erasing the memory contents of the Smart*XA*2 processor after an interruption (loss of power or RESET). Further, this TSF does not include erasing any memory contents of an application program run in the smart card under the control of the operating system. The deletion of memory contents in these situations is the responsibility of the IT environment of the TOE-IC.

6.1.2 TOE Security Functions / TOE-ES

The following section gives a survey of the TSFs of the TOE´s Smartcard Embedded Software under consideration of the requirements in the Tachograph Card Specification /TachAn1B/, Appendix 10 (Tachograph Card Generic Security Target).

TOE Security Functions / TOE-ES	
Access Control	
F.ACS	**Security Attribute Based Access Control**
	The TSF enforces for the personalisation phase of the TOE the SFP Personalisation Access Control (PERS-AC_SFP) and for the end-usage phase of the TOE the SFP Access Control (AC_SFP) as defined in chap. 5.1.1.2.1.
Identification and Authentication	
F.IA_KEY	**Key Based User / TOE Authentication**
	Users of the TOE can be authenticated with regard to the TOE by means of a challenge-response procedure using random numbers (external authentication). Vice versa, the TOE itself can be authenticated with regard to the external world as well by means of a challenge-response procedure using random numbers (internal authentication). In both cases, the TSF makes use of asymmetric cryptography (with encryption, decryption, generation of a digital signature resp. verification of a digital signature) and of the generation of random numbers and is therefore connected with the TSFs F.ENC, F.DEC, F.GEN_DIGSIG, F.VER_DIGSIG and F.RND_Access. For an internal authentication, the TSF generates and returns an authentication token by using the operations “generation of a digital signature” and “encryption” on random numbers of the external world and of the TOE itself. In detail, the TSF uses the relevant private key to sign the authentication data including the randoms and then uses the public key currently selected to encrypt the signature and form the authentication token which will be returned to the external world. For an external authentication, the TSF verifies an authentication token delivered by the external world (containing random numbers of the external world and of the TOE itself) by using the operations “decryption” and “verification of a digital signature”. In detail, the TSF uses the currently selected public key to decrypt the authentication token and uses then the relevant private key to verify the signature within the delivered authentication token. The external authentication process needs a preceding Get Challenge - operation. The private key necessary on the card´s side for authentication purposes is stored on the card (during initialisation resp. personalisation of the TOE) and is implicitly connected with the corresponding commands. The necessary public keys whereas are already stored on the card or have to be imported in the form of certificates. In each case, they have to be explicitly referenced for usage. The import of a public key by a certificate is connected with the verification of the respective certificate under use of the TSF F.VER_DIGSIG. The access to the keys is controlled by the SFP Personalisation Access Control (PERS-AC_SFP) within the personal-

	isation phase of the TOE and by the SFP Access Control (AC_SFP) within the end-usage phase of the TOE as defined in chap. 5.1.1.2.1, which is realised by the TSF F.ACS. In case of a successful external authentication attempt a corresponding actual security state is set. The combination of a successful internal authentication process followed by a successful external authentication process leads to the generation of a new session key (with send sequence counter) which will be used for securing the following data transfer. In detail, the following conditions are valid: If the internal authentication process does not fail, the current session key, if existing, is erased and no longer available. In order to have a new session key available, a following external authentication process must be successfully performed. If the external authentication does not fail, and if the first part of the session key is available from the preceding successful internal authentication, the session key is generated and set for future commands using Secure Messaging. If the first session key part is not available from a previous internal authentication, the second part of the session key, sent by the external world within the authentication token, is not stored in the card. The generation of session keys is task of the TSF F.GEN_SES. For the Tachograph card type Workshop Card the mutual authentication process described above is only possible after a successful preceding password based user authentication (see F.IA_PWD).
F.IA_PWD	**Password Based User Authentication** (only relevant for the Tachograph Card type Workshop Card)
	Users of the TOE can be authenticated by means of a card holder authentication process. For the card holder authentication process, the TSF compares the cardholder verification information, here a password (PIN), provided by a subject with a corresponding secret reference data stored in the card. The TSF is internally connected with the card´s unique password stored on the card (loaded in the framework of the TOE´s personalisation). The access to the password is controlled by the SFP Access Control (AC_SFP) as defined in chap. 5.1.1.2.1, which is realised by the TSF F.ACS. The TSF detects when a defined number of consecutive unsuccessful authentication attempts occurs related to the card holder authentication process. Each consecutive unsuccessful comparison of the presented password with the reference value stored on the card is recorded by the TSF in order to limit the number of further authentication attempts with the password. For this purpose, the TSF manages a mandatory error counter for the password. In case of a successful authentication attempt a corresponding actual security state for the password is set and the error counter is reinitialised. Within the actual implementation of the TOE, the security state reached by a successful password based user authentication will not be evaluated by the TOE. If an authentication attempt with the password fails, the corresponding actual security state is reset and the password´s error counter is decreased. When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF blocks the corresponding password. There is no way to reset the error counter in order to unblock the password so that the password is invalid for each further authentication process. For security reasons, the initial value for the error counter is set to a sufficiently small finite value (here: 5). The TSF does not check the quality of the used password, this check is in responsibility of the external world. Furthermore, there is no possibility to change the password while the card is in

	operational status. The transfer of the password to the TOE for authentication attempts is executed in unsecured mode (i.e. without use of Secure Messaging) or optional in secured mode with Secure Messaging. In the latter case, the TSFs F.EX_CONF and F.EX_INT are involved.
Integrity of Stored Data	
F.DATA_INT	**Stored Data Integrity Monitoring and Action**
	The TSF monitors data stored within the TOE for integrity errors. This concerns all elementary files and dedicated files as well as all secrets (esp. passwords and cryptographic keys) stored outside the filesystem within the EEPROM area. The monitoring is based on the following attributes: - a checksum (CRC) attached to each header of a file - a checksum (CRC) attached to the data contained in a file - a checksum (CRC) attached to each secret stored outside the filesystem within the EEPROM area Before the TOE accesses to an elementary or dedicated file or a secret stored outside the filesystem, the TSF carries out an integrity check on base of the mentioned attributes. Upon detection of a data integrity error, the TSF informs the user about this fault. If the checksum of the header of a file has been detected as corrupted, the data contained in the affected file is no longer accessible. If the data contained in a file is not of integrity, the affected data will be treated in the following way: - For the Read access, the affected data will be exported, but the data export will be connected with a warning. - For the Update access, the integrity error of the affected data will be ignored, and the data imported by the command will be stored and a new checksum will be computed. - For all remaining access modes, the affected data will not be used for data processing. If a secret stored outside the filesystem is corrupted, the secret will not be processed.
Data Exchange	
F.EX_CONF	**Confidentiality of Data Exchange**
	The TSF provides the capability to ensure that secret data which is exchanged between the TOE and the user remains confidential during transmission. For this purpose, encryption based on symmetric cryptography is applied to the secret data. The TSF ensures that the user and the user data's access condition have indicated confidentiality for the data exchange. Securing the data transfer with regard to data confidentiality will be done by Secure Messaging according to the standards ISO/IEC 7816-4. The cryptographic keys used for securing the data transfer are session keys which are generated during a preceding mutual authentication process between the card and the external world which is realised by the TSFs F.IA_KEY and F.GEN_SES).

	For encryption, the TSF makes use of the TSF F.DES of the underlying IC resp. its Dedicated Support Software.
F.EX_INT	**Integrity and Authenticity of Data Exchange**
	The TSF provides the capability to ensure that data which is exchanged between the TOE and the user remains integer and authentic during transmission. For this purpose, cryptographic checksums based on symmetric cryptography are applied to the data. The TSF ensures that the user and the user data's access condition have indicated integrity and authenticity for the data exchange. Securing the data transfer with regard to data integrity and authenticity will be done by Secure Messaging according to the standards ISO/IEC 7816-4. The cryptographic keys used for securing the data transfer are session keys which are generated during a preceding mutual authentication process between the card and the external world which is realised by the TSFs F.IA_KEY and F.GEN_SES). For checksum securing, the TSF makes use of the TSF F.DES of the underlying IC resp. its Dedicated Support Software.
Object Reuse	
F.RIP	**Residual Information Protection**
	The TSF ensures that any previous information content of a resource is explicitly erased upon the allocation of the resource used for any of the following components: - volatile and non-volatile memories used for operations in which security relevant material (e.g. secret keys or other secrets like passwords) is involved The TSF makes use of the TSF F.Object_Reuse of the underlying IC resp. its Dedicated Support Software.
Protection	
F.FAIL_-PROT	**Hardware and Software Failure Protection**
	The TSF preserves a secure operation state of the card when the following types of failures occur: - induced hardware or software failures (transient or permanent) during the execution of an operation resp. command - tampering The TSF makes use of hardware and software based security features and corresponding mechanisms to monitor and detect induced hardware and software failures and tampering attacks. In particular, the TSF is supported by the IC specific TSFs F.OPC and F.PHY. Upon the detection of a failure of the above mentioned type the TSF reacts in such a way that the TSP is not violated. The TOE changes immediately to a locked state and cannot be used any longer within the actual session. Depending on the type of the detected attack to the underlying IC (incl. its Dedicated Software) or to the Smartcard Embedded Software code the TOE will be irreversible locked resp. can be reactivated by a reset.

F.SIDE_-CHAN	**Side Channel Analysis Control**
	The TSF manages suitable hardware and software based mechanisms to prevent attacks by a side channel analysis like Simple Power Analysis (SPA), Differential Power Analysis (DPA), Differential Fault Analysis (DFA) and Timing attacks. The TSF ensures that all countermeasures available are used in such a way that they support each other. In particular, the TSF is supported by the TSF F.LOG of the underlying IC and its Dedicated Support Software. The TSF acts in such a manner that all security relevant operations of the TOE (esp. the TOE´s cryptographic operations) are suitably secured by these hardware and software countermeasures. The TSF enforces that a secure session is installed before any cryptographic key is generated, loaded into volatile / non-volatile memories (esp. of dedicated IC cryptographic modules) and processed in a cryptographic operation or in an authentication process.
F.SELFTEST	**Self Test**
	The TSF provides the capability of conducting a self test during initial start-up, i.e. after each reset to demonstrate the correct operation of its TSFs. Under the assumption that the TOE performs at least one reset-operation each day, the self test fulfills the requirement of being performed periodically during normal operation. The TOE´s self tests consist of the verification of the integrity of any software code stored in the EEPROM area by checking a related checksum of the code. Furthermore, the TSF provides authorised users - here the Smartcard Embedded Software of the TOE (TOE-ES) itself - with the capability to verify the integrity of TSF data. For this task, the TSF is supported by the TSF F.DATA_INT. Additionally, the TSF provides authorised users with the capability to verify the integrity of stored TSF executable code. This concerns only the production phase, more precise the initialisation phase of the TOE (phase 5 of the product´s life cycle). Prior to the initialisation of the TOE, the ROM-code of the TOE can be verified by the Smartcard Embedded Software developer. The integrity of the EEPROM-code is checked by the TOE during the storage of the initialisation file in the framework of the initialisation. The TSF supports all other TSFs defined for the Smartcard Embedded Software (TOE-ES).
Cryptographic Operations	
F.GEN_SES	**Generation of Session Keys**
	The TSF generates session keys for symmetric cryptography used for securing the data exchange between the TOE and the external world with regard to data confidentiality and data integrity and authenticity. The TSF enforces that the key material meets the following requirements: - random numbers generated by the card and used in the key generation process have a high quality - the symmetric keys generated by the TOE are checked by the TSF with regard to their cryptographic strength, and only cryptographically strong keys (with the required key length) will be accepted by the TSF

	The TSF for generation of session keys is connected with the TSF F.RNG_Access for the generation of random numbers with high quality. Furthermore, the TSF for generation of session keys is directly connected with the TSF F.IA_KEY which realises the internal and external authentication process.
F.GEN_DIG SIG	**Generation of Digital Signatures**
	The TSF provides a digital signature functionality based on asymmetric cryptography, particularly the RSA algorithm with a key length of 1024 bit. The TSF digital signature function will be used for several purposes with different signature keys and different formats for the digital signature input: - Explicit generation of digital signatures of data using the signature scheme with appendix (signature generation operation) according to the standard PKCS#1 V2.0 and with hash algorithm SHA-1. In this case, the TSF digital signature function is implicitly combined with the Tachograph Card´s dedicated and unique private signature key stored in the card. - Within authentication processes for the creation of authentication tokens using the signature scheme with message recovery (signature generation operation) according to the standard ISO 9796-2 and with hash algorithm SHA-1. Within the Tachograph Application, the TSF digital signature function is implicitly combined with the Tachograph Card´s dedicated private personalisation key (during the personalisation phase) resp. with the Tachograph Card´s dedicated and unique private signature key (during the end-usage phase of the card). Within the Card Manager Application, the TSF uses the card´s dedicated private authentication key which is intended for the proof of the card´s authenticity. Random numbers necessary for the generation of digital signatures are generated by using the TSF F.RND_Access of the underlying IC resp. its Dedicated Support Software for random number generation. For the signature mechanism itself, the TSF makes use of the TSF F.RSA of the underlying IC resp. its Dedicated Support Software. For the computation of hash values the TSF F.SHA-1 of the underlying IC resp. its Dedicated Support Software is used. Furthermore, the security of the TSF is supported by the TSFs F.LOG and F.SIDE_CHAN of the IC and its Dedicated Support Software resp. the Smartcard Embedded Software. Note: Each pivate key used for the signature generation function is generated by the external world and loaded onto the card (during initialisation resp. personalisation). It is in the responsibility of the external world to guarantee for a sufficient cryptographic strength of the private key and to handle the private key outside the card in a sufficient secure manner. Under the assumption that the external world meets the requirements on the key handling set above, the TSF digital signature function works in such a manner that the private key cannot be derived from the signature and the signature cannot be generated by other individuals not possessing that secret. Furthermore, the TSF digital signature function works in a manner that no information about the private key may be disclosed during the generation of the digital signature.
F.VER_-DIGSIG	**Verification of Digital Signatures**
	The TSF provides a functionality to verify digital signatures based on asymmetric cryptography, particularly the RSA algorithm with a key length of 1024 bit. The TSF function to verify a digital signature will be used for several purposes with different keys and different formats for the digital signature input:

	- Explicit verification of digital signatures of data using the signature scheme with appendix (signature verification operation) according to the standard PKCS#1 V2.0 and with hash algorithm SHA-1. - Within authentication processes for the verification of authentication tokens using the signature scheme with message recovery (signature verification operation) according to the standard ISO 9796-2 and with hash algorithm SHA-1. - Within the verification and unwrapping of imported certificates using the signature scheme with message recovery (signature verification operation) according to the standard ISO 9796-2 and with hash algorithm SHA-1. In all cases, the TSF function to verify a digital signature uses the public key which has been referenced before. In this connection, the public key is either stored in the card (e.g. within a certificate) or is loaded onto the card within a certificate by a suitable preceding operation. For the verification mechanism itself, the TSF makes use of the TSF F.RSA of the underlying IC resp. its Dedicated Support Software. For the computation of hash values the TSF F.SHA-1 of the underlying IC resp. its Dedicated Support Software is used.
F.ENC	**Encryption**
	The TSF provides a functionality to encrypt data based on asymmetric cryptography, particularly the RSA algorithm with a key length of 1024 bit. The TSF encryption function will be used for the following purpose: - Within authentication processes for the generation of authentication tokens using the encryption primitive according to the standard PKCS#1 V2.0. The TSF encryption function uses the public key which has been referenced before. In this connection, the public key is either stored in the card (e.g. within a certificate) or is loaded onto the card within a certificate by a suitable preceding operation. For the encryption mechanism itself, the TSF makes use of the TSF F.RSA of the underlying IC resp. its Dedicated Support Software.
F.DEC	**Decryption**
	The TSF provides a functionality to decrypt data based on asymmetric cryptography, particularly the RSA algorithm with a key length of 1024 bit. The TSF decryption function will be used for the following purpose: - Within authentication processes for the verification of authentication tokens using the decryption primitive according to the standard PKCS#1 V2.0. Within the Tachograph Application, the TSF decryption function is implicitly combined with the Tachograph Card´s dedicated private personalisation key (during the personalisation phase) resp. with the Tachograph Card´s dedicated and unique private signature key (during the end-usage phase of the card). Within the Card Manager Application, the TSF uses the card´s dedicated private authentication key which is intended for the proof of the card´s authenticity. Note: The pivate key is generated by the external world and loaded on the card (during initialisation resp. personalisation). It is in the responsibility of the external world to guarantee for a sufficient cryptographic strength of the private key and to handle the private key outside the card in a sufficient secure manner. Under the assumption that the external world meets the requirements on the key handling set above, the TSF decryption function works in such a manner that no information about the private key may be disclosed during the decryption operation.

	For the decryption mechanism itself, the TSF makes use of the TSF F.RSA of the underlying IC resp. its Dedicated Support Software. Furthermore, the security of the TSF is supported by the TSFs F.LOG and F.SIDE_CHAN of the IC and its Dedicated Support Software resp. the Smartcard Embedded Software.

6.2 SOF Claim for TOE Security Functions

According to Common Criteria, /CCPart1/ and /CCPart3/, all TOE security functions which are relevant for the assurance requirement AVA_SOF.1 are identified in this section.

The TOE security functions using mechanisms which can be analysed for their permutational or probabilistic properties and which contribute to AVA_SOF.1 are the following:

- The generation of random numbers by the hardware RNG within the TSF F.RNG resp. by the software RNG within the TSF F.RNG_Access can be analysed with probabilistic methods.
- The quality of the mechanism contributing to the leakage attacks of the TSF F.LOG, especially for the TSF F.HW_DEA can be analysed using probabilistic methods on power consumption of the TOE.
- The implementations of the algorithms for F.DES, F.RSA (only decryption part), F.GEN_DIGSIG and F.DEC are resistant to Simple Power Analysis (SPA), Differential Power Analysis (DPA), Differential Fault Analysis (DFA) and timing attacks. This concerns as well all security critical mechanisms of the TSF F.IA_KEY.
- The implementation of the password based authentication mechanism as used within the TSF F.IA_PWD can be analysed with permutational methods.

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

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

6.3 Assurance Measures

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

For the Smartcard Embedded Software part of the TOE (TOE-ES), the following table gives a mapping between the assurance requirements and the documents containing the relevant information for the respective requirement. All these documents concerning the TOE-ES are

provided by the developer of the TOE-ES. The table below contains only the directly related documents, references to further documentation can be taken from the mentioned documents.

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

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

As mentioned, the evaluation of the TOE will be done as composite evaluation on basis of the evaluated IC "Philips P16WX064V0C Secure 16-bit Smart Card Controller with Caernarvon Cryptographic Library on Philips Smart *XA*2 as IC Dedicated Support Software" provided by Philips Semiconductors GmbH. Therefore, for the TOE-IC the following documents will be at least provided by the IC developer:

Overview of Developer´s TOE-IC related Documents	
Class	**Documents**
Security Target	Security Target (Lite) of the IC evaluation
	Security Target (Lite) of the IC evaluation incl. Crypto Library
Evaluation Report	Evaluation Technical Report Lite (ETR Lite) of the IC evaluation
	Evaluation Technical Report Lite (ETR Lite) of the IC evaluation incl. Crypto Library
Configuration List	Configuration List for composite evaluation with ORGA
User Guidances	User Guidance for the IC
	Data Sheet for the IC
	Instruction Set for the IC
	User Guidance for the Crypto Library

7 PP Claims

Not applicable. Refer to chap. 1.3.

8 Rationale

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

8.1 Security Objectives Rationale

According to the requirements of Common Criteria, /CCPart1/ and /CCPart3/, the security objectives rationale demonstrates that the stated security objectives are traceable to all of the aspects identified in the TOE security environment and are suitable to cover them. In detail, the security objectives rationale demonstrates that the stated security objectives for the TOE and its environment are suitable to counter the identified threats to security and to cover all of the identified organisational security policies and assumptions. Vice versa, the security objective rationale shows that each security objective of the TOE and its environment at least counters one threat or is correlated to one organisational security policy or assumption.

8.1.1 Threats - Security Objectives

8.1.1.1 Threats of the TOE-IC

The threats of the TOE-IC as defined in chap. 3.3.1 are covered completely by the security objectives for the TOE-IC in chap. 4.1.1. The mapping of the threats of the TOE-IC to the relevant security objectives is done within the CC evaluation of the IC resp. within the associated Security Target.

8.1.1.2 General Threats of the TOE-ES

The general threats of the TOE-ES as defined in chap. 3.3.2 are covered completely by the general security objectives for the TOE-ES and the general security objectives for the environment of the TOE as listed in chap. 4.1.2 and 4.2.1. The mapping of the general threats of the TOE-ES to the relevant security objectives is done within the CC evaluation of the Protection Profile /PP9911/, chap. 8.2.2.

For the TOE-ES, the assumptions A.Plat-Appl, A.Process-Card and A.Check-Init for the TOE-IC within the Security Target related to the evaluation of the IC incl. its Crypto Library have been redefined suitably as security objectives O.Plat-Appl, O.Process-Card and O.Check-Init for the TOE-ES resp. for the environment of the TOE. The following supplements hold concerning these additional security objectives for the TOE-ES resp. the environment of the TOE:

O.Plat-Appl

As the Smartcard Embedded Software (TOE-ES) is designed in such a manner that the requirements from the TOE-IC guidance documents (hardware data sheet, application notes etc.) and the findings of the TOE-IC evaluation reports relevant for the Smartcard Embedded Software are met, this security objective contributes to the defense of the threats T.CLON*, T.DIS_ES2, T.T_ES, T.T_CMD, T.MOD_LOAD, T.MOD_EXE, T.MOD_SHARE and T.MOD_SOFT* (and therefore contributes indirectly to the defense of the Tachograph Card specific threats).

O.Process-Card

This security objective guarantees for secure delivery procedures for the TOE or parts of it by the TOE Manufacturer during the phases 4 to 6 of the product life-cycle with the goal to maintain confidentiality and integrity of the TOE and to prevent any possible copy, modification, retention, theft or unauthorised use. It therefore counters the threats T.CLON*, T.DIS_DEL1, T.DIS_DEL2, T.MOD_DEL1, T.MOD_DEL2 and T.T_ES (and therefore contributes indirectly to the defense of the Tachograph Card specific threats).

O.Check-Init

The security objective O.Check-Init provides the capability for the external world to check the identity of the TOE-IC by specific IC data. This security objective supplements the security objective O.Identification of the TOE-IC and therefore, the mapping to the relevant threats, assumptions or organisational policies is covered by the CC evaluation of the IC.

8.1.1.3 Tachograph Card Specific Threats

The Tachograph Card specific threats as defined in chap. 3.3.3 are covered completely by the Tachograph Card specific security objectives and some of the general security objectives for the TOE-ES as listed in chap. 4.1.3 and 4.1.2. The mapping of the Tachograph Card specific threats to the relevant security objectives is done in the following.

Threats	**Objectives**									
	O.Card_ Identification_ Data	**O.Card_ Activity_ Storage**	**O.Data_ Access**	**O.Pers_ Access**	**O.Secure_ Communications**	**O.FLAW***	**O.OPERATE***	**O.MOD_ MEMORY***	**O.Resp-Appl**	**O.Key-Function**
T.Ident_ Data	X					X	X	X	X	
T.Activity_Data		X	X			X	X	X	X	X
T.Data_ exchange					X	X	X		X	X

T.Pers_ Data				X		X	X		X	X
T.Pers_ exchange					X	X	X		X	X

In the following, for each Tachograph Card specific threat it will be explained why and how it is adressed by the security objectives listed in the table above.

T.Ident_Data

The unalterable storage of personalised identification data of the TOE (cardholder identification data, card identification data) as defined in the security objective O.Card_-Identification_Data counters directly the threat T.Ident_Data. Furthermore, the more general security objective O.MOD_MEMORY* for the TOE-ES prevents unauthorized modification of the TOE´s storage.

As the Smartcard Embedded Software (TOE-ES) treats all its user data as defined for the specific application context, i.e. according to the Tachograph Card specification, the security objective O.Resp-Appl counters the Tachograph Card specific threat T.Ident_Data.

The security objectives O.OPERATE* and O.FLAW* support the objectives O.Card_-Identification_Data, O.MOD_MEMORY* and O.Resp-Appl as they provide for the secure operation of the TOE resp. the absence of flaws, and therefore guarantee for a correct and effective realisation of the objectives O.Card_Identification_Data, O.MOD_MEMORY* and O.Resp-Appl.

T.Activity_Data

The combination of the security objectives O.Data_Access, O.Card_Activity_Storage, O.MOD_MEMORY*, O.Resp-Appl and O.Key-Function counter the threat T.Activity_Data for the following reasons:

First, the Tachograph Card specific security objective O.Data_Access limits the user data write access to authenticated vehicle units so that the modification of activity data by regular card commands can be conducted only by authenticated card interface devices.

The Tachograph Card specific security objective O.Card_Activity_Storage as well as the general security objective O.MOD_MEMORY* for the TOE-ES guarantee that a manipulation of the storage areas for activity data by other means than by regular card commands is not possible.

As the Smartcard Embedded Software (TOE-ES) treats all its user data as defined for the specific application context, i.e. according to the Tachograph Card specification, the security objective O.Resp-Appl counters the Tachograph Card specific threat T.Activity_Data.

According to the security objective O.Key-Function, key-dependent functions are implemented in the TOE-ES in a way that they are not susceptible to leakage attacks. This counters the Tachograph Card specific threat T.Activity_Data.

The security objectives O.OPERATE* and O.FLAW* support the objectives O.Data_Access, O.Card_Activity_Storage, O.MOD_MEMORY*, O.Resp-Appl and O.Key-Function as they provide for the secure operation of the TOE resp. the absence of flaws, and therefore guarantee for a correct and effective realisation of the objectives O.Data_Access, O.Card_Activity_Storage, O.MOD_MEMORY*, O.Resp-Appl and O.Key-Function.

T.Data_exchange

The Tachograph Card specific security objective O.Secure_Communications provides the support for secure communication protocols and procedures between the TOE and card interface devices. Especially, this objective supports the securing of the data transfer between the TOE and card interface devices with the goal to prevent modifications during data import and export. This counters directly the threat T.Data_exchange.

As the Smartcard Embedded Software (TOE-ES) treats all its user data as defined for the specific application context, i.e. according to the Tachograph Card specification, the security objective O.Resp-Appl counters the Tachograph Card specific threat T.Data_exchange.

According to the security objective O.Key-Function, key-dependent functions are implemented in the TOE-ES in a way that they are not susceptible to leakage attacks. This counters the Tachograph Card specific threat T.Data_exchange.

The security objectives O.OPERATE* and O.FLAW* support the objectives O.Secure_Communications, O.Resp-Appl and O.Key-Function as they provide for the secure operation of the TOE resp. the absence of flaws, and therefore guarantee for a correct and effective realisation of the objectives O.Secure_Communications, O.Resp-Appl and O.Key-Function.

T.Pers_Data

The Tachograph Card specific security objective O.Pers_Access counters the threat T.Pers_Data for the following reason: The security objective O.Pers_Access limits the personalisation data write access to authenticated personalisation units.

As the Smartcard Embedded Software (TOE-ES) treats all its user data as defined for the specific application context, i.e. according to the Tachograph Card specification, the security objective O.Resp-Appl counters the Tachograph Card specific threat T.Pers_Data concerning the personalisation phase of the TOE.

According to the security objective O.Key-Function, key-dependent functions are implemented in the TOE-ES in a way that they are not susceptible to leakage attacks. This counters the Tachograph Card specific threat T.Pers_Data concerning the personalisation phase of the TOE.

The security objectives O.OPERATE* and O.FLAW* support the objectives O.Pers_Access, O.Resp-Appl and O.Key-Function as they provide for the secure operation of the TOE resp. the absence of flaws, and therefore guarantee for a correct and effective realisation of the objectives O.Pers_Access, O.Resp-Appl and O.Key-Function.

T.Pers_exchange

The Tachograph Card specific security objective O.Secure_Communications provides the support for secure communication protocols and procedures between the TOE and card interface devices. Especially, this objective supports the securing of the data transfer between the TOE and card interface devices with the goal to prevent modifications during data import and export. This counters directly the threat T.Pers_exchange.

As the Smartcard Embedded Software (TOE-ES) treats all its user data as defined for the specific application context, i.e. according to the Tachograph Card specification, the security objective O.Resp-Appl counters the Tachograph Card specific threat T.Pers_exchange concerning the personalisation phase of the TOE.

According to the security objective O.Key-Function, key-dependent functions are implemented in the TOE-ES in a way that they are not susceptible to leakage attacks. This counters the Tachograph Card specific threat T.Pers_exchange concerning the personalisation phase of the TOE.

The security objectives O.OPERATE* and O.FLAW* support the objectives O.Secure_-Communications, O.Resp-Appl and O.Key-Function as they provide for the secure operation of the TOE resp. the absence of flaws, and therefore guarantee for a correct and effective realisation of the objectives O.Secure_Communications, O.Resp-Appl and O.Key-Function.

8.1.2 Assumptions - Security Objectives

The assumptions for the environment of the TOE as defined in chap. 3.2.1 except the assumption A.PERS are covered completely by the general security objectives for the environment of the TOE as listed in chap. 4.2.1. The mapping of the assumptions for the environment of the TOE to the relevant security objectives is done within the CC evaluation of the Protection Profile /PP9911/, chap. 8.2.3. Furthermore, the security objective O.PERS for the environment of the TOE covers directly the assumption A.PERS, as its definition shows.

8.1.3 Organisational Security Policies - Security Objectives

The security objective O.Process-Card requires the developer of the TOE to implement measures as assumed in the organisational security policy P.Process-Card, thus the security objective is covered by the mentioned organisational security policy.

Furthermore, the organisational security policy P.Design-Software obviously covers the security objectives O.Plat-Appl, O.Resp-Appl, O.Check-Init and O.Key-Function as their definition shows.

8.2 Security Requirements Rationale

According to the requirements of Common Criteria, /CCPart1/ and /CCPart3/, the security requirements rationale demonstrates that the set of security requirements of the TOE is suitable to meet and is traceable to the security objectives for the TOE and its environment. In detail, the following will be demonstrated:

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

8.2.1 Security Functional Requirements Rationale

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

8.2.1.1 Security Objectives for the TOE-IC - Security Functional Requirements

The security objectives for the TOE-IC of chap. 4.1.1 are related to the SARs and SFRs for the TOE defined in chap. 5.1.3 and 5.1.1.1. The mapping of the security objectives for the TOE-IC to the relevant SARs and SFRs is done within the CC evaluation of the IC resp. within the associated Security Target.

8.2.1.2 Security Objectives for the TOE-ES - Security Functional Requirements

The general security objectives for the TOE-ES of chap. 4.1.2 except O.Plat-Appl, O.Resp-Appl, O.Check-Init and O.Key-Function are related to the SARs and SFRs for the TOE de-

fined in chap. 5.1.3 and 5.1.1.2. The mapping of these general security objectives for the TOE-ES to the relevant SARs and SFRs is done within the CC evaluation of the Protection Profile /PP9911/, chap. 8.3.1.

For the TOE-ES, as mentioned before, the assumptions A.Plat-Appl, A.Resp-Appl, A.Check-Init and A.Key-Function of the TOE-IC within the Security Target related to the evaluation of the IC incl. its Crypto Library have been redefined suitably as security objectives for the TOE-ES. The following supplements hold concerning these additional security objectives for the TOE-ES:

O.Plat-Appl, O.Resp-Appl

The design of the TOE-ES in such a manner, that the requirements from the TOE-IC guidance documents (hardware data sheet, application notes etc.), from the findings of the TOE-IC evaluation reports relevant for the Smartcard Embedded Software and from the requirements of the Tachograph Card specification are met (O.Plat-Appl, O.Resp-Appl), is covered by the SARs for the whole TOE. In particular, the components of the class ADV with its design documentation and implementation representation (refer to chap. 5.1.3) contribute to the fulfillment of the security objectives O.Plat-Appl and O.Resp-Appl.

O.Key-Function

The design of the TOE-ES in such a manner, that the key-dependent functions are implemented in the TOE-ES in such a way that they are not susceptible to leakage attacks (O.Key-Function), is covered by the SARs for the whole TOE. In particular, the components of the class ADV with its design documentation and implementation representation and the components of the class AVA for vulnerability analysis (refer to chap. 5.1.3) contribute to the fulfillment of the security objective O.Key-Function.

O.Check-Init

The security objective O.Check-Init provides the capability for the external world to check the identity of the TOE-IC by specific IC data. This security objective supplements the security objective O.Identification of the TOE-IC and therefore, the mapping to the relevant SFRs and SARs is covered by the CC evaluation of the IC. Furthermore, this security objective is covered by the SARs for the whole TOE, in particular by the components of the class ADV with its design documentation and implementation representation (refer to chap. 5.1.3).

8.2.1.3 Tachograph Card Specific Security Objectives - Security Functional Requirements

The Tachograph Card specific security objectives as defined in chap. 4.1.3 are related to the SFRs for the TOE-ES as defined in chap. 5.1.1.2. The mapping of the Tachograph Card specific security objectives to the relevant SFRs is done in the following.

The table below gives an overview which SFRs for the TOE-ES contribute to the satisfaction of each Tachograph Card specific security objective. For clarity, the table does not identify indirect dependencies.

Security Objectives	SFRs	
	Principal SFRs	**Supporting SFRs**
O.Card_-Identification_Data	FAU_SAA.1-1 FDP_ACC.2-1 FDP_ACF.1-1 FDP_SDI.2-1	FPT_FLS.1-1 FPT_PHP.3-1 FPT_SEP.1-1 FPT_TST.1-1
O.Card_-Activity_Storage	FAU_SAA.1-1 FDP_ACC.2-1 FDP_ACF.1-1 FDP_SDI.2-1	FPT_FLS.1-1 FPT_PHP.3-1 FPT_SEP.1-1 FPT_TST.1-1
O.Data_Access	FDP_ACC.2-1 FDP_ACF.1-1 FIA_AFL.1-1 FIA_ATD.1-1 FIA_UAU.1-1 FIA_UAU.3-1 FIA_UID.1-1 FIA_USB.1-1	FPT_FLS.1-1 FPT_PHP.3-1 FPT_SEP.1-1 FPT_TST.1-1
O.Pers_Access	FDP_ACC.2-2 FDP_ACF.1-2 FIA_AFL.1-1 FIA_ATD.1-1 FIA_UAU.1-1 FIA_UAU.3-1 FIA_UID.1-1 FIA_USB.1-1	FPT_FLS.1-1 FPT_PHP.3-1 FPT_SEP.1-1 FPT_TST.1-1
O.Secure_-Communications	FAU_SAA.1-1 FCO_NRO.1-1 FCS_CKM.1-1 FCS_CKM.2-1 FCS_CKM.2-2 FCS_CKM.3-1 FCS_CKM.3-2 FCS_CKM.3-3 FCS_CKM.3-4 FCS_CKM.3-5 FCS_CKM.4-1 FCS_CKM.4-2 FCS_COP.1-1 FCS_COP.1-2 FCS_COP.1-3 FCS_COP.1-4 FCS_COP.1-5 FDP_DAU.1-1 FDP_ACC.2-1 FDP_ACF.1-1 FDP_ACC.2-2 FDP_ACF.1-2 FDP_ETC.1-1	FDP_RIP.1-1 FPT_FLS.1-1 FPT_PHP.3-1 FPT_SEP.1-1 FPT_TST.1-1

	FDP_ETC.2-1 FDP_ITC.1-1 FIA_UAU.3-1 FIA_UAU.4-1 FPR_UNO.1-1 FPR_UNO.1-2 FPT_TDC.1-1	

In the following, for each Tachograph Card specific security objective it will be explained why and how it is satisfied by the SFRs listed in the preceding table.

O.Card_Identification_Data

According to the security objective O.Card_Identification_Data, the TOE preserves identification data stored in the framework of the card´s personalisation.

Within phase 7 of the TOE´s life-cycle (end-usage phase), the access to the TOE´s data, especially to the identification data is regulated by the security function policy AC_SFP as defined in chap. 5.1.1.2.1. This SFP, accomplished by the components FDP_ACC.2-1 and FDP_ACF.1-1, denies explicitly the write access to personalised identification data.

The integrity of the stored data within the TOE, especially the integrity of the identification data is secured by the component FDP_SDI.2-1. In case of an integrity error detected by the component FAU_SAA.1-1, the TOE will indicate a violation of the TSP.

Finally, the components FPT_FLS.1-1, FPT_PHP.3-1, FPT_SEP.1-1 and FPT_TST.1-1 support the correct and secure operation of the TOE with regard to the stored identification data and their modification.

O.Card_Activity_Storage

According to the security objective O.Card_Activity_Storage, the TOE preserves user data stored in the card by vehicle units.

Within phase 7 of the TOE´s life-cycle (end-usage phase), the access to the TOE´s data, especially to the user data is regulated by the security function policy AC_SFP as defined in chap. 5.1.1.2.1. This SFP, accomplished by the components FDP_ACC.2-1 and FDP_ACF.1-1, restricts explicitly the write access to user data to authenticated vehicle units.

The integrity of the stored data within the TOE, especially the integrity of the user data written by vehicle units is secured by the component FDP_SDI.2-1. In case of an integrity error detected by the component FAU_SAA.1-1, the TOE will indicate a violation of the TSP.

Finally, the components FPT_FLS.1-1, FPT_PHP.3-1, FPT_SEP.1-1 and FPT_TST.1-1 support the correct and secure operation of the TOE with regard to the user data written by vehicle units and their modification.

O.Data_Access

The security objective O.Data_Access limits the user data write access in the TOE´s end-usage phase to authenticated vehicle units.

Within phase 7 of the TOE´s life-cycle (end-usage phase), the access to the TOE´s data, especially to the user data is regulated by the security function policy AC_SFP as defined in chap. 5.1.1.2.1. This SFP, accomplished by the components FDP_ACC.2-1 and FDP_ACF.1-1, restricts explicitly the write access to user data to authenticated vehicle units.

The component FIA_USB.1-1 together with FIA_ATD.1-1 with its definition of the user security attributes supplies a distinction between vehicle units and other card interface devices (which complies with the definitions in the security function policy AC_SFP). The components FIA_UID.1-1 and FIA_UAU.1-1 ensure that especially write access to user data is not possible without a preceding successful authentication process. If the authentication fails, the component FIA_AFL.1-1 reacts with a warning to the connected entity, and the user will be assumed as different from a vehicle unit. The component FIA_UAU.3-1 prevents the use of forged authentication data.

Finally, the components FPT_FLS.1-1, FPT_PHP.3-1, FPT_SEP.1-1 and FPT_TST.1-1 support the correct and secure operation of the TOE with regard to user data write access.

O.Pers_Access

The security objective O.Pers_Access limits the personalisation data write access in the TOE´s personalisation phase to authenticated personalisation units.

Within phase 6 of the TOE´s life-cycle (personalisation phase), the access to the TOE´s personalisation data is regulated by the security function policy AC-PERS_SFP as defined in chap. 5.1.1.2.1. This SFP, accomplished by the components FDP_ACC.2-2 and FDP_ACF.1-2, restricts explicitly the write access to personalisation data to authenticated personalisation units.

The component FIA_USB.1-1 together with FIA_ATD.1-1 with its definition of the user security attributes supplies a distinction between personalisation units and other card interface devices (which complies with the definitions in the security function policy AC-PERS_SFP). The components FIA_UID.1-1 and FIA_UAU.1-1 ensure that especially write access to personalisation data is not possible without a preceding successful authentication process. If the authentication fails, the component FIA_AFL.1-1 reacts with a warning to the connected entity, and the user will be assumed as different from a personalisation unit. The component FIA_UAU.3-1 prevents the use of forged authentication data.

Finally, the components FPT_FLS.1-1, FPT_PHP.3-1, FPT_SEP.1-1 and FPT_TST.1-1 support the correct and secure operation of the TOE with regard to personalisation data write access.

O.Secure_Communications

The security objective O.Secure_Communications contains the capability of the TOE to support secure communication protocols and procedures between the card and the card interface device when required by the application. This concerns on the one side a secured data

exchange between the TOE and the card interface device over a trusted channel, and on the other side a special data download functionality.

The component FTP_ITC.1-1 together with FDP_ETC.1-1 and FDP_ITC.1-1 offers the possibility to secure the data exchange (i.e. the data import and export) between the TOE and the card interface device by using a trusted channel assuring identification of its end points and protection of the data transfer from modification and from disclosure. Hereby, both parties are capable of verifying the received data with regard to their integrity and authenticity. The trusted channel assumes a successful preceding mutual key based authentication process between the TOE and the card interface device with agreement of session keys which is covered by the components FCS_CKM.1-1, FCS_CKM.2-1, FCS_CKM.2-2, FCS_CKM.3-1, FCS_CKM.3-2, FCS_CKM.3-3, FCS_CKM.3-4, FCS_CKM.4-1, FCS_CKM.4-2, FCS_COP.1-2 and FCS_COP.1-3 for cryptographic support. The cryptographic components FCS_CKM.3-5, FCS_COP.1-4 and FCS_COP.1-5 realise the securing of the data exchange itself. The components FPR_UNO.1-1 and FPR_UNO.1-2 guarantee for the unobservability of the install process of the trusted channel and for the unobservability of the data exchange itself which both contributes to a secure data transfer. As well, the components FIA_UAU.3-1 and FIA_UAU.4-1 support the security of the trusted channel as the TOE prevents the use of forged authentication data and as the TOE´s input for the authentication tokens and for the session keys within the preceding authentication process is used only one time. During data exchange, upon detection of an integrity error of the imported data, the TOE will indicate a violation of the TSP and will send a warning to the entity sending the data, which is realised by the component FAU_SAA.1-1.

Furthermore, the TOE offers a data download functionality with specific properties. The TOE provides the capability to generate an evidence of origin for the data downloaded to the external media, to verify this evidence of origin by the recipient of the data downloaded and to download the data to external media in such a manner that the data integrity can be verified. All these requirements are covered by FDP_ETC.2-1, FCO_NRO.1-1 and FDP_DAU.1-1. The corresponding cryptographic components for conducting the data download process with its security features are given with FCS_CKM.3-1, FCS_CKM.3-2 and FCS_COP.1-1.

For both secure communication protocols, the component FPT_TDC.1-1 ensures for a consistent interpretation of the security related data shared between the TOE and the external world.

The necessity for the usage of a secure communication protocol as well as the access to the relevant card´s keys is deposited in the security function policies AC_SFP (for the end-usage phase of the TOE´s life-cycle) resp. AC-PERS_SFP (for the personalisation phase of the TOE´s life-cycle). These policies correspond directly to the SFRs FDP_ACC.2-1 and FDP_ACF.1-1 resp. FDP_ACC.2-2 and FDP_ACF.1-2.

Finally, the components FDP_RIP.1-1, FPT_FLS.1-1, FPT_PHP.3-1, FPT_SEP.1-1 and FPT_TST.1-1 support the correct and secure operation of the TOE with regard to the secure communication protocols of the security objective O.Secure_Communications.

8.2.2 Security Functional Requirements Dependencies

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

8.2.2.1 SFRs of the TOE-IC

The dependencies under the SFRs for the TOE-IC of chap. 5.1.1 are considered in the scope of the CC evaluation of the IC resp. within the associated Security Target.

8.2.2.2 SFRs of the TOE-ES

The table below gives an overview of all SFRs defined for the TOE-ES and their dependencies. For each SFR, an information is provided about which dependency is relevant and wether and by which other SFRs of this ST the dependency is satisfied. Hereby, if there exist according to the definitions in /CCPart2/ alternative dependencies, only the chosen one is listed. Furthermore, only direct dependencies are considered.

Number	SFR		(Direct) Dependencies	Comment / Line Number
1	**FAU_SAA.1-1**	**Potential Violation Analysis**	- **FAU_GEN.1** Audit data generation	See below.
2	**FCO_NRO.1-1**	**Selective Proof of Origin**	- FIA_UID.1-1 Timing of identification	34
3	**FCS_CKM.1-1**	**Cryptographic Key Generation**	- [FCS_CKM.2-1 Cryptographic key distribution] - FCS_CKM.4-1 Cryptographic key destruction	4, 11
4	**FCS_CKM.2-1**	**Cryptographic Key Distribution**	- [FCS_CKM.1-1 Cryptographic key generation] - FCS_CKM.4-1 Cryptographic key destruction	3, 11
5	**FCS_CKM.2-2**	**Cryptographic Key Distribution**	- [FDP_ITC.1-1 Import of user data without security attributes] - FCS_CKM.4-2 Cryptographic key destruction	25, 12
6	**FCS_CKM.3-1**	**Cryptographic Key Access**	- [**FDP_ITC.1** Import of user data without security attributes or **FCS_CKM.1** Cryptographic key generation] - **FCS_CKM.4** Cryptographic key destruction	---
7	**FCS_CKM.3-2**	**Cryptographic Key Access**	- [FDP_ITC.1-1 Import of user data without security attributes] - FCS_CKM.4-2 Cryptographic key destruction	25, 12

8	**FCS_CKM.3-3**	**Cryptographic Key Access**	- [**FDP_ITC.1** Import of user data without security attributes or **FCS_CKM.1** Cryptographic key generation] - **FCS_CKM.4** Cryptographic key destruction	---
9	**FCS_CKM.3-4**	**Cryptographic Key Access**	- [FDP_ITC.1-1 Import of user data without security attributes] - FCS_CKM.4-2 Cryptographic key destruction	25, 12
10	**FCS_CKM.3-5**	**Cryptographic Key Access**	- [FCS_CKM.1-1 Cryptographic key generation] - FCS_CKM.4-1 Cryptographic key destruction	3, 11
11	**FCS_CKM.4-1**	**Cryptographic Key Destruction**	- [FCS_CKM.1-1 Cryptographic key generation]	3
12	**FCS_CKM.4-2**	**Cryptographic Key Destruction**	- [FDP_ITC.1-1 Import of user data without security attributes]	25
13	**FCS_COP.1-1**	**Cryptographic Operation**	- [**FDP_ITC.1** Import of user data without security attributes or **FCS_CKM.1** Cryptographic key generation] - **FCS_CKM.4** Cryptographic key destruction	
14	**FCS_COP.1-2**	**Cryptographic Operation**	- [FDP_ITC.1-1 Import of user data without security attributes] - FCS_CKM.4-2 Cryptographic key destruction	25, 12
15	**FCS_COP.1-3**	**Cryptographic Operation**	- [FDP_ITC.1-1 Import of user data without security attributes] - FCS_CKM.4-2 Cryptographic key destruction	25, 12
16	**FCS_COP.1-4**	**Cryptographic Operation**	- [FCS_CKM.1-1 Cryptographic key generation] - FCS_CKM.4-1 Cryptographic key destruction	3, 11
17	**FCS_COP.1-5**	**Cryptographic Operation**	- [FCS_CKM.1-1 Cryptographic key generation] - FCS_CKM.4-1 Cryptographic key destruction	3, 11
18	**FDP_ACC.2-1**	**Complete Access Control**	- FDP_ACF.1-1 Security attribute based access control	20
19	**FDP_ACC.2-2**	**Complete Access Control**	- FDP_ACF.1-2 Security attribute based access control	21

20	**FDP_ACF.1-1**	**Security Attribute Based Access Control**	- FDP_ACC.2-1 Complete access control	18 (higher hierarchical element)
21	**FDP_ACF.1-2**	**Security Attribute Based Access Control**	- FDP_ACC.2-2 Complete access control	19 (higher hierarchical element)
22	**FDP_DAU.1-1**	**Basic Data Authentication**	none	---
23	**FDP_ETC.1-1**	**Export of User Data without Security Attributes**	- [FDP_ACC.2-1 Complete access control - [FDP_ACC.2-2 Complete access control	18, 19 (higher hierarchical elements)
24	**FDP_ETC.2-1**	**Export of User Data with Security Attributes**	- [FDP_ACC.2-1 Complete access control	18 (higher hierarchical element)
25	**FDP_ITC.1-1**	**Import of User Data without Security Attributes**	- [FDP_ACC.2-1 Complete access control - [FDP_ACC.2-2 Complete access control	18, 19 (higher hierarchical elements)
26	**FDP_RIP.1-1**	**Subset Residual Information Protection**	none	---
27	**FDP_SDI.2-1**	**Stored Data Integrity Monitoring and Action**	none	---
28	**FIA_AFL.1-1**	**Authentication Failure Handling**	- FIA_UAU.1-1 Timing of authentication	31
29	**FIA_AFL.1-2**	**Authentication Failure Handling**	- FIA_UAU.1-1 Timing of authentication	31
30	**FIA_ATD.1-1**	**User Attribute Definition**	none	---
31	**FIA_UAU.1-1**	**Timing of Authentication**	- FIA_UID.1-1 Timing of identification	34
32	**FIA_UAU.3-1**	**Unforgeable Authentication**	none	---
33	**FIA_UAU.4-1**	**Single-use Authentication Mechanisms**	none	---
34	**FIA_UID.1-1**	**Timing of Identification**	none	---
35	**FIA_USB.1-1**	**User-Subject Binding**	- FIA_ATD.1-1 User attribute definition	30
36	**FMT_MOF.1**	**Management of Security Functions Behaviour**	SFR not applicable (see below).	

37	**FMT_MSA.1**	**Management of Security Attributes**	SFR not applicable (see below).	
38	**FMT_MSA.2**	**Secure Security Attributes**	SFR not applicable (see below).	
39	**FMT_MSA.3**	**Static Attribute Initialisation**	SFR not applicable (see below).	
40	**FMT_MTD.1**	**Management of TSF Data**	SFR not applicable (see below).	
41	**FMT_SMR.1**	**Security Roles**	SFR not applicable (see below).	
42	**FPR_UNO.1-1**	**Unobservability**	none	---
43	**FPR_UNO.1-2**	**Unobservability**	none	---
44	**FPT_FLS.1-1**	**Failure with Preservation of Secure State**	- ADV_SPM.1 Informal TOE security policy model	Given by assurance class (see below).
45	**FPT_PHP.3-1**	**Resistance to Physical Attack**	none	---
46	**FPT_SEP.1-1**	**TSF Domain Separation**	none	---
47	**FPT_TDC.1-1**	**Inter-TSF Basic TSF Data Consistency**	none	---
48	**FPT_TST.1-1**	**TSF Testing**	- **FPT_AMT.1** Abstract machine testing	See below.
49	**FTP_ITC.1-1**	**Inter-TSF trusted channel**	none	---

The preceding table shows that the functional component dependencies are satisfied by any functional component defined in this ST except for the components stated in the fourth row in bold characters and the FMT-components, which is explained as follows:

The **dependency of FAU_SAA.1-1 with FAU_GEN.1** (Audit Data Generation) is not applicable to the TOE. The FAU_GEN.1 component forces many security relevant events to be recorded (due to dependencies with other functional security components) and this is not achievable in a smartcard since many of these events result in card being in an insecure state where recording of the event itself could cause a security breach. The function FAU_SAA.1-1 is still be used and the specific audited events are defined in the ST independently of FAU_GEN.1.

The **dependency of FPT_TST.1-1 with FPT_AMT.1** (Abstract Machine Testing) is not relevant for a smartcard. FPT_TST.1-1 is self-consistent for the TOE (hardware and software)

and does not require the FPT_AMT.1 function. The TOE software is not tested inside the scope of FPT_TST.1-1. In its relations with external devices, the TOE is always the slave. This is why FPT_TST.1-1 is-self consistent, and FPT_AMT.1 is not applicable.

The **dependency of FCS_CKM.3-1 and FCS_CKM.3-3 with FDP_ITC.1** (Import of user data without security attributes) **or FCS_CKM.1** (Cryptographic key generation) **and FCS_CKM.4** (Cryptographic key destruction) is not applicable as the SFRs FCS_CKM.3-1 and FCS_CKM.3-3 contain the access to a private RSA-key which is stored on the card. Neither import nor generation nor key destruction are relevant for this private key.

The **dependency of FCS_COP.1-1 with FDP_ITC.1** (Import of user data without security attributes) **or FCS_CKM.1** (Cryptographic key generation) **and FCS_CKM.4** (Cryptographic key destruction) is not applicable as the SFR FCS_COP.1-1 contains the access to a public RSA-key which is stored on the card. Neither import nor generation nor key destruction are relevant for this public key.

As the TOE´s functionality as defined in the Tachograph Card specification /TachAn1B/ requires no functionality regarding the management of TOE Security Functions, security attributes, roles or TSF Data, all **FMT-components** of /PP9911/ are not applicable for the TOE.

8.2.3 Strength of Function Level Rationale

Due to the requirements in the Tachograph Card specification /TachAn1B/, main body and Appendix 10 (Tachograph Card Generic Security Target), and under consideration of the JIL interpretations /JILDigTacho/, the level for the strength of the TOE´s security functional requirements is claimed as SOF-high. The TOE is considered as a product with critical security mechanisms which only have to be defeated by attackers possessing a high level of expertise, opportunity and resources, and whereby successful attack is judged beyond normal practicability.

8.2.4 Security Assurance Requirements Rationale

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

Assurance Requirements	Name	Type
EAL4	Methodically Designed, Tested and Reviewed	Assurance Level / Class
ADO_IGS.2	Generation Log	Higher hierarchical component
ADV_IMP.2	Implementation of the TSF	Higher hierarchical component
ATE_DPT.2	Testing: Low-Level Design	Higher hierarchical component

AVA_VLA.4	Highly Resistant	Higher hierarchical component

8.2.4.1 Evaluation Assurance Level Rationale

Due to the requirements in the Tachograph Card specification /TachAn1B/, main body and Appendix 10 (Tachograph Card Generic Security Target) concerning the evaluation level of the TOE and under consideration of the JIL interpretations /JILDigTacho/, chap. 2.2 and Annex A, the assurance level for the TOE is chosen as EAL4 augmented by ADO_IGS.2, ADV_IMP.2, ATE_DPT.2 and AVA_VLA.4. Hereby, all assurance components will be used as defined in /CCPart3/ and /CEMPart2/ with the refinements as noted in the JIL interpretations /JILDigTacho/, Annex A.3 and A.5 (refer to chap. 5.1.4 of this ST). The choice of the CC assurance components for the TOE incl. the chosen augmentations and refinements provides an assurance level comparable to the evaluation level ITSEC E3 high.

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

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

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

8.2.4.2 Assurance Augmentations Rationale

The following section gives reason for the choice of the assurance components augmenting the evaluation assurance class EAL4.

Apriori, the assurance components ADO_IGS.2, ADV_IMP.2, ATE_DPT.2 and AVA_VLA.4 are chosen with respect to the requirements in the JIL interpretations /JILDigTacho/, Annex A.3 and A.5 in order to achieve a CC assurance level comparable to ITSEC E3 high as required in the Tachograph Card specification /TachAn1B/, main body and Appendix 10 (Tachograph Card Generic Security Target).

In detail, the following deliberations are of interest:

ADO_IGS.2 Generation Log

Installation, generation and start-up procedures are useful for ensuring that the TOE has been installed, generated and started up in a secure manner as intended by the developer. The requirements for installation, generation and start-up call for a secure transition from the TOE's implementation representation being under configuration control to its initial operation in the user environment.

The assurance component ADO_IGS.2 is a higher hierarchical component to EAL4, which only requires ADO_IGS.1 „Installation, generation, and start-up procedures".

The augmentation by ADO_IGS.2 with the interpretation and refinement given in the JIL interpretations /JILDigTacho/, Annex A.3 Note 2 (refer to chap. 5.1.4 of this ST) is required with regard to the evaluation level ITSEC E3 high. It is important for the TOE and its assurance that the evaluator does not evaluate only the steps necessary for a secure installation, generation and start-up of the TOE. Moreover, the procedures shall be capable of creating a log containing the generation options used to generate the TOE in such a way that it is possible to determine exactly how and when the TOE was generated.

ADV_IMP.2 Implementation of the TSF

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

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

The augmentation by ADV_IMP.2 with the interpretation given in the JIL interpretations /JILDigTacho/, Annex A.3 Note 5 is required with regard to the evaluation level ITSEC E3 high. It is important for the TOE and its assurance that the evaluator evaluates the implementation representation of the *entire* TSF to determine that the SFRs as defined in the ST are addressed by the representation of the TSF and that the implementation representation is an accurate and complete instantiation of the TOE´s SFRs. This provides a direct correspondence between the TOE´s SFRs and the implementation representation, in addition to the pairwise correspondences required by the ADV_RCR family.

ATE_DPT.2 Testing: Low-Level Design

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

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

The augmentation by ATE_DPT.2 with the interpretation given in the JIL interpretations /JILDigTacho/, Annex A.3 Note 8 is required with regard to the evaluation level ITSEC E3

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

AVA_VLA.4 Highly Resistant

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

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

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

The augmentation by AVA_VLA.4 with the interpretation given in the JIL interpretations /JILDigTacho/, Annex A.3 Note 9 and 11 is required with regard to the evaluation level IT-SEC E3 high. For AVA_VLA.4, a systematical vulnerability analysis is performed by the developer to ascertain the presence of security vulnerabilities, and to confirm that they cannot be exploited in the intended environment for the TOE. Hereby, the analysis shall provide a justification that the analysis completely addresses the TOE deliverables. The evaluator performs independent penetration testing, supported by the evaluator’s independent vulnerability analysis, to determine that the TOE is resistant to penetration attacks performed by attackers possessing a high attack potential.

8.2.5 Security Assurance Requirements Dependencies

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

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

ADO_IGS.2 has a dependency with AGD_ADM.1“ Administrator Guidance”. This dependency is satisfied by EAL4.

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

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

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

8.2.6 Security Requirements – Mutual Support and Internal Consistency

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

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

- The assurance class EAL4 is an established set of mutually supportive and internally consistent assurance requirements.
- The dependency analysis for the additional assurance components in chap. 8.2.5 shows that the assurance requirements are mutually supportive and internally consistent as all (additional) dependencies are satisfied and no inconsistency appears.
- The dependency analysis in chap. 8.2.2 for the security functional requirements of the TOE-IC and the TOE-ES shows that the basis for mutual support and internal consistency between all defined functional requirements is satisfied. All dependencies between the chosen functional components are analysed, and non-dissolved dependencies are appropriately explained.

  The mutual support and internal consistency of the functional requirements is shown for the TOE-IC relevant SFRs in the scope of the CC evaluation of the TOE-IC resp. in the correlated ST and for the TOE-ES relevant SFRs in chap. 8.2.1.2 and 8.2.1.3 within the mapping of the security objectives to the SFRs.

  Concerning the SFRs of the TOE-ES, the SFRs drawn from /PP9911/ build as shown in the rationale of the protection profile a mutually supportive and internally consistent whole. The additional SFRs resulting from the Tachograph Card specification /TachAn1B/, Appendix 10 (Tachograph Card Generic Security Target) and the JIL interpretations /JILDigTacho/, Annex B suitably supplement the SFRs of the protection profile and do not lead to any inconsistency or any weakness as can be seen from the deliberations in chap. 8.2.1.2 and 8.2.1.3.
- All operations conducted on the CC functional components lead to a consistent and meaningful whole.

  For the TOE-IC relevant SFRs the evidence is done within the scope of the CC evaluation of the TOE-IC resp. in the correlated ST.

For the TOE-ES relevant SFRs the following deliberations are important. First, all operations on the chosen SFRs are done under consideration of the requirements in the Tachograph Card specification /TachAn1B/, Appendix 10 (Tachograph Card Generic Security Target) and the JIL interpretations /JILDigTacho/, Annex B. Furthermore, the following holds:

- Assignment and selection operations:

  All assignment and selection operations are conducted in such a way that they do not contradict each other and build an internally consistent security system which reflects the security requirements of the Tachograph Card system as specified in the Tachograph Card specification /TachAn1B/. Especially, this concerns the defined access control policies AC_SFP and AC-PERS_SFP.

- Iteration operations:

  The iteration of the functional components FDP_ACC.2 and FDP_ACF.1 are necessary to differentiate between the personalisation phase and the end-usage phase of the TOE and their phase-specific access control functionality.

  The iteration of the functional components for cryptographic support, FCS_CKM.2, FCS_CKM.3, FCS_CKM.4 and FCS_COP.1, are necessary to differentiate between the different cryptographic algorithms and mechanisms of the TOE.

  The iteration of the functional component FIA_AFL.1 is necessary to differentiate between the two different authentication mechanisms of the TOE.

- Refinement operations:

  Not conducted.

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

8.3 TOE Summary Specification Rationale

According to the requirements of Common Criteria, /CCPart1/ and /CCPart3/, the TOE summary specification rationale demonstrates that the TOE security functions (TSFs) and assurance measures are suitable to meet the TOE security requirements. In detail, the following will be demonstrated:

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

8.3.1 Security Functions Rationale

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

8.3.1.1 Security Functional Requirements for the TOE-IC – TOE Security Functions

The SFRs for the TOE-IC of chap. 5.1.1.1 are related to the TSFs of the TOE-IC defined in chap. 6.1.1. The mapping of the SFRs for the TOE-IC to the relevant TSFs is done within the CC evaluation of the IC resp. within the associated Security Target.

8.3.1.2 Security Functional Requirements for the TOE-ES – TOE Security Functions

The SFRs for the TOE-ES of chap. 5.1.1.2 are related to the TSFs of the TOE-ES defined in chap. 6.1.2. The mapping of the SFRs for the TOE-ES to the relevant TSFs is done in the following.

The tables below give an overview of which TSFs of the TOE-ES contribute to the satisfaction of the SFRs for the TOE-ES.

TOE Security Function	SFR											
	FAU_SAA.1-1	FCO_NRO.1-1	FCS_CKM.1-1	FCS_CKM.2-1	FCS_CKM.2-2	FCS_CKM.3-1	FCS_CKM.3-2	FCS_CKM.3-3	FCS_CKM.3-4	FCS_CKM.3-5	FCS_CKM.4-1	FCS_CKM.4-2
F.ACS		X		(X)		X	X	X	X	X		
F.IA_KEY			X	X	X							
F.IA_-PWD	X											
F.DATA_-INT	X											
F.EX_-CONF	X											
F.EX_INT	X											
F.RIP											X	X
F.FAIL_-PROT	X											
F.SIDE_-CHAN												
F.SELF-TEST	X											
F.GEN_-SES			X	X								
F.GEN_-DIGSIG		X		X								
F.VER_-DIGSIG		X		X	X							
F.ENC				X								
F.DEC				X								

TOE Security Function	SFR											
	FCS_COP.1-1	FCS_COP.1-2	FCS_COP.1-3	FCS_COP.1-4	FCS_COP.1-5	FDP_ACC.2-1	FDP_ACC.2-2	FDP_ACF.1-1	FDP_ACF.1-2	FDP_DAU.1-1	FDP_ETC.1-1	FDP_ETC.1-2
F.ACS	(X)	(X)	(X)	(X)	(X)	X	X	X	X	X		X
F.IA_KEY												
F.IA_-PWD												

F.DATA_-INT												
F.EX_-CONF											X	
F.EX_INT											X	
F.RIP												
F.FAIL_-PROT												
F.SIDE_-CHAN												
F.SELF-TEST												
F.GEN_-SES												
F.GEN_-DIGSIG	X	X								X		X
F.VER_-DIGSIG	X	X								X		
F.ENC												
F.DEC												

TOE Security Function	**SFR**											
	FDP_ITC.1-1	**FDP_RIP.1-1**	**FDP_SDI.2-1**	**FIA_AFL.1-1**	**FIA_AFL.1-2**	**FIA_ATD.1-1**	**FIA_UAU.1-1**	**FIA_UAU.3-1**	**FIA_UAU.4-1**	**FIA_UID.1-1**	**FIA_USB.1-1**	**FMT**
F.ACS						X	X			X	X	Not applicable.
F.IA_KEY				X				X	X			
F.IA_-PWD					X			X				
F.DATA_-INT			X									
F.EX_-CONF	X											
F.EX_INT	X											
F.RIP		X										

F.FAIL_-PROT												
F.SIDE_-CHAN												
F.SELF-TEST												
F.GEN_-SES												
F.GEN_-DIGSIG												
F.VER_-DIGSIG												
F.ENC												
F.DEC												

TOE Security Function	**SFR**										
	FPR_UNO.1-1	**FPR_UNO.1-2**	**FPT_FLS.1-1**	**FPT_PHP.3-1**	**FPT_SEP.1-1**	**FPT_TDC.1-1**	**FPT_TST.1-1**	**FPT_ITC.1-1**			
F.ACS					X						
F.IA_KEY	X					X		X			
F.IA_-PWD						X					
F.DATA_-INT											
F.EX_-CONF		X				X		X			
F.EX_INT						X		X			
F.RIP											
F.FAIL_-PROT			X								
F.SIDE_-CHAN				X							
F.SELF-TEST							X				
F.GEN_-SES	X					X					
F.GEN_-DIGSIG						X					

F.VER_-DIGSIG						X					
F.ENC						X					
F.DEC						X					

Note: The “x” resp. “(x)” means that the TSF realises resp. supports the functionality required by the respective SFRs.

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

8.3.2 Assurance Measures Rationale

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

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

8.3.3 TOE Security Functions – Mutual Support and Internal Consistency

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

The deliberations in chap. 8.3.1 support this result. Additionally, for the TSFs of the TOE-IC as defined in chap. 6.1.1 such analysis is done in the scope of the IC evaluation resp. within the correlated ST.

8.3.4 Strength of Functions

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

Reference

I Bibliography

/CCPart1/
Title: Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model
Identification: CCIMB-99-031
Version: Version 2.1
Date: August 1999
Author: CC Project Sponsoring Organisations CSE, SCSSI, BSI, NLNCSA, CESG, NIST, NSA

/CCPart2/
Title: Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements
Identification: CCIMB-99-032
Version: Version 2.1
Date: August 1999
Author: CC Project Sponsoring Organisations CSE, SCSSI, BSI, NLNCSA, CESG, NIST, NSA

/CCPart3/
Title: Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements
Identification: CCIMB-99-032
Version: Version 2.1
Date: August 1999
Author: CC Project Sponsoring Organisations CSE, SCSSI, BSI, NLNCSA, CESG, NIST, NSA

/CEMPart1/
Title: Common Methodology for Information Technology Security Evaluation, Part 1: Introduction and General Model
Identification: CEM99/045
Version: Draft 0.6
Date: Jan. 1997
Author: CC Project Sponsoring Organisations CSE, SCSSI, BSI, NLNCSA, CESG, NIST, NSA

/CEMPart2/
Title: Common Methodology for Information Technology Security Evaluation, Part 2: Evaluation Methodology
Identification: CEM-97/017
Version: V1.0
Date: Aug. 1999
Author: CC Project Sponsoring Organisations CSE, SCSSI, BSI, NLNCSA, CESG, NIST, NSA

/JILDigTacho/
Title: JIL Security Evaluation and Certification of Digital Tachographs
Version: Version 1.12
Date: June 2003
Author: JIL Working Group (BSI, CES, DCSSI, NLNCSA)

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

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

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

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

/ST-ICPhilips/
Title: Security Target - Evaluation of the Philips P16WX064V0C Secure 16-bit Smart Card Controller
Identification: BSI-DSZ-CC-0203
Version: Version 1.1
Date: Jan. 24th 2003
Publisher: Philips Semiconductors GmbH, Business Unit Identification

/TachAn1B/
Title: Annex 1B of Commission Regulation (EC) No.1360/2002 on recording equipment in road transport: Requirements for Construction, Testing, Installation and Inspection (in: Official Journal of the European Communities, L 207 / 1 ff.)
Date: 05.08.2002
Publisher: Commission of the European Communities

/ISO9796-2/
Title: Information Technology – Security Techniques – Digital Signature Schemes Giving Message Recovery – Part 2: Mechanisms Using a Hash Function
Identification: ISO/IEC 9796-2
Version: First Edition
Date: 1997
Publisher: ISO / IEC

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

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

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

Publisher: International Organization for Standardization/International Electrotechnical Commission

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

/SHA-1/
Title: Secure Hash Standard
Identification: FIPS Publication 180-1
Date: April 1995
Publisher: National Institute of Standards and Technology (NIST)

/TDES/
Title: Data Encryption Standard
Identification: FIPS Publication 46-3
Date: Draft 1999
Publisher: National Institute of Standards and Technology (NIST)

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

/PKCS1/
Title: RSA Encryption Standard
Identification: PKCS#1
Version: Version 2.0
Date: Oct. 1998
Publisher: RSA Laboratories

/JCAPI21/
Title: Java Card 2.1.1 Application Programming Interface
Date: May 2000
Publisher: Sun Microsystems Inc.

/JCRE21/
Title: Java Card 2.1.1 Runtime Environment (JCRE) Specification
Date: May 2000
Publisher: Sun Microsystems Inc.

/JCVM21/
Title: Java Card 2.1.1 Virtual Machine Specification
Date: May 2000
Publisher: Sun Microsystems Inc.

II Summary of abbreviations

A.x	Assumption
AC	Access Condition
AID	Application Identifier
ALW	Always
AM	Access Mode
JCAPI	Java Card Application Programming Interface
JCRE	Java Card Runtime Environment
JCVM	Java Card Virtual Machine
AR	Access Rule
AS	Application Software
ATR	Answer To Reset
AUT	Key Based Authentication
BS	Basic Software
CC	Common Criteria
DES	Data Encryption Standard
DPA	Differential Power Analysis
DF	Dedicated File
DFA	Differential Fault Analysis
EAL	Evaluation Assurance Level
EF	Elementary File
ES	Embedded Software
IC	Integrated Circuit
IFD	Interface Device
ITSEC	Information Technology Security Evaluation Criteria
JIL	Joint Interpretation Library
MAC	Message Authentication Code
MF	Master File
O.x	Security Objective
OS	Operating System
P.x	Organisational Security Policy
PIN	Personal Identification Number
PP	Protection Profile
PW	Password
PWD	Password Based Authentication
RSA	Rivest-Shamir-Adleman Algorithm
SAR	Security Assurance Requirement
SFP	Security Function Policy
SFR	Security Functional Requirement
SM	Secure Messaging
SOF	Strength of Functions
SPA	Simple Power Analysis
SPM	TOE Security Policy Model
SSC	Send Sequence Counter

ST	Security Target
ST-Lite	Security Target Lite
T.x	Threat
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy
VU	Vehicle Unit

III Glossary

For explanation of technical terms refer to the following documents:

/PP9911/, Annex A

/BSI-PP-0002/, Chap. 8.7

/ST-ICPhilips/, Glossary

/TachAn1B/, Annex 1B Main Body, Chap. I Definitions

/TachAn1B/, Appendix 10, Tachograph Card Generic Security Target, Chap. 2

Appendix

Definition of components FCS_RND.2 and FPT_TST.2

To define the IT security functional requirements of the TOE-IC an additional family FCS_RND (generation of random numbers) of the class FCS (cryptographic support) and an additional component of the family FPT_TST (TSF self test) are defined.

The family FCS_RND describes the functional requirements for random number generation used for cryptographic purposes. The definition of this family was already begun in the PP /BSI-PP-0002/ and its augmentations with FCS_RND.1; a new component FCS_RND.2 is added here. For ease of reading, the definition of the whole family will be repeated here.

The family FPT_TST describes the functional requirements for TSF self tests. A new component FPT_TST.2 is added to the family. The definition of the component FPT_TST.2 has already been given in the augmentation paper to the PP /BSI-PP-0002/. For ease of reading, the definition of this component is repeated here. For the definition of the family FPT_TST and of the component FPT_TST.1 see /CCPart2/.

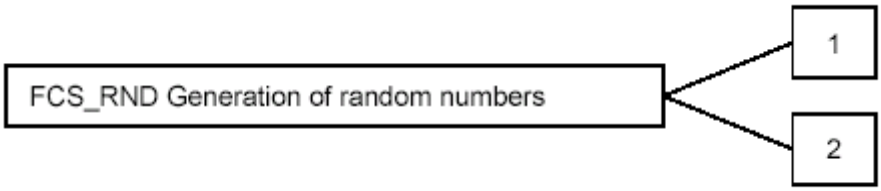
I. Generation of random numbers (FCS_RND)

Family Behaviour

This family describes the functional requirements for random number generation used for cryptographic purposes.

In order to ensure that a random number generator can be employed for different cryptographic purposes, the random number generation must assure that the generated random numbers posses certain properties. Typical properties include assurance that a given quality metric (e.g. minimum entropy) is achieved or that an implementation meets a given standard.

Component levelling



FCS_RND.1 Quality Metric for Random Numbers requires that random numbers meet a defined quality metric.

FCS_RND.2 Random Number Generation requires that random number generation is performed based on an assigned standard.

Management: FCS_RND.1, FCS_RND.2

There are no management activities foreseen.

Audit: FCS_RND.1, FCS_RND.2

There are no actions defined to be auditable.

FCS_RND.1 Quality Metric for Random Numbers

Hierarchical to: No other components

FCS_RND.1.1 The TSF shall provide a mechanism to generate random numbers that meet [assignment: *a defined quality metric*].

Dependencies: No dependencies.

FCS_RND.2 Random Number Generation

Hierarchical to: No other components

FCS_RND.2.1 The TSF shall provide a mechanism to generate random numbers that meet the following: [assignment: *list of standards*].

Dependencies: No dependencies.

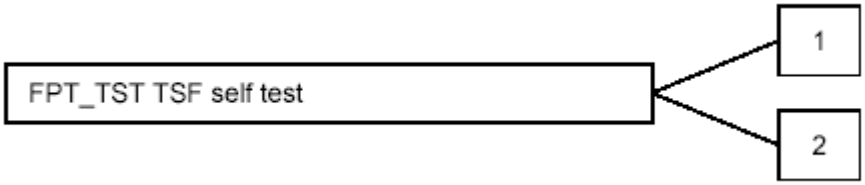
II. TST self test (FPT_TST)

To define the IT security functional requirements of the TOE an additional component FPT_TST.2 of the family FPT_TST (TSF self test) is defined here. The family FPT_TST is taken from /CCPart2/. The new component FPT_TST.2 has already been defined in the augmentation paper of the PP /BSI-PP-0002/. Its definition is repeated here for ease of reading.

Family behaviour

The behaviour of the family FPT_TST remains unchanged if compared to its definition within /CCPart2/.

Component levelling



FPT_TST.1 TSF testing, provides the ability to test the TSF's correct operation. These tests may be performed at start-up, periodically, at the request of the authorised user, or when

other conditions are met. It also provides the ability to verify the integrity of TSF data and executable code.

FPT_TST.2 Subset TOE security testing, provides the ability to test the correct operation of particular security functions or mechanisms. These tests may be performed at startup, periodically, at the request of the authorised user, or when other conditions are met. It also provides the ability to verify the integrity of TSF data and executable code.

Management: FPT_TST.1, FPT_TST.2

The management activities foreseen for FPT_TST.1 remain unchanged, i.e. as specified within /CCPart2/. These management activities may also be considered for FPT_TST.2. There are no other management activities foreseen for FPT_TST.2.

Audit: FPT_TST.1, FPT_TST.2

The actions defined to be auditable for FPT_TST.1 remain unchanged, i.e. as specified within /CCPart2/. The same action may also be considered for FPT_TST.2. There are no other auditable action defined for FPT_TST.2.

FPT_TST.2 Subset TOE security testing

Hierarchical to: No other components.

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

Dependencies: FPT_AMT.1 Abstract machine testing