Security Target Lite

for ORION

(Microcontroller ORION_CB_03 and ORION_DB_03)

Reference :	ORION_ST_Lite
Revision :	1.61
Date :	September 12, 2025
Distribution :	PUBLIC

Revision History

Revision	Date	Comments
1.0	07/04/2017	Creation
1.1	19/04/2017	Update of guidance revision number.
1.2	23/02/2018	Update of guidance revision number.
1.3	04/03/2019	Update of guidance revision number.
1.4	10/07/2020	Update rev D
1.5	30/04/2021	Update of guidance revision number
1.51	10/05/2021	Update of guidance revision number
1.52	20/04/2023	Update of security guidance revision number and company name and 1.2.8 Forms of Delivery
1.53	19/04/2024	Update of security guidances revision number
1.6	18/08/2025	Update CC2002
1.61	12/09/2025	Update of security guidance revision number and SF_CRYPTO is removed for consistency with the absence of a cryptographic services package in this TOE

Contents

List of Tables

List of Figures

References

[1] Common Criteria for Information Technology Security Evaluation Part 1: Introduction and General Model; November 2022, CC2022 Revision 1

[2] Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Components; November 2022, CC2022 Revision 1

[3] Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Components; November 2022, CC2022 Revision 1

[4] Common Criteria for Information Technology Security Evaluation, Part 4: Framework for the specification of evaluation methods and activities; November 2022, CC2022 Revision 1,

[5] Common Criteria for Information Technology Security Evaluation, Part 5: Pre-defined packages of Security Requirements; November 2022, CC2022 Revision 1.

[5.1] Errata and Interpretation for CC:2022 (Release 1) and CEM:2022 (Release 1), Version 1.1, 2024-07-22.

[6] Common Methodology for Information Technology Security Evaluation (CEM), Evaluation Methodology; CC2022, Revision 1

[7] Security IC Platform Protection Profile; January 2014, Version 1.0, BSI-PP-0084

[8] Joint Interpretation Library: Application of Attack Potential to Smartcards, February 2024, Version 3.2.1

[9] Supporting Document, Mandatory Technical Document: The Application of CC to Integrated Circuits, March 2009, Version 3.0, Revision 1, CCDB-2009-03-002

[10] Supporting Document Guidance: Smartcard Evaluation, February 2010, Version 2.0, CCDB-2010-03-001

[11] Supporting Document Guidance Security Architecture requirements (ADV_ARC) for smart cards and similar devices, April 2012, Version 2.0, CCDB-2012-04-003

[12] Supporting Document Mandatory Technical Document: Application of Attack Potential to Smartcards April 2012, Version 2.8, CCDB-2012-04-002

[13] Supporting Document: Composite product evaluation for Smart Cards and similar devices, April 2024, Version 1.6

[14] Joint Interpretation Library: Minimum Site Security Requirements , Version 3.1, 2023

[15] **Orion - User Manual, version 1.2, April 11, 2017**.

[16] **Secure 32 bits CPU Embedded Application Binary Interface (EABI), version 0.6, March 2013**.

[17] **Secure 32 bits CPU Instruction Set Architecture (ISA), version 1.1b, 29 January 2019**.

[18] **Security Guidance, version 0.30, September 11, 2025**.

[19] PP0084: Interpretations, reference: PP0084, version 03, 01/06/2016 from ANSSI.

[20] Smartcard Integrated Circuit Platform Augmentations, version 1.00, March 8, 2002, developed by Atmel, Hitachi Europe, Infineon Technologies, and Philips Semiconductors.

[21] Supporting Document Guidance ETR template for composite evaluation of Smart Cards and similar devices, September 2007, Version 1.0, Revision 1, CCDB-2007-09-002.

[22] **Orion Loader – User Manual, version 1.7, January 16, 2017**.

[23] **Guidance – Secure Delivery, version 1.0, December 12, 2016**.

[24] **Orion – Assembly Instructions, version 0.2, November 13, 2015**.

[25] Security Target for ORION (Microcontroller ORION_CB_03 and ORION_DB_03), version 3.22, September 12, 2025.

[26] ISO/IEC 7816-3. Identification cards — integrated circuit cards. Part 3: Cards with contacts Electrical interface and transmission protocols.

Acronyms

CC	Common Criteria
EAL	Evaluation Assurance Level
EC	Elliptic Curve
ECC	Error Correcting Code
DRNG	Digital Random Number Generator
IC	Integrated circuit
Loader	Loader to load SW in the IC
MPU	Memory Protection Unit
PEOS	Product Engineering Operating System
PKI	Public Key Infrastructure
PP	Protection Profile
PTRNG	Pseudo True Random Number Generator
RNG	Random Number Generator
SAR	Security Assurance Requirement
SF	Security Function
SFR	Security Functional Requirement
ST	Security Target
SWP	Single Wire Protocol
TOE	Target Of Evaluation
TSF	TOE Security Functionality

1 ST Introduction

This chapter ST introduction contains the following sections:

ST Reference (1.1)

TOE Overview (1.2)

1.1 ST Reference

Title:	Security Target Lite for ORION
Reference:	ORION_ST_Lite
Version Number:	1.61
Date:	12/09/2025
Provided by:	THALES DIS France SAS, Arteparc – Bât D, Route de la côte d'Azur, 13590 Meyreuil, France
Evaluator:	CEA LETI, Grenoble
Evaluation Scheme:	France - Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI)

1.2 TOE Overview

1.2.1 TOE Identification

The Target of Evaluation (TOE) is a Secure Microcontroller (Secure IC) with a Dedicated Support Software.

The TOE is identified as below:

TOE reference	**ORION_TOE_v5**	
Commercial Name	**ORION, HYDRA, ORION 1M, ORION 1M2, ORION 1M4, ORIONM2M**	
Product Name	**ORION_CB_03**	**ORION_DB_03**
Hardware Revision	C	D
Platform ROM Firmware Revision	B	B
Platform FLASH Firmware Revision • *BIOS* • *Loader*	03 *Version 2.0* *Version 2.0*	
Crypto Support Library	None	

Guidance	• User Manual [15] • Secure 32 bits CPU Embedded Application Binary Interface [16] • Secure 32 bits CPU Instruction Set Architecture [17] • Security Guidance [18] • Orion Loader – User Manual [22] • Guidance – Secure Delivery [23] • Orion – Assembly Instructions [24]

The security needs for the TOE can be summarized as being able to:

- Maintain the integrity and the confidentiality of the sensitive content of the TOE memories as required by the end application(s)
- Maintain the correct execution of the software residing on the TOE.

1.2.2 TOE Main security features

The main security features of the ORION integrated circuit are:

- the active shield;
- the security sensors;
- memories and bus encryption mechanisms;
- data integrity mechanisms;
- the random number generator (PTRNG).
- the HW Cryptographic Accelerator (providing acceleration instructions to support implementation of cryptographic algorithms TDES and AES);
- the PKI Engine (providing acceleration instructions to support implementation of cryptographic algorithms RSA, ECDSA and ECDH).

Note that the secure crypto lib is not part of the TOE.

1.2.3 TOE Definition

The TOE comprises:

- Hardware Secure Chip – see description below
- Associated IC Dedicated Support Software
  - Bootloader to start the product
  - Loader to load SW in the IC by the customer
- TOE Guidance Documentation

More detail is given at the end of this section.

Figure 1 provides an overview of the ORION product.

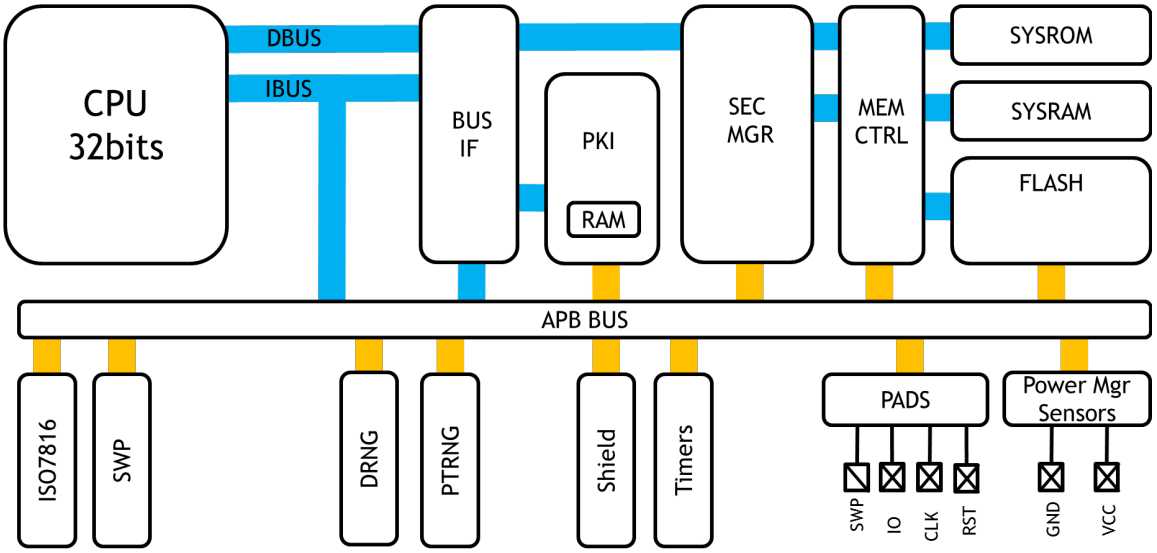


Figure 1 : Block Diagram of the TOE

- Operating condition

Voltage	1.62 Volt < VCC < 5.5 Volt
Temperature rev D	-40°C up to +105°C
Temperature rev C	-25°C up to +85°C

Table 1 : ORION operating conditions

- CPU: CPU Secure 32-bit
- Memories

Memory
ROM
RAM
NVM (Flash)

Table 2 : Memory Size

- MPU
  - Access rights control, with interruption request if bad access
- Interfaces
  - Compliant with:

ISO7816-3 [26] (contact)
ETSI TS 102 613 (SWP) (contactless)

Table 3 : ORION Interface

  - ISO7816-3 and SWP can communicate in the same time
- Crypto-coprocessors
  - PKI Engine for RSA, ECDSA and ECDH
  - HW Cryptographic Accelerator for DES/TDES and AES
  - 16/32 bits CRC
  - 2 Random Number Generator: one is designed to be FIPS140-2 compliant (DRNG) and second one is AIS31-PTG.2 compliant (PTRNG)

- Internal clocks and power consumption
  - Standby mode for power saving
- Resets
  - Internal Power on reset
  - Only software and alarm can generate a system reset
- Environment Control
  - Active shield protection
  - Environment Sensors Monitoring
- Data Integrity and redundancy mechanism
- Timers
  - Two system timers
  - Two external clocks timers (ISO7816-3 and ETSI TS 102 613)
- ESD Robustness

The ROM of the TOE contain a Dedicated Software allowing to configure the product and start the product (boot/start-up) – the Bootloader –, and including a Dedicated Software which provides a very reduced set of commands for final test (the Product Engineering Operating System for final test, called "PEOS"), not intended for the Security IC Embedded Software usage, and not available in User configuration. As it is not available in User Mode, the PEOS is not included in the TOE.

The System ROM and NVM of the TOE contain a Dedicated Support Software called Loader, enabling to securely and efficiently download the Security IC Embedded Software (ES) into the NVM. It also allows the evaluator to load software into the TOE for test purpose. The Loader is available in User configuration but is erased after usage.

The cryptographic library is out of the scope of TOE at this stage.

1.2.4 TOE Life Cycle

This Security Target is fully conformant to the claimed (BSI-PP-0084), the full details of the Security IC life cycle is shown in the PP. This Security Target gives a short summary of the information given in the PP. Information is also given within this Security Target to expand on the applicable phases of the life cycle of the TOE.

Open samples are provided to customer with Loader in.

The complex development and manufacturing processes of a Composite Product can be separated into seven distinct phases. The phases 2 and 3 of the Composite Product life cycle cover the IC development and production:

- IC Development (Phase 2):

  - IC design,

  - IC Dedicated Software development,

- the IC Manufacturing (Phase 3):

  - Integration and photomask fabrication,

  - IC production,

  - IC testing,

  - Initialisation, and

  - Pre-personalisation

In addition, five important stages have to be considered in the Composite Product life cycle:

  - Security IC Embedded Software Development (Phase 1),

  - the Composite Product IC packaging (Phase 4),

  - the Composite Product finishing process, preparation and shipping to the personalisation line for the Composite Product (Composite Product Integration Phase 5),

  - the Composite Product personalisation and testing stage where the user data of the Composite TOE is loaded into the Security IC's memory (Personalisation Phase 6),

  - the Composite Product usage by its issuers and consumers (Operational Usage Phase 7) which may include loading and other management of applications in the field.

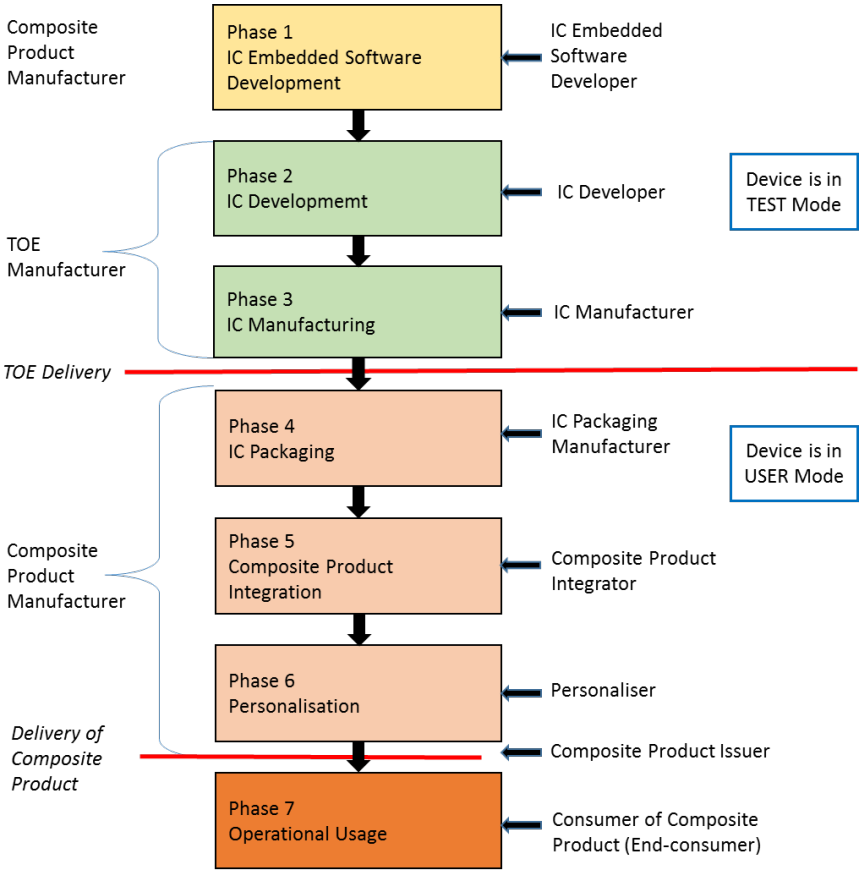


Figure 2 : ORION Life Cycle

The Security IC Embedded Software is developed outside the TOE development in Phase 1. The TOE is developed in Phase 2 and produced in Phase 3. The TOE is delivered in form of wafers or sawn wafers (dice).

In the following the term “TOE Delivery” (refer to Figure 2) is uniquely used to indicate that after Phase 3 (or before Phase 4) the TOE is delivered in form of wafers or sawn wafers (dice).

In the following the term “TOE Manufacturer” (refer to Figure 2) which includes the following roles:

- the IC Developer (Phase 2) and

- the IC Manufacturer (Phase 3)

Hence the “TOE Manufacturer” comprise all roles beginning with Phase 2 and before “TOE Delivery”. Starting with “TOE Delivery” another party takes over the control of the TOE.

In the following, the term “Composite Product Manufacturer” which includes all roles (outside TOE development and manufacturing) except the End-consumer as user of the Composite Product (refer to Figure 2) which are the following:

- Security IC Embedded Software development (Phase 1)
- the IC Packaging Manufacturer (Phase 4)
- the Composite Product Manufacturer (Phase 5) and
- the Personalizer (Phase 6).

During Phase 2 and Phase 3, the following sites are involved:

Function	**Company**
Phase 2: IC Development	
IC Design	**THALES DIS France SAS** Arteparc – Bâtiment D, Route de la côte d’Azur 13590 Meyreuil FRANCE
IC dedicated software development & test	**THALES DIS France SAS** Arteparc – Bâtiment D, Route de la côte d’Azur 13590 Meyreuil FRANCE
Validation	**MU-Electronics** 49 rue Jabal Tazekka, 1er étage, Agdal, 10000 Rabat MOROCCO
Loader	**THALES DIS France SAS** La Vigie, Avenue du Jujubier, Z.I. Athelia IV 13705 La Ciotat Cedex FRANCE

Phase 3: IC Manufacturing	
Wafer fab / Warehouse	**UMC Fab 12i** No.3, Pasir Ris Drive 12, Singapore 519528 SINGAPORE
Data Prep & Mask Shop	**PDMC** **Masks Manufacturing (1A)** 1stFloor, N°2, Li-Hsin Rd, Science Park, Hsinchu 30078 TAÏWAN **Masks Manufacturing (1B)** N°13, Tongshan Rd, Daya District, Taichung 42879 TAÏWAN **Masks Manufacturing (1D)** (contain only the server room of the whole PDMC company) N°6, Li-Hsin 7th Rd, Science Park, Hsinchu 30078 TAÏWAN
Testing	**UTAC** 5 Serangoon North Avenue 5, Singapore 554916 SINGAPORE

Table 4 : List of sites

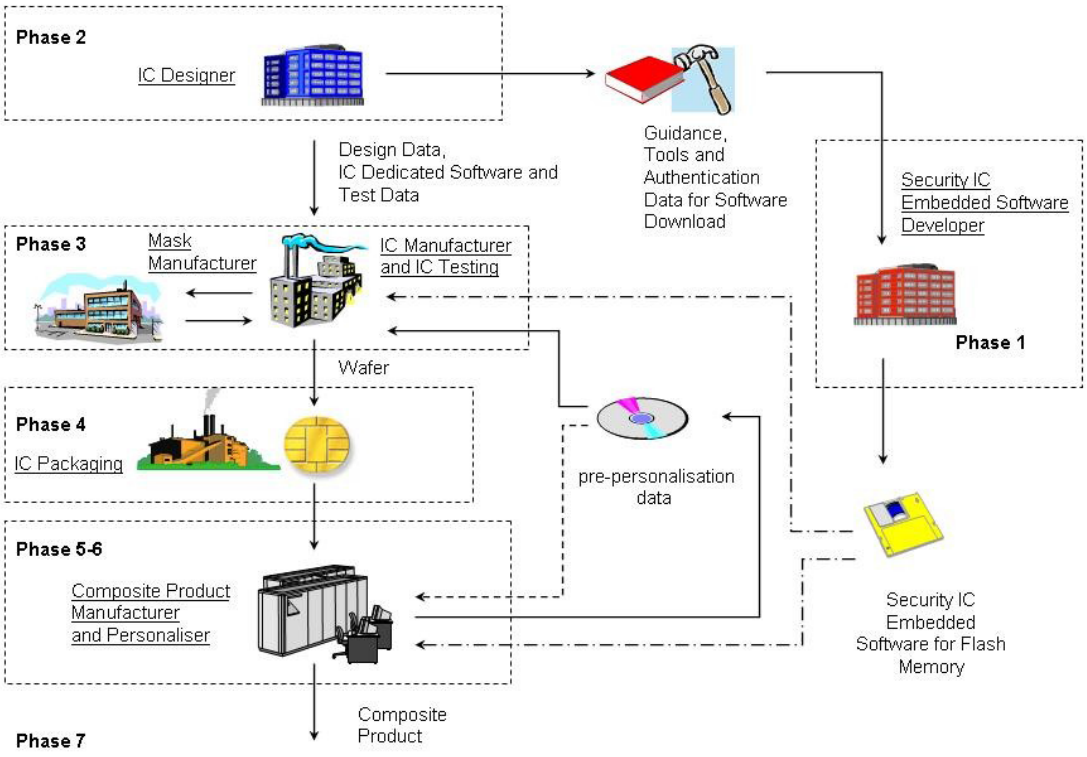


Figure 3 : Secure IC Life-Cycle

1.2.5 Modes of Operation and Life Cycles Phases

The TOE has three distinct modes of operation: boot mode, test mode, user mode.

Test mode is done in a secure environment during manufacturing and testing (Phase 3) and User mode is the operational mode after delivery (after phase 3 from chip point of view).

Boot Mode — This mode is the first entry mode used at each start-up.

Test mode — This mode is designed to allow test engineer to access to test feature of the TOE (Phase 3). This mode is disabled before delivery (at the end of phase 3) and not accessible in operational Mode.

User Mode — This is the mode of operation that the end Secure IC user is intended to be used. The mode is available via the life cycle of the TOE (after phase 3). It is not possible to come back to test mode at this stage.

The Bootloader, including PEOS (not included in the TOE) and Loader, is in the product in Phase 3. Loader will allow to load (in sense of Loader Package 1 and Package 2 Lite of the BSI-CC-PP-0084-2014 [7] and ANSSI interpretation [19]) the Operating system in Phase 5. Loader is used in operational mode and then blocked irreversibly in Phase 5.

1.2.6 TOE Interfaces

In User Mode, the TOE has the following interfaces:

- Physical interface of the TOE with the external environment: the entire chip surface. This interface is taken into account as it contains sensors in order to prevent physical attacks.
- Electrical interfaces of the TOE with the external environment: the pads (the contacted lines I/O, RST, CLK and the power supply lines VCC and GND), as well as the contactless interface (SWP). The communication meets the ISO 7816-3 and ETSI TS 102 613 standards.
- Software interfaces of the TOE with the hardware: registers and CPU instructions.
- Loader interfaces: commands to load the Operating System in phase 5. After the loading, Loader is blocked irreversibly.

1.2.7 TOE Intended Usage

The Secure IC is a platform dedicated to mobile applications running a Customer Operating System (COS).

The Secure IC will be used in a variety of secure applications, including embedded system, authentication, identification, ciphering system.

1.2.8 Forms of delivery

Item Type	Item	Version	Date	Form of delivery
Hardware	ORION_CB_03 & DB_03 microcontroller for Smart Card	CB_03 & DB_03	-	Wafer or dies
Software	BIOS	2.0	-	Included in ORION_CB_03 & DB_03
Software	Loader	2.0	-	Included in ORION_CB_03 & DB_03
Document	Orion – User Manual (Orion_User_Manual_Rev1.2.pdf)	1.2	11/04/2017	Electronic document
Document	Orion Loader – User Manual (UserManual_CC_Loader_v1.7.pdf)	1.7	16/01/2017	Electronic document
Document	Orion – Security Guidance (Orion_Security_Guidance_v30.pdf)	0.30	11/09/2025	Electronic document
Document	Secure 32 bits CPU Instruction Set Architecture (ISA)	1.1b	29/01/2019	Electronic document
Document	Secure 32 bits CPU Embedded Application Binary Interface (EABI)	0.6	March 2013	Electronic document
Document	Guidance – Secure delivery (AGD- Secure delivery-v1.0.pdf)	1.0	12/12/2016	Electronic document
Document	Orion – Assembly Instructions (Orion Assembly – rev 0.2.pdf)	0.2	13/11/2015	Electronic document

Table 5 : Deliveries

The product can be delivered:

- In form of wafer.
- In form of sawn wafer (dice).

The product is sent by a standard transportation

Les TOE user guidance documents are delivered in electronic form. The format of the user guidance documents is .pdf.

2 Conformance Claims

This chapter contains the following sections:

CC Conformance Claim (2.1)

PP Claim (2.2)

Package Claim (2.3)

PP Claim Rationale (2.4)

2.1 CC Conformance Claim

This Security Target claims to be conformant to the Common Criteria 2022 Revision 1.

Furthermore it claims to be **CC Part 2 extended** and **CC Part 3 conformant**. The extended Security Functional Requirements are defined in chapter 5.

This Security IC Security Target has been built with the Common Criteria for Information Technology Security Evaluation; **CC2022, Revision 1**

which comprises

[1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; November 2022, CC2022, Revision 1,

[2] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; November 2022, CC2022, Revision 1

[3] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components; November 2022, CC2022, Revision 1

[4] Common Criteria for Information Technology Security Evaluation, Part 4: Framework for the specification of evaluation methods and activities; November 2022, CC2022 Revision 1.

[5] Common Criteria for Information Technology Security Evaluation, Part 5: Pre-defined packages of Security Requirements; November 2022, CC2022 Revision 1.

[5.1] Errata and Interpretation for CC:2022 (Release 1) and CEM:2022 (Release 1), Version 1.1, 2024-07-22.

The

[6] Common Methodology for Information Technology Security Evaluation (CEM), Evaluation Methodology; November 2022, CC2022, Revision 1

has been taken into account.

2.2 PP Claim

This Security Target is in **strict conformance** to

[7] Security IC Platform Protection Profile; January 2014, Version 1.0, BSI-CC-PP-0084-2014

with additional packages from the BSI-CC-PP-0084-2014 [7] :

- Package “Authentication of the Security IC”.
- Package “Loader dedicated for usage in secured environment only” (Package 1).

This security target is also compliant to a part of the Package “Loader dedicated for usage by authorized users only” (Package 2) contained in the BSI-CC-PP-0084-2014 [7] and named “Package 2 Lite” (Package 2 without confidentiality requirement) in the ANSSI interpretation of the BSI-CC-PP-0084-2014 [19]:

- P.Ctrl_Loader Controlled usage to Loader Functionality.
- O.Ctrl_Auth_Loader Access control and authenticity for the Loader.
- OE.Loader_Usage Secure communication and usage of the Loader.
- FDP_UIT.1 Data exchange integrity.
- FDP_ACC.1/Loader Subset access control – Loader.
- FDP_ACF.1/Loader Security attribute based access control – Loader.

This Security Target take into account the ANSSI interpretation of the BSI-CC-PP-0084-2014 [19].

To be compliant with the ANSSI interpretation of the BSI-CC-PP-0084-2014 [19], the following SFR is added in this security target:

- FTP_TRP.1 Trusted path.

This ST does not claim conformance to any other PP.

2.3 Package Claim

The assurance level for this Security Target is **EAL5** augmented with **AVA_VAN.5** and **ALC_DVS.2**

2.4 PP Claims Rationale

This security target claims strict conformance only to one PP, the “Security IC Platform Protection Profile” BSI-CC-PP-0084-2014 [7].

The Evaluation Assurance Level (EAL) of the Protection Profile BSI-CC-PP-0084-2014 [7] is EAL4 augmented with ALC_DVS.2 and AVA_VAN.5. The Assurance Level required for this TOE is EAL5 augmented with ALC_DVS.2 and AVA_VAN.5 for the TOE. It is to be noted that the following assurance components are added to the assurance level required by the BSI-CC-PP-0084-2014 [7]: ADV_FSP.5, ADV_INT.2, ADV_TDS.4, ALC_CMS.5, ALC_TAT.2 and ATE_DPT.3.

The TOE is an integrated circuit as defined in the protection profile BSI-CC-PP-0084-2014 [7]. So the TOE is consistent with the TOE type of the protection profile BSI-CC-PP-0084-2014 [7].

The security problem definition of this security target is consistent with the statement of the security problem definition in the protection profile BSI-CC-PP-0084-2014 [7], as the security target claims strict conformance to the protection profile BSI-CC-PP-0084-2014 [7]. Additional threats, organisational

security policies and assumptions are introduced in this ST, according to the additional packages contained in the protection profile [7], to the ANSSI Interpretation [19] and to [20]:

- Package “Authentication of the Security IC”:
  - T.Masquerade_TOE Masquerade the TOE.
- Package “Loader dedicated for usage in secured environment only” (Package 1):
  - P.Lim_Block_Loader Limiting and Blocking the Loader Functionality.
- Part of Package “Loader dedicated for usage by authorized users only” (Package 2 Lite):
  - P.Ctrl_Loader Controlled usage to Loader Functionality.
- Additional threats (from [19] and [20]):
  - T.Open_Samples_Diffusion Diffusion of open samples.
  - T.Mem-Access Memory Access Violation.

The security objectives of this security target are consistent with the statement of the security objectives in the protection profile BSI-CC-PP-0084-2014 [7], as the security target claims strict conformance to the protection profile BSI-CC-PP-0084-2014 [7]. Additional security objectives are added in this ST, according to the additional packages contained in the protection profile [7], to the ANSSI Interpretation [19] and to [20]:

- Package “Authentication of the Security IC”:
  - O.Authentication Authentication of external entities.
  - OE.TOE_Auth External entities authenticating of the TOE.
- Package “Loader dedicated for usage in secured environment only” (Package 1):
  - O.Cap_Avail_Loader Capability and availability of the Loader.
  - OE.Lim_Block_Loader Limitation of capability and blocking the Loader.
- Part of Package “Loader dedicated for usage by authorized users only” (Package 2 Lite):
  - O.Ctrl_Auth_Loader Access control and authenticity for the Loader.
  - OE.Loader_Usage Secure communication and usage of the Loader.
- Additional security objectives (from [19] and [20]):
  - O.Prot_TSF_Confidentiality Protection of the confidentiality of the TSF.
  - O.Mem-Access Area based Memory Access Control.

The security requirements of this security target are consistent with the statement of the security requirements in the protection profile BSI-CC-PP-0084-2014 [7], as the security target claims strict conformance to the protection profile BSI-CC-PP-0084-2014 [7]. Additional security requirements are added in this ST:

- Package “Authentication of the Security IC” (from the protection profile BSI-CC-PP-0084-2014 [7]):
  - FIA_API.1 Authentication Proof of Identity.
- Package “Loader dedicated for usage in secured environment only” (Package 1) (from the protection profile BSI-CC-PP-0084-2014 [7]):
  - FMT_LIM.1/Loader Limited capabilities – Loader.

  - FMT_LIM.2/Loader Limited availability – Loader.

- Package 2 Lite “Loader dedicated for usage by authorized users only” (Package 2 without confidentiality requirements) (from the protection profile BSI-CC-PP-0084-2014 [7] and the ANSSI Interpretation [19]):
  - FDP_UIT.1 Data exchange integrity.
  - FDP_ACC.1/Loader Subset access control – Loader.
  - FDP_ACF.1/Loader Security attribute based access control – Loader.
  - FTP_TRP.1 Trusted path.
- Security Functional Requirement for Memory Access Control:
  - FDP_ACC.1 Subset access control.
  - FDP_ACF.1 Security Attribute based access control.
  - FMT_MSA.1 Management of security attributes.
  - FMT_MSA.3 Static attribute initialisation.
  - FMT_SMF.1 Specification of Management Functions.

3 Security Problem Definition

This chapter contains the following sections:

Description of Assets (3.1)

Threats (3.2)

Organisational Security Policies (3.3)

Assumptions (3.4)

3.1 Description of Assets

The assets (related to standard functionality) to be protected are

- the user data of the Composite TOE,
- the Security IC Embedded Software, stored and in operation,
- the security services provided by the TOE for the Security IC Embedded Software.

The user (consumer) of the TOE places value upon the assets related to high-level security concerns:

SC1 integrity of user data of the Composite TOE,

SC2 confidentiality of user data of the Composite TOE being stored in the TOE's protected memory areas,

SC3 correct operation of the security services provided by the TOE for the Security IC Embedded Software.

Note the Security IC Embedded Software is user data and shall be protected while being executed/processed and while being stored in the TOE's protected memories.

The Security IC may not distinguish between user data which is public knowledge or kept confidential. Therefore the security IC shall protect the user data of the Composite TOE in integrity and in confidentiality if stored in protected memory areas, unless the Security IC Embedded Software chooses to disclose or modify it.

In particular integrity of the Security IC Embedded Software means that it is correctly being executed which includes the correct operation of the TOE's functionality. Parts of the Security IC Embedded Software which do not contain secret data or security critical source code, may not require protection from being disclosed. Other parts of the Security IC Embedded Software may need to be kept confidential since specific implementation details may assist an attacker.

The Protection Profile requires the TOE to provide at least one security service: the generation of random numbers by means of a physical Random Number Generator. The Security Target may require additional security services as described in these packages or define TOE specific security services. It is essential that the TOE ensures the correct operation of all security services provided by the TOE for the Security IC Embedded Software.

According to the Protection Profile there is the following high-level security concern related to security service:

SC4 deficiency of random numbers.

To be able to protect these assets (SC1 to SC4) the TOE shall self-protect its TSF. Critical information about the TSF shall be protected by the development environment and the operational environment. Critical information may include:

- logical design data, physical design data, IC Dedicated Software, and configuration data,
- Initialisation Data and Pre-personalisation Data, specific development aids, test and characterisation related data, material for software development support, and photomasks.

Such information and the ability to perform manipulations assist in threatening the above assets.

Note that there are many ways to manipulate or disclose the user data of the Composite TOE: (i) An attacker may manipulate the Security IC Embedded Software or the TOE. (ii) An attacker may cause malfunctions of the TOE or abuse Test Features provided by the TOE. Such attacks usually require design information of the TOE to be obtained. They pertain to all information about (i) the circuitry of the IC (hardware including the physical memories), (ii) the IC Dedicated Software with the parts IC Dedicated Test Software (if any) and IC Dedicated Support Software (if any), and (iii) the configuration data for the TSF. The knowledge of this information may enable or support attacks on the assets. Therefore the TOE Manufacturer must ensure that the development and production of the TOE (refer to Section 1.2.3) is secure so that no restricted, sensitive, critical or very critical information is unintentionally made available for attacks in the operational phase of the TOE (cf. [8] for details on assessment of knowledge of the TOE in the vulnerability analysis).

The TOE Manufacturer must apply protection to support the security of the TOE. This not only pertains to the TOE but also to all information and material exchanged with the developer of the Security IC Embedded Software. This covers the Security IC Embedded Software itself if provided by the developer of the Security IC Embedded Software or any authentication data required to enable the download of software. This includes the delivery (exchange) procedures for Phase 1 and the Phases after TOE Delivery as far as they can be controlled by the TOE Manufacturer. These aspects enforce the usage of the supporting documents and the refinements of SAR defined in the protection profile.

The information and material produced and/or processed by the TOE Manufacturer in the TOE development and production environment (Phases 2 up to TOE Delivery) can be grouped as follows:

- logical design data,
- physical design data,
- IC Dedicated Software, Initialisation Data and Pre-personalisation Data,
- Security IC Embedded Software, provided by the Security IC Embedded Software developer and implemented by the IC manufacturer,
- specific development aids,
- test and characterisation related data,
- material for software development support, and
- photomasks and products in any form

as long as they are generated, stored, or processed by the TOE Manufacturer.

3.2 Threats

The threats are directed against the assets and/or the security functions of the TOE. An overview on attacks is given in PP [7] section 3.2.

The high-level security concerns are refined below by defining threats as required by the Common Criteria (refer to Figure 4). Note that manipulation of the TOE is only a means to threaten user data and is not a success for the attacker in itself.

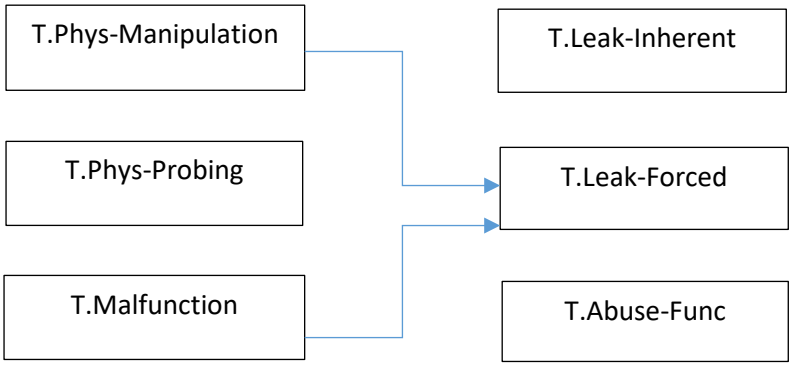


Figure 4: Standards Threats

The high-level security concern related to security service is refined below by defining threats as required by the Common Criteria (refer to Figure 5).

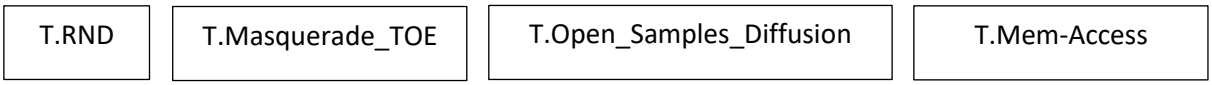


Figure 5: Threats related to security services

The threats to security are defined and described in PP [7] section 3.2.

T.Phys-Manipulation	Physical manipulation
T.Phys-Probing	Physical Probing
T.Malfunction	Malfunction due to Environmental
T.Leak-Inherent	Inherent Information Leakage
T.leak-Forced	Forced Information Leakage
T.Abuse-Func	Abuse of Functionality
T.RND	Deficiency of Random Numbers
T.Masquerade_TOE	Masquerade the TOE
T.Open_Samples_Diffusion	Diffusion of open samples
T.Mem-Access	Memory Access Violation

Table 6 : Threats

Standard Threats

T.Leak-Inherent	Inherent Information Leakage An attacker may exploit information which is leaked from the TOE during usage of the Security IC in order to disclose confidential user data as part of the assets.
T.Phys-Probing	Physical Probing An attacker may perform physical probing of the TOE in order (i) to disclose user data while stored in protected memory areas, (ii) to disclose/reconstruct the user data while processed or (iii) to disclose other critical information about the operation of the TOE to enable attacks disclosing or manipulating the user data of the Composite TOE or the Security IC Embedded Software.
T.Malfunction	Malfunction due to Environmental Stress An attacker may cause a malfunction of TSF or of the Security IC Embedded Software by applying environmental stress in order to (i) modify security services of the TOE or (ii) modify functions of the Security IC Embedded Software (iii) deactivate or affect security mechanisms of the TOE to enable attacks disclosing or manipulating the user data of the Composite TOE or the Security IC Embedded Software. This may be achieved by operating the Security IC outside the normal operating conditions.
T.Phys-Manipulation	Physical Manipulation An attacker may physically modify the Security IC in order to (i) modify user data of the Composite TOE, (ii) modify the Security IC Embedded Software, (iii) modify or deactivate security services of the TOE, or (iv) modify security mechanisms of the TOE to enable attacks disclosing or manipulating the user data of the Composite TOE or the Security IC Embedded Software.
T.Leak-Forced	Forced Information Leakage An attacker may exploit information which is leaked from the TOE during usage of the Security IC in order to disclose confidential user data of the Composite TOE as part of the assets even if the information leakage is not inherent but caused by the attacker.
T.Abuse-Func	Abuse of Functionality An attacker may use functions of the TOE which may not be used after TOE Delivery in order to (i) disclose or manipulate user data of the Composite TOE, (ii) manipulate (explore, bypass, deactivate or change) security services of the TOE or (iii) manipulate (explore, bypass, deactivate or change) functions of the Security IC Embedded Software or (iv) enable an attack disclosing or manipulating the user data of the Composite TOE or the Security IC Embedded Software.

Threats related to security services

T.RND Deficiency of Random Numbers

An attacker may predict or obtain information about random numbers generated by the TOE security service for instance because of a lack of entropy of the random numbers provided.

Package "Authentication of the Security IC"

T.Masquerade_TOE Masquerade the TOE

An attacker may threaten the property being a genuine TOE by producing an IC which is not a genuine TOE but wrongly identifying itself as genuine TOE sample.

Additional threats (provided by [19] and [20])

T.Open_Samples_Diffusion Diffusion of open samples

An attacker may get access to open samples of the TOE and use them to gain information about the TSF (loader, memory management unit, ROM code…). He may also use the open samples to characterize the behavior of the IC and its security functionalities (for example: characterization of side channel profiles, perturbation cartography…). The execution of a dedicated security features (for example: execution of a DES computation without countermeasures or by de-activating countermeasures) through the loading of an adequate code would allow this kind of characterization and the execution of enhanced attacks on the IC.

T.Mem-Access Memory Access Violation

Parts of the Smartcard Embedded Software may cause security violations by accidentally or deliberately accessing restricted data (which may include code). Any restrictions are defined by the security policy of the specific application context and must be implemented by the Smartcard Embedded Software.

3.3 Organisational Security Policies

Core PP

The IC Developer / Manufacturer must apply the policy "Identification during TOE Development and Production (P.Process-TOE)" as specified below.

P.Process-TOE Identification during TOE Development and Production

An accurate identification must be established for the TOE. This requires that each instantiation of the TOE carries this unique identification.

The accurate identification is introduced at the end of the production test in phase 3. Therefore the production environment must support this unique identification.

Package 1: Loader dedicated for usage in secured environment only

The organisational security policy "Limiting and Blocking the Loader Functionality (P.Lim_Block_Loader)" applies to Loader dedicated for usage in secured environment.

P.Lim_Block_Loader Limiting and Blocking the Loader Functionality

The composite manufacturer uses the Loader for loading of Security IC Embedded Software, user data of the Composite Product or IC Dedicated Support Software in charge of the IC Manufacturer. He limits the capability and blocks the availability of the Loader in order to protect stored data from disclosure and manipulation.

Package 2 Lite: Loader dedicated for usage by authorized users only

The organisational security policy "Controlled usage to Loader Functionality (P.Ctlr_Loader)" applies to Loader dedicated for usage by authorized users only.

P.Ctlr_loader Controlled usage to Loader Functionality

Authorized user controls the usage of the loader functionality in order to protect stored and loader user data from disclosure and manipulation.

3.4 Assumptions

The TOE assumptions on the operational environment are defined and described in PP [7] section 3.4.

A.Process-Sec-IC	Protection during Packaging, Finishing and Personalisation
A.Resp-Appl	Treatment of user data of the Composite TOE

Table 7 : Assumptions

Core PP

Before being delivered to the consumer the TOE is packaged. Many attacks require the TOE to be removed from the carrier. Though this extra step adds difficulties for the attacker no specific assumptions are made here regarding the package.

Appropriate "Protection during Packaging, Finishing and Personalisation (A.Process-Sec-IC)" must be ensured after TOE Delivery up to the end of Phase 6, as well as during the delivery to Phase 7 as specified below.

A.Process-Sec-IC Protection during Packaging, Finishing and Personalisation

It is assumed that security procedures are used after delivery of the TOE by the TOE Manufacturer up to delivery to the end-consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use).

The information and material produced and/or processed by the Security IC Embedded Software Developer in Phase 1 and by the Composite Product Manufacturer can be grouped as follows:

- the Security IC Embedded Software including specifications, implementation and related documentation,

- Pre-personalisation Data and Personalisation Data including specifications of formats and memory areas, test related data,

- the user data of the Composite TOE and related documentation, and

- material for software development support

as long as they are not under the control of the TOE Manufacturer. Details must be defined in the Protection Profile or Security Target for the evaluation of the Security IC Embedded Software and/or Security IC.

The developer of the Security IC Embedded Software must ensure the appropriate usage of Security IC while developing this software in Phase 1 as described in the (i) TOE guidance documents (refer to the Common Criteria assurance class AGD) such as the hardware data sheet, and the hardware application notes, and (ii) findings of the TOE evaluation reports relevant for the Security IC Embedded Software as documented in the certification report.

Note that particular requirements for the Security IC Embedded Software are often not clear before considering a specific attack scenario during vulnerability analysis of the Security IC (AVA_VAN). A summary of such results is provided in the document "ETR for composite evaluation" (ETR-COMP), see [21]. This document will be provided for the evaluation of the composite product (see [13]). The ETR-COMP may also include guidance for additional tests being required for the combination of hardware and software. The TOE evaluation must be completed before evaluation of the Security IC Embedded Software can be completed. The TOE evaluation can be conducted before and independently from the evaluation of the Security IC Embedded Software.

The Security IC Embedded Software must ensure the appropriate “Treatment of user data of the Composite TOE (A.Resp-Appl)” as specified below.

A.Resp-Appl	Treatment of user data of the Composite TOE All user data of the Composite TOE are owned by Security IC Embedded Software. Therefore, it must be assumed that security relevant user data of the Composite TOE (especially cryptographic keys) are treated by the Security IC Embedded Software as defined for its specific application context.

The application context specifies how the user data of the Composite TOE shall be handled and protected. The evaluation of the Security IC according to this Security Target is conducted on generalized application context. The concrete requirements for the Security IC Embedded Software shall be defined in the Protection Profile respective Security Target for the Security IC Embedded Software. The Security IC cannot prevent any compromise or modification of user data of the Composite TOE by malicious Security IC Embedded Software.

4 Security Objectives

This chapter Security Objectives contains the following sections:

Security Objectives for the TOE (4.1)

Security Objectives for the Security IC Embedded Software *(4.2)*

Security Objectives for the operational Environment *(4.3)*

Security Objectives Rationale (4.4)

The full details of the Security Objectives are listed in PP-BSI-0084 [7].

4.1 Security Objectives for the TOE

The user have the following standard high-level security goals related to the assets:

SG1 maintain the integrity of user data (when being executed/processed and when being stored in the TOE's memories) as well as

SG2 maintain the confidentiality of user data (when being processed and when being stored in the TOE's protected memories).

SG3 maintain the correct operation of the security services provided by the TOE for the Security IC Embedded Software.

Note, the Security IC may not distinguish between user data which are public known or kept confidential. Therefore the security IC shall protect the user data in integrity and in confidentiality if stored in protected memory areas, unless the Security IC Embedded Software chooses to disclose or modify it. Parts of the Security IC Embedded Software which do not contain secret data or security critical source code, may not require protection from being disclosed. Other parts of the Security IC Embedded Software may need kept confidential since specific implementation details may assist an attacker.

These standard high-level security goals in the context of the security problem definition build the starting point for the definition of security objectives as required by the Common Criteria (refer to Figure 6 Standard Security Objectives). Note that the integrity of the TOE is a means to reach these objectives.

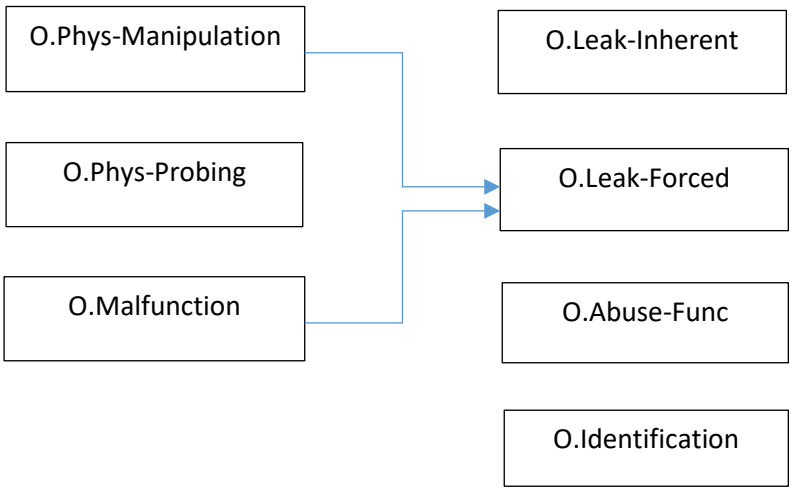


Figure 6: Standard Security Objectives

According to the Protection Profile there is the following high-level security goal related to specific functionality:

SG4 provide true random numbers.

The additional high-level security considerations are refined below by defining security objectives as required by the Common Criteria.

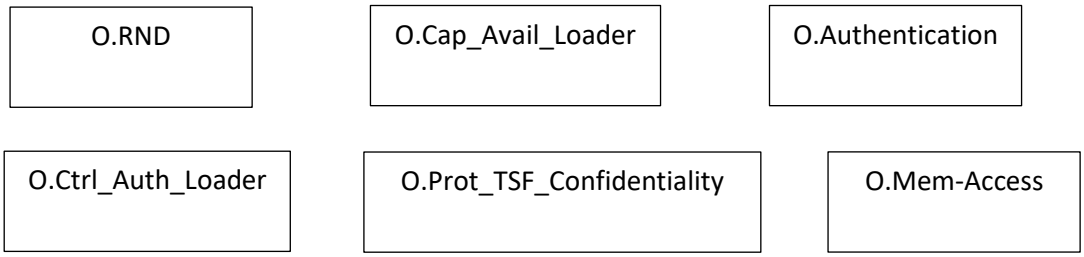


Figure 7: Security Objectives related to Specific Functionality

The security objectives of the TOE are:

O.Leak-Inherent	Protection against Inherent Information Leakage
O.Phys-Probing	Protection against Physical Probing
O.Malfunction	Protection against Malfunctions
O.Phys-Manipulation	Protection against Physical Manipulation
O.Leak-Forced	Protection against Forced Information Leakage
O.Abuse-Func	Protection against Abuse of Functionality
O.Identification	TOE Identification
O.RND	Random Numbers

O.Cap_Avail_Loader	Capability and availability of the Loader
O.Authentication	Authentication to external entities
O.Ctrl_Auth_Loader	Access control and authenticity for the loader
0.Prot_TSF_Confidentiality	Protection of the confidentiality of the TSF
O.Mem-Access	Area based Memory Access Control

Table 8 : Objectives for the TOE

Standard Security Objectives

O.Leak-Inherent Protection against Inherent Information Leakage

The TOE must provide protection against disclosure of confidential data stored and/or processed in the Security IC

- by measurement and analysis of the shape and amplitude of signals (for example on the power, clock, or I/O lines) and

- by measurement and analysis of the time between events found by measuring signals (for instance on the power, clock, or I/O lines).

O.Phys-Probing Protection against Physical Probing

The TOE must provide protection against disclosure/reconstruction of user data while stored in protected memory areas and processed or against the disclosure of other critical information about the operation of the TOE.

This includes protection against

- measuring through galvanic contacts which is direct physical probing on the chips surface except on pads being bonded (using standard tools for measuring voltage and current) or

- measuring not using galvanic contacts but other types of physical interaction between charges (using tools used in solid-state physics research and IC failure analysis)

with a prior reverse-engineering to understand the design and its properties and functions.

The TOE 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 must ensure its correct operation.

The TOE must indicate or prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent malfunctions. Examples of environmental conditions are voltage, clock frequency, temperature, or external energy fields.

O.Phys-Manipulation	Protection against Physical Manipulation The TOE must provide protection against manipulation of the TOE (including its software and TSF data), the Security IC Embedded Software and the user data of the Composite TOE. This includes protection against - reverse-engineering (understanding the design and its properties and functions), - manipulation of the hardware and any data, as well as - undetected manipulation of memory contents.
O.Leak-Forced	Protection against Forced Information Leakage The Security IC must be protected against disclosure of confidential data processed in the Security IC (using methods as described under O.Leak-Inherent) even if the information leakage is not inherent but caused by the attacker - by forcing a malfunction (refer to “Protection against Malfunction due to Environmental Stress (O.Malfunction)” and/or - by a physical manipulation (refer to “Protection against Physical Manipulation (O.Phys-Manipulation)”. If this is not the case, signals which normally do not contain significant information about secrets could become an information channel for a leakage attack.
O.Abuse-Func	Protection against Abuse of Functionality The TOE must prevent that functions of the TOE which may not be used after TOE Delivery can be abused in order to (i) disclose critical user data of the Composite TOE, (ii) manipulate critical user data of the Composite TOE, (iii) manipulate Security IC Embedded Software or (iv) bypass, deactivate, change or explore security features or security services of the TOE. Details depend, for instance, on the capabilities of the Test Features provided by the IC Dedicated Test Software which are not specified here.
O.Identification	TOE Identification The TOE 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 identification.

Security Objectives related to Specific Functionality (referring to SG4)

O.RND	Random Numbers The TOE will ensure the cryptographic quality of random number generation. For instance random numbers shall not be predictable and shall have a sufficient entropy.

The TOE will ensure that no information about the produced random numbers is available to an attacker since they might be used for instance to generate cryptographic keys.

Package 1: Loader dedicated for usage in secured environment only

The TOE shall provide “Capability and availability of the Loader O.Cap_Avail_Loader)” as specified below.

O.Cap_Avail_Loader	Capability and availability of the Loader The TSF provides limited capability of the Loader functionality and irreversible termination of the Loader in order to protect stored user data from disclosure and manipulation.

Package “Authentication of the Security IC”

The TOE shall provide “Authentication to external entities (O.Authentication)” as specified below.

O.Authentication	Authentication to external entities The TOE shall be able to authenticate itself to external entities. The Initialisation Data (or parts of them) are used for TOE authentication verification data.

Package 2 Lite: Loader dedicated for usage by authorized users only

The TOE shall provide “Access control and authenticity for the Loader (O.Ctrl_Auth_Loader)” as specified below.

O.Ctrl_Auth_Loader	Access control and authenticity for the loader The TSF provides trusted communication channel with authorized user, supports authentication of the user data to be loaded and access control for usage of the Loader functionality.

Additional security objective for the TOE (provided by [19] and [20]):

The TOE shall provide “Protection of the confidentiality of the TSF (O.Prot_TSF_Confidentiality)” as specified below:

O.Prot_TSF_Confidentiality	Protection of the confidentiality of the TSF The TOE must provide protection against disclosure of confidential operations of the Security IC (loader, memory management unit…) through the use of a dedicated code loaded on open samples.

The TOE shall provide “Area based Memory Access Control (O.Mem-Access)” as specified below:

O.Mem-Access	Area based Memory Access Control The TOE must provide the Smartcard Embedded Software with the capability to define restricted access memory areas. The TOE must then enforce the partitioning of such memory areas so that access of software to memory areas is controlled as required, for example, in a multi-application environment.

4.2 Security Objective for the Security IC Embedded Software

The development of the Security IC Embedded Software is outside the development and manufacturing of the TOE (cf. section 1.2.4). The Security IC Embedded Software defines the operational use of the TOE. This section describes the security objective for the Security IC Embedded Software.

Note, in order to ensure that the TOE is used in a secure manner the Security IC Embedded Software shall be designed so that the requirements from the following documents are met: (i) hardware data sheet for the TOE, (ii) data sheet of the IC Dedicated Software of the TOE, (iii) TOE application notes, other guidance documents, and (iv) findings of the TOE evaluation reports relevant for the Security IC Embedded Software as referenced in the certification report.

Core PP

The Security IC Embedded Software shall provide “Treatment of user data of the Composite TOE (OE.Resp-Appl)” as specified below.

OE.Resp-Appl	Treatment of user data of the Composite TOE Security relevant user data of the Composite TOE (especially cryptographic keys) are treated by the Security IC Embedded Software as required by the security needs of the specific application context.

For example the Security IC Embedded Software will not disclose security relevant user data of the Composite TOE to unauthorised users or processes when communicating with a terminal.

4.3 Security Objectives for the Operational Environment

TOE Delivery up to the end of Phase 6

Appropriate “Protection during Packaging, Finishing and Personalisation (OE.Process-Sec-IC)” must be ensured after TOE Delivery up to the end of Phases 6, as well as during the delivery to Phase 7 as specified below.

OE.Process-Sec-IC	Protection during composite product manufacturing Security procedures shall be used after TOE Delivery up to delivery to the end-consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data

(to prevent any possible copy, modification, retention, theft or unauthorised use).

This means that Phases after TOE Delivery up to the end of Phase 6 (refer to Section 1.2.4) must be protected appropriately. For a preliminary list of assets to be protected refer to paragraph 96 (page 29) of PP [7].

Package 1: Loader dedicated for usage in secured environment only

The operational environment of the TOE shall provide "Limitation of capability and blocking the Loader (OE.Lim_Block_Loader)" as specified below.

OE.Lim_Block_Loader Limitation of capability and blocking the loader

The Composite Product Manufacturer will protect the Loader functionality against misuse, limit the capability of the Loader and terminate irreversibly the Loader after intended usage of the Loader.

Package "Authentication of the Security IC"

The operational environment shall provide "External entities authenticating of the TOE OE.TOE_Auth)".

OE.TOE_Auth External entities authenticating of the TOE

The operational environment shall support the authentication verification mechanism and know authentication reference data of the TOE.

Package 2 Lite: Loader dedicated for usage by authorized users only

The operational environment of the TOE shall provide "Secure communication and usage of the Loader (OE.Loader_Usage)" as specified below.

OE.Loader_Usage Secure communication and usage of the loader

The authorized user must fulfil the access conditions required by the Loader.

4.4 Security Objectives Rationale

Table 8 below gives an overview, how the assumptions, threats, and organisational security policies are addressed by the objectives. The text following after the table justifies this in detail.

Assumption, Threat or Organisational Security Policy	Security Objective	Notes
A.Resp-Appl	OE.Resp-Appl	
P.Process-TOE	O.identification	Phase 2 – 3 optional Phase 4
A.Process-Sec-IC	OE.Process-Sec-IC	Phase 5 – 6 optional Phase 4
T.Leak-Inherent	O.Leak-Inherent	
T.Phys-Probing	O.Phys-Probing	

T.Malfunction	O.Malfunction	
T.Phys-Manipulation	O.Phys-Manipulation	
T.Leak-Forced	O.Leak-Forced	
T.Abuse-Func	O.Abuse-Func O.Cap_Avail_Loader	
T.RND	O.RND	
T.Open_Samples_Diffusion	O.Prot_TSF_Confidentiality O.Leak-Inherent O.Leak-Forced	
P.Lim_Block_Loader	O.Cap_Avail_Loader OE.Lim_Block_Loader	Phase 3 to phase 5.
T.Masquerade_TOE	O.Authentication OE.TOE.Auth	
P.Ctlr_loader	O.Ctrl_Auth_Loader OE.Loader_Usage	Phase 3 to phase 5.
T.Mem-Access	O.Mem-Access	

Table 9 : Security Objective versus Assumptions, Threats or Policy

Core PP

The justification related to the assumption “Treatment of user data of the Composite TOE (**A.Resp-Appl**)” is as follows:

Since OE.Resp-Appl requires the Security IC Embedded Software to implement measures as assumed in A.Resp-Appl, the assumption is covered by the objective.

The justification related to the organisational security policy “Protection during TOE Development and Production (**P.Process-TOE**)” is as follows:

O.Identification requires that the TOE has to support the possibility of a unique identification. The unique identification can be stored on the TOE. Since the unique identification is generated by the production environment the production environment must support the integrity of the generated unique identification. The technical and organisational security measures that ensure the security of the development environment and production environment are evaluated based on the assurance measures that are part of the evaluation. For a list of material produced and processed by the TOE Manufacturer refer to section 3.1 page 25 (paragraph 69, page 21 in the PP [7]). All listed items and the associated development and production environments are subject of the evaluation. Therefore, the organisational security policy P.Process-TOE is covered by this objective, as far as organisational measures are concerned.

The justification related to the assumption “Protection during Packaging, Finishing and Personalisation (**A.Process-Sec-IC**)” is as follows:

Since OE.Process-Sec-IC requires the Composite Product Manufacturer to implement those measures assumed in A.Process-Sec-IC, the assumption is covered by this objective.

The justification related to the threats “Inherent Information Leakage (**T.Leak-Inherent**)”, “Physical Probing (**T.Phys-Probing**)”, “Malfunction due to Environmental Stress (**T.Malfunction**)”, “Physical Manipulation (**T.Phys-Manipulation**)”, “Forced Information Leakage (**T.Leak-Forced**)“, “Abuse of Functionality (**T.Abuse-Func**)” and “Deficiency of Random Numbers (**T.RND**)” is as follows:

For all threats the corresponding objectives (refer to Table 9) are stated in a way, which directly corresponds to the description of the threat (refer to Section 3.2). It is clear from the description of each objective (refer to Section 4.1), that the corresponding threat is removed if the objective is valid. More specifically, in every case the ability to use the attack method successfully is countered, if the objective holds.

Package “Authentication of the Security IC”

The threat “Masquerade the TOE (**T.Masquerade_TOE**)” is directly covered by the TOE security objective “Authentication to external entities (O.Authentication)” describing the proving part of the authentication and the security objective for the operational environment of the TOE “External entities authenticating of the TOE (OE.TOE_Auth)” the verifying part of the authentication.

Package 1: Loader dedicated for usage in secured environment only

The organisational security policy Limitation of capability and blocking the Loader (**P.Lim_Block_Loader**) is directly implemented by the security objective for the TOE “Capability and availability of the Loader (O.Cap_Avail_Loader)” and the security objective for the TOE environment “Limitation of capability and blocking the Loader (OE.Lim_Block_Loader)”.

The TOE security objective “Capability and availability of the Loader” (O.Cap_Avail_Loader)” mitigates also the threat “Abuse of Functionality “(**T.Abuse-Func**) if attacker tries to misuse the Loader functionality in order to manipulate security services of the TOE provided or depending on IC Dedicated Support Software or user data of the TOE as IC Embedded Software, TSF data or user data of the smartcard product.

Additional threats (provided by [19] and [20]):

The threat “Diffusion of open samples” (**T.Open_Samples_Diffusion**) is directly covered by the TOE security objective “Protection of the confidentiality of the TSF” (O.Prot_TSF_Confidentiality) based on the self-protection of the TOE and the authentication mechanism of the Loader.

Additionally, **T.Open_Samples_Diffusion** threat is countered by “Protection against Inherent Information Leakage” (O.Leak-Inherent) and “Protection against Forced Information Leakage” (O.Leak-Forced) from the PP.

The TOE security objective “Area based Memory Access Control” (O.Mem-Access) counters the threats “Memory Access Violation” (**T.Mem-Access**). According to O.Mem-Access the TOE must enforce the partitioning of memory areas so that access of software to memory areas is controlled. Any restrictions are to be defined by the Smartcard Embedded Software. Thereby security violations caused by accidental or deliberate access to restricted data (which may include code) can be prevented. The threat T.Mem-Access is therefore removed if the objective is met.

The TOE shall provide access control functions as a means to be used by the Smartcard Embedded Software. This is further emphasised by the clarification of “Treatment of User Data (OE.Resp-Appl)” which reminds that the Smartcard Embedded Software must not undermine the restrictions.

Package 2 Lite: Loader dedicated for usage by authorized users only

The organisational security policy “Controlled usage to Loader Functionality” (**P.Ctlr_Loader**) is directly implemented by the security objective for the TOE “Access control and authenticity for the Loader (O.Ctrl_Auth_Loader)” and the security objective for the TOE environment “Secure communication and usage of the Loader (OE.Loader_Usage)”.

5 Extended Components Definitions

5.1 Definition of the Family FAU_SAS

To define the security functional requirements of the TOE an additional family (FAU_SAS) of the Class FAU (Security Audit) is defined here. This family describes the functional requirements for the storage of audit data. It has a more general approach than FAU_GEN, because it does not necessarily require the data to be generated by the TOE itself and because it does not give specific details of the content of the audit records.

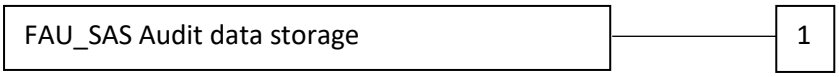
The family “Audit data storage (FAU_SAS)” is specified as follows.

FAU_SAS Audit data storage

Family behaviour

This family defines functional requirements for the storage of audit data.

Component levelling



FAU_SAS.1 Requires the TOE to provide the possibility to store audit data.

Management: FAU_SAS.1

There are no management activities foreseen.

Audit: FAU_SAS.1

There are no actions defined to be auditable.

FAU_SAS.1 Audit storage

Hierarchical to: No other components.

Dependencies: No dependencies.

FAU_SAS.1.1 The TSF shall provide [assignment: *list of subjects*] with the capability to store [assignment: *list of audit information*] in the [assignment: *type of persistent memory*].

6 IT Security Requirements

This chapter *IT Security* Requirements contains the following sections:

Security Functional Requirements for the TOE (6.1)

Security Assurance Requirements for the TOE (6.2)

Security Requirements Rationale (6.3)

- R*ationale for the security functional requirements (6.3.1)*
- D*ependencies of security functional requirements (6.3.2)*
- R*ationale for the Assurance Requirements (6.3.3)*
- S*ecurity Requirements are Internally Consistent (6.3.4)*

6.1 Security Functional Requirements for the TOE

6.1.1 Convention

In order to define the Security Functional Requirements Part 2 of the Common Criteria was used. However, some Security Functional Requirements have been refined. The refinements are described below the associated SFR.

The **refinement** operation is used to add detail to a requirement, and, thus, further restricts a requirement. When an interpretation refinement is given, an extra paragraph starting with “Refinement” is given.

The **selection** operation is used to select one or more options provided by the CC in stating a requirement. Selections having been made by the BSI-CC-PP-0084-2014 author are denoted as underlined text. Selections fill in by this ST author appear are denoted as underlined and *italicised* text, like this.

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments having been made by the BSI-CC-PP-0084-2014 author are denoted as underlined text. Assignments fill in by this ST author are denoted as underlined and *italicised* text, like this.

The **iteration** operation is used when a component is repeated with varying operations. Iteration is denoted by showing a slash “/”, and the iteration indicator after the component identifier.

The security functional requirements (SFR) for the TOE are defined and described in the PP [7] section 6.1 and in the following description.

6.1.2 Malfunction

The TOE shall meet the requirement “Limited fault tolerance (FRU_FLT.2)” as specified below.

FRU_FLT.2	**Limited fault tolerance**
Hierarchical to:	FRU_FLT.1 Degraded fault tolerance
Dependencies:	FPT_FLS.1 Failure with preservation of secure state.
FRU_FLT.2.1	The TSF shall ensure the operation of all the TOE’s capabilities when the following failures occur: exposure to operating conditions which are not detected according to the requirement Failure with preservation of secure state (FPT_FLS.1).
Refinement:	**The term “failure” above means “circumstances”. The TOE prevents failures for the “circumstances” defined above.**

The TOE shall meet the requirement “Failure with preservation of secure state (FPT_FLS.1)” as specified below.

FPT_FLS.1	**Failure with preservation of secure state**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
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”. The TOE prevents failures for the “circumstances” defined above.**

6.1.3 Abuse of Functionality

The TOE shall meet the requirement “Limited capabilities (FMT_LIM.1)” as specified below.

FMT_LIM.1	**Limited capabilities**
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.2 Limited availability.
FMT_LIM.1.1	The TSF shall be designed and implemented in a manner that limits their capabilities so that in conjunction with “Limited availability (FMT_LIM.2)” the following policy is enforced: Deploying Test Features after TOE Delivery does not allow user data of the Composite TOE 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.

The TOE shall meet the requirement “Limited availability (FMT_LIM.2)” as specified below.

FMT_LIM.2	**Limited availability**
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.1 Limited capabilities.
FMT_LIM.2.1	The TSF shall be designed and implemented in a manner that limits their availability so that in conjunction with “Limited capabilities (FMT_LIM.1)” the following policy is enforced: Deploying Test Features after TOE Delivery does not allow user data of the Composite TOE 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.

The TOE shall meet the requirement “Audit storage (FAU_SAS.1)” as specified below (Common Criteria Part 2 extended).

FAU_SAS.1	**Audit storage**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FAU_SAS.1.1	The TSF shall provide the test process before TOE Delivery with the capability to store *the Initialization Data and/or Pre-personalization Data and/or supplements of the Security IC Embedded Software* in the *Non-Volatile Memory (FLASH).*

6.1.4 Physical Manipulation and Probing

The TOE shall meet the requirement “Stored data confidentiality (FDP_SDC.1)” as specified below.

FDP_SDC.1	**Stored data confidentiality**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FDP_SDC.1.1	The TSF shall ensure the confidentiality of the information of the user data while it is stored in the *any memories.*

The TOE shall meet the requirement "Stored data integrity monitoring and action (FDP_SDI.2)" as specified below.

FDP_SDI.2/RAM	**Stored data integrity monitoring and action – RAM**
Hierarchical to:	FDP_SDI.1 Stored data integrity monitoring
Dependencies:	No dependencies.
FDP_SDI.2.1/RAM	The TSF shall monitor user data stored in containers controlled by the TSF for *redundancy bits* on all objects, based on the following attributes: *RAM.*
FDP_SDI.2.2/RAM	Upon detection of a data integrity error, the TSF shall *send an alarm to the Alarm Management within SEC Manager*.

FDP_SDI.2/NVM	**Stored data integrity monitoring and action – NVM**
Hierarchical to:	FDP_SDI.1 Stored data integrity monitoring
Dependencies:	No dependencies.
FDP_SDI.2.1/NVM	The TSF shall monitor user data stored in container controlled by TSF for *Anti Re-routing mechanism* on all objects, based on the following attributes: *NVM.*
FDP_SDI.2.2/NVM	Upon detection of a data integrity error, the TSF shall *send an alarm to the Alarm Management within SEC Manager*.

FDP_SDI.2/Register&Bus	**Stored data integrity monitoring and action – Register&Bus**
Hierarchical to:	FDP_SDI.1 Stored data integrity monitoring
Dependencies:	No dependencies.
FDP_SDI.2.1/Register&Bus	The TSF shall monitor user data stored in containers controlled by the TSF for *redundancy bits* on all objects, based on the following attributes: *Registers and Buses.*
FDP_SDI.2.2/Register&Bus	Upon detection of a data integrity error, the TSF shall *send an alarm to the Alarm Management within SEC Manager*.

The TOE shall meet the requirement “Resistance to physical attack (FPT_PHP.3)” as specified below.

FPT_PHP.3	**Resistance to physical attack**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_PHP.3.1	The TSF shall resist physical manipulation and physical probing to the TSF by responding automatically such that the SFRs are always enforced.
Refinement:	**The TSF will implement appropriate mechanisms to continuously counter physical manipulation and physical probing. Due to the nature of these attacks (especially manipulation) the TSF can by no means detect attacks on all of its elements. Therefore, permanent protection against these attacks is required ensuring that security functional requirements are enforced. Hence, “automatic response” means here (i) assuming that there might be an attack at any time and (ii) countermeasures are provided at any time.**

6.1.5 Leakage

The TOE shall meet the requirement “Basic internal transfer protection (FDP_ITT.1)” as specified below.

FDP_ITT.1	**Basic internal transfer protection**
Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
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.
Refinement:	**The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor) are seen as physically-separated parts of the TOE.**

The TOE shall meet the requirement “Basic internal TSF data transfer protection (FPT_ITT.1)” as specified below.

FPT_ITT.1	**Basic internal TSF data transfer protection**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_ITT.1.1	The TSF shall protect TSF data from disclosure when it is transmitted between separate parts of the TOE.

Refinement: **The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor) are seen as separated parts of the TOE.**

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.

The TOE shall meet the requirement "Subset information flow control (FDP_IFC.1)" as specified below:

FDP_IFC.1	**Subset information flow control**
Hierarchical to:	No other components.
Dependencies:	FDP_IFF.1 Simple security attributes
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 or by the Security IC Embedded Software.

The following Security Function Policy (SFP) Data Processing Policy is defined for the requirement "Subset information flow control (FDP_IFC.1)":

"User data of the Composite TOE and TSF data shall not be accessible from the TOE except when the Security IC Embedded Software decides to communicate the user data of the Composite TOE via an external interface. The protection shall be applied to confidential data only but without the distinction of attributes controlled by the Security IC Embedded Software."

6.1.6 Random Numbers

The TOE shall meet the requirement "Quality metric for random numbers (FCS_RNG.1)" as specified below.

FCS_RNG.1 /PTG.2	**Random number generation – PTG.2**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FCS_RNG.1.1/PTG.2	The TSF shall provide a *physical* random number generator that implements:
(PTG.2.1)	A total failure test detects a total failure of entropy source immediately when the RNG has started. When a total failure is detected, no random numbers will be output.
(PTG.2.2)	If a total failure of the entropy source occurs while the RNG is being operated, the RNG *prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source*.

(PTG.2.3) The online test shall detect non-tolerable statistical defects of the raw random number sequence (i) immediately when the RNG has started, and (ii) while the RNG is being operated. The TSF must not output any random numbers before the power-up online test has finished successfully or when a defect has been detected.

(PTG.2.4) The online test procedure shall be effective to detect non-tolerable weaknesses of the random numbers soon.

(PTG.2.5) The online test procedure checks the quality of the raw random number sequence. It is triggered *continuously*. The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time.

FCS_RNG.1.2 /PTG.2 The TSF shall provide *32-bit numbers* that meet

(PTG.2.6) Test procedure A does not distinguish the internal random numbers from output sequences of an ideal RNG.

(PTG.2.7) The average Shannon entropy per internal random bit exceeds 0.997.

6.1.7 Loader – Package 1

The TOE Functional Requirement “Limited capabilities – Loader (FMT_LIM.1/Loader)” is specified as follows.

FMT_LIM.1/Loader	**Limited capabilities – Loader**
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.2 Limited availability.
FMT_LIM.1.1/Loader	The TSF shall limit its capabilities so that in conjunction with “Limited availability (FMT_LIM.2)” the following policy is enforced: Deploying Loader functionality *after full loading of Embedded Software and locking of the Loader* does not allow stored user data to be disclosed or manipulated by unauthorized user.

The TOE Functional Requirement “Limited availability – Loader (FMT_LIM.2/Loader)” is specified as follows.

FMT_LIM.2/Loader	**Limited availability - Loader**
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.1 Limited capabilities.
FMT_LIM.2.1/Loader	The TSF shall be designed in a manner that limits its availability so that in conjunction with “Limited capabilities (FMT_LIM.1)” the following policy is

enforced: The TSF prevents deploying the Loader functionality after *full loading of Embedded Software and locking of the Loader*.

6.1.8 Authentication Proof of Identity

The TOE shall meet the requirement "Authentication Proof of Identity (FIA_API.1)" as specified below.

FIA_API.1	**Authentication Proof of Identity**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_API.1.1	The TSF shall provide a *mutual cryptographic authentication mechanism* to prove the identity of the TOE, *IC loader authorized people* to an external entity.

6.1.9 Loader Package 2 Lite

The TOE Functional Requirement "Data exchange integrity (FDP_UIT.1)" is specified as follows.

FDP_UIT.1	**Data exchange integrity**
Hierarchical to:	No other components.
Dependencies:	[FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_UIT.1.1	The TSF shall enforce the Loader SFP to receive user data in a manner protected from modification, deletion, insertion errors.
FDP_UIT.1.2	The TSF shall be able to determine on receipt of user data, whether modification, deletion, insertion has occurred.

The TOE Functional Requirement "Subset access control - Loader (FDP_ACC.1/Loader)" is specified as follows.

FDP_ACC.1/ Loader	**Subset access control - Loader**
Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control.
FDP_ACC.1.1/ Loader	The TSF shall enforce the Loader SFP on (1) the subjects *Loader authorized persons*, (2) the objects user data in *Non Volatile Memory (Flash)*, (3) the operation deployment of Loader

The TOE Functional Requirement “Security attribute based access control – Loader (FDP_ACF.1/Loader)” is specified as follows.

FDP_ACF.1/ Loader	**Security attribute based access control – Loader**
Hierarchical to:	No other components.
Dependencies:	FMT_MSA.3 Static attribute initialisation FDP_ACC.1 Subset access control - Loader
FDP_ACF.1.1/Loader	The TSF shall enforce the Loader SFP to objects based on the following: (1) the subjects *Loader authorized persons* with security attributes *controlling the right address range access* (2) the objects user data in *Non Volatile Memory (Flash)* with security attributes *controlling the right address range access*
FDP_ACF.1.2/ Loader	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed*: the loading operation is allowed if and only if the subject has been successfully authenticated to the TSF by mutual authentication, and the load address of the object is located inside the address range dedicated for loading.*
FDP_ACF.1.3/ Loader	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *none*
FDP_ACF.1.4/Loader	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *blocking of the Loader*.

6.1.10 Trusted path

The TOE Functional Requirement “Trusted path” (FTP_TRP.1) is specified as follows.

FTP_TRP.1	**Trusted path**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FTP_TRP.1.1	The TSF shall provide a communication path between itself and *remote, local* users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from *modification*.
FTP_TRP.1.2	The TSF shall permit *local users, remote users* to initiate communication via the trusted path.
FTP_TRP.1.3	The TSF shall require the use of the trusted path for *initial user authentication*.

6.1.11 Memory Access Control

The TOE Functional Requirement “Subset access control” (FDP_ACC.1) is specified as follows.

FDP_ACC.1	**Subset access control**
Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1	The TSF shall enforce the *Memory Access Control Policy (MPU)* on *all subjects (software), all objects (data including code stored in memories) and all operations defined in the Memory Access Control Policy*.

The TOE Functional Requirement “Security attribute based access control” (FDP_ACF.1) is specified as follows.

FDP_ACF.1	**Security attribute based access control**
Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1	The TSF shall enforce the *Memory Access Control Policy (MPU)* to objects based on the following: *the memory area where the software is executed from and/or the memory area where the access is performed to and/or the operation to be performed*.
FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *evaluate the corresponding permission control information before, during or after the access so that accesses to be denied cannot be utilised by the subject attempting to perform the operation*.
FDP_ACF.1.3	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *None*.
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *None*.

The TOE Functional Requirement “Static attribute initialisation” (FMT_MSA.3) is specified as follows.

FMT_MSA.3	**Static attribute initialisation**
Hierarchical to:	No other components.
Dependencies:	FMT_MSA.1 Management of security attributes, FMT_SMR.1 Security roles
FMT_MSA.3.1	The TSF shall enforce the *Memory Access Control Policy (MPU)* to provide *restrictive* default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow ~~the~~ *any subject (provided that the Memory Access Control Policy is enforced and the necessary access is therefore allowed)* to specify alternative initial values to override the default values when an object or information is created.

The TOE Functional Requirement “Management of security attributes” (FMT_MSA.1) is specified as follows.

FMT_MSA.1	**Management of security attributes**
Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control], FMT_SMR.1 Security roles, FMT_SMF.1 Specification of Management Functions
FMT_MSA.1.1	The TSF shall enforce the *Memory Access Control Policy (MPU)* to restrict the ability to *change_default, modify or delete* the security attributes *read, write, execute* to *a software in a privileged mode (the trusted Operating System)*.

The TOE Functional Requirement “Specification of Management Functions” (FMT_SMF.1) is specified as follows.

FMT_SMF.1	**Specification of Management Functions**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FMT_SMF.1.1	The TSF shall be capable of performing the following management functions: *access to the control registers of the MPU*.

6.2 Security Assurance Requirements for the TOE

The Security Assurance Requirements for the TOE and its development and operating environment are those taken from EAL5 and augmented by the following components ALC_DVS.2 and AVA_VAN.5.

Class	Family	
ADV **Development**	ADV_ARC.1	Security architecture description
	ADV_FSP.5	Complete semi-formal functional specification with additional error information
	ADV_IMP.1	Implementation representation of the TSF
	ADV_INT.2	Well-structured internals
	ADV_TDS.4	Semiformal modular design
AGD **Guidance documents**	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
ALC **Life-cycle support**	ALC_CMC.4	Production support, acceptance procedures and automation
	ALC_CMS.5	Development tools CM coverage
	ALC_DEL.1	Delivery procedures
	ALC_DVS.2	Sufficiency of security measures
	ALC_LCD.1	Developer defined life-cycle model
	ALC_TAT.2	Compliance with implementation standards
ASE **Security Target Evaluation**	ASE_INT.1	Security target introduction
	ASE_CCL.1	Conformance claims
	ASE_SPD.1	Security problem definition
	ASE_OBJ.2	Security objectives
	ASE_ECD.1	Extended components definition
	ASE_REQ.2	Derived security requirements
	ASE_TSS.1	TOE summary specification
ATE **Tests**	ATE_COV.2	Analysis of coverage
	ATE_DPT.3	Analysis of coverage
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing – sample
AVA **Vulnerability assessment**	**AVA_VAN.5**	Advanced methodical vulnerability analysis

The Protection Profile BSI-CC-PP-0084-2014 [7] gives refinements of the TOE Assurance Requirements. Refer to the BSI-CC-PP-0084-2014 [7] for more details.

6.3 Security Requirements Rationale

6.3.1 Rationale for the security functional requirements

Table 10 below gives an overview, how the security functional requirements are combined to meet the security objectives. The detailed justification follows after the table.

Objective	TOE Security Functional and Assurance Requirements
O.Leak-Inherent	- FDP_ITT.1 “Basic internal transfer protection” - FPT_ITT.1 “Basic internal TSF data transfer protection” - FDP_IFC.1 “Subset information flow control”
O.Phys-Probing	- FDP_SDC.1 “Stored data confidentiality” - FPT_PHP.3 “Resistance to physical attack”
O.Malfunction	- FRU_FLT.2 “Limited fault tolerance - FPT_FLS.1 “Failure with preservation of secure state”
O.Phys-Manipulation	- FDP_SDI.2/RAM “Stored data integrity monitoring and action – RAM” - FDP_SDI.2/NVM “Stored data integrity monitoring and action – NVM” - FDP_SDI.2/Register&Bus “Stored data integrity monitoring and action – Register&Bus” - FPT_PHP.3 “Resistance to physical attack”
O.Leak-Forced	All requirements listed for O.Leak-Inherent - FDP_ITT.1, FPT_ITT.1, FDP_IFC.1 plus those listed for O.Malfunction and O.Phys-Manipulation - FRU_FLT.2, FPT_FLS.1, FPT_PHP.3
O.Abuse-Func	- FMT_LIM.1 “Limited capabilities” - FMT_LIM.2 “Limited availability” plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced - FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1
O.Identification	- FAU_SAS.1 “Audit storage”
O.RND	- FCS_RNG.1/PTG.2 “Quality metric for random numbers” plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced - FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1
O.Cap_Avail_Loader	- FMT_LIM.1/Loader “Limited capabilities” - FMT_LIM.2/Loader “Limited availability - Loader“
O.Authentication	- FIA_API.1 “Authentication Proof of Identity”
O.Ctrl_Auth_Loader	- FDP_UIT.1 “Data exchange integrity” - FDP_ACC.1/Loader “Subset access control – Loader” - FDP_ACF.1/Loader “Security Attribute based access control – Loader” - FTP_TRP.1 “Trusted path”
O.Prot_TSF_Confidentiality	- FDP_ACC.1/Loader “Subset access control – Loader” - FDP_ACF.1/Loader “Security Attribute based access control – Loader”

Objective	TOE Security Functional and Assurance Requirements
O.Mem-Access	- FDP_ACC.1 “Subset access control” - FDP_ACF.1 “Security Attribute based access control” - FMT_MSA.3 “Static attribute initialisation” - FMT_MSA.1 “Management of security attributes” - FMT_SMF.1 “Specification of Management Functions”
OE.Resp-Appl	Not Applicable.
OE.Process-Sec-IC	Not Applicable.
OE.Lim-Block-Loader	Not Applicable.
OE.TOE_Auth	Not Applicable.
OE.Loader_Usage	Not Applicable.

Table 10 : Security Requirements versus Security Objectives

Core PP

The justification related to the security objective “Protection against Inherent Information Leakage (**O.Leak-Inherent**)” is as follows:

The refinements of the security functional requirements **FPT_ITT.1** and **FDP_ITT.1** together with the policy statement in **FDP_IFC.1** explicitly require the prevention of disclosure of secret data (TSF data as well as user data) when transmitted between separate parts of the TOE or while being processed. This includes that attackers cannot reveal such data by measurements of emanations, power consumption or other behaviour of the TOE while data are transmitted between or processed by TOE parts.

It is possible that the TOE needs additional support by the Security IC Embedded Software (e.g. timing attacks are possible if the processing time of algorithms implemented in the software depends on the content of secret). This support must be addressed in the Guidance Documentation. Together with this **FPT_ITT.1**, **FDP_ITT.1** and **FDP_IFC.1** are suitable to meet the objective.

The justification related to the security objective “Protection against Physical Probing (**O.Phys-Probing**)” is as follows:

The SFR **FDP_SDC.1** requires the TSF to protect the confidentiality of the information of the user data stored in specified memory areas and prevent its compromise by physical attacks bypassing the specified interfaces for memory access. The scenario of physical probing as described for this objective is explicitly included in the assignment chosen for the physical tampering scenarios in **FPT_PHP.3**. Therefore, it is clear that this security functional requirement supports the objective.

It is possible that the TOE needs additional support by the Security IC Embedded Software (e.g. to send data over certain buses only with appropriate precautions). This support must be addressed in the Guidance Documentation. Together with this **FPT_PHP.3** is suitable to meet the objective.

The justification related to the security objective “Protection against Malfunctions (**O.Malfunction**)” is as follows:

The definition of this objective shows that it covers a situation, where malfunction of the TOE might be caused by the operating conditions of the TOE (while direct manipulation of the TOE is covered O.Phys-Manipulation). There are two possibilities in this situation: Either the operating conditions are

inside the tolerated range or at least one of them is outside of this range. The second case is covered by **FPT_FLS.1**, because it states that a secure state is preserved in this case. The first case is covered by **FRU_FLT.2** because it states that the TOE operates correctly under normal (tolerated) conditions. The functions implementing **FRU_FLT.2** and **FPT_FLS.1** must work independently so that their operation cannot affected by the Security IC Embedded Software (refer to the refinement). Therefore, there is no possible instance of conditions under O.Malfunction, which is not covered.

The justification related to the security objective "Protection against Physical Manipulation (**O.Phys-Manipulation**)" is as follows:

The SFR **FDP_SDI.2/RAM**, **FDP_SDI.2/NVM** and **FDP_SDI.2/Register&Bus** require the TSF to detect the integrity errors of the stored user data and react in case of detected errors. The scenario of physical manipulation as described for this objective is explicitly included in the assignment chosen for the physical tampering scenarios in **FPT_PHP.3**. Therefore, it is clear that this security functional requirement supports the objective.

The justification related to the security objective "Protection against Forced Information Leakage (**O.Leak-Forced**)" is as follows:

This objective is directed against attacks, where an attacker wants to force an information leakage, which would not occur under normal conditions. In order to achieve this the attacker has to combine a first attack step, which modifies the behaviour of the TOE (either by exposing it to extreme operating conditions or by directly manipulating it) with a second attack step measuring and analysing some output produced by the TOE. The first step is prevented by the same mechanisms which support **O.Malfunction** and **O.Phys-Manipulation**, respectively. The requirements covering **O.Leak-Inherent** also support O.Leak-Forced because they prevent the attacker from being successful if he tries the second step directly.

The justification related to the security objective "Protection against Abuse of Functionality (**O.Abuse-Func**)" is as follows:

This objective states that abuse of functions (especially provided by the IC Dedicated Test Software, for instance in order to read secret data) must not be possible in Phase 7 of the life-cycle. There are two possibilities to achieve this: (i) They cannot be used by an attacker (i. e. its availability is limited) or (ii) using them would not be of relevant use for an attacker (i. e. its capabilities are limited) since the functions are designed in a specific way. The first possibility is specified by **FMT_LIM.2** and the second one by **FMT_LIM.1**. Since these requirements are combined to support the policy, which is suitable to fulfil O.Abuse-Func, both security functional requirements together are suitable to meet the objective.

Other security functional requirements which prevent attackers from circumventing the functions implementing these two security functional requirements (for instance by manipulating the hardware) also support the objective. The relevant objectives are also listed in Table 9.

It was chosen to define **FMT_LIM.1** and **FMT_LIM.2** explicitly (not using Part 2 of the Common Criteria) for the following reason: Though taking components from the Common Criteria catalogue makes it easier to recognise functions, any selection from Part 2 of the Common Criteria would have made it harder for the reader to understand the special situation meant here. As a consequence, the statement of explicit security functional requirements was chosen to provide more clarity.

The justification related to the security objective “TOE Identification (**O.Identification**)“ is as follows:

Obviously the operations for **FAU_SAS.1** are chosen in a way that they require the TOE to provide the functionality needed for O.Identification. The Initialisation Data (or parts of them) are used for TOE identification. The technical capability of the TOE to store Initialisation Data and/or Pre-personalisation Data is provided according to FAU_SAS.1.

It was chosen to define **FAU_SAS.1** explicitly (not using a given security functional requirement from Part 2 of the Common Criteria) for the following reason: The security functional requirement FAU_GEN.1 in Part 2 of the CC requires the TOE to generate the audit data and gives details on the content of the audit records (for instance data and time). The possibility to use the functions in order to store security relevant data which are generated outside of the TOE, is not covered by the family FAU_GEN or by other families in Part 2. Moreover, the TOE cannot add time information to the records, because it has no real time clock. Therefore, the new family FAU_SAS was defined for this situation.

The justification related to the security objective “Random Numbers (**O.RND**)” is as follows:

FCS_RNG.1/PTG.2 requires the TOE to provide random numbers of good quality. To specify the exact metric is left to the individual Security Target for a specific TOE.

Other security functional requirements, which prevent physical manipulation and malfunction of the TOE (see the corresponding objectives listed in the table) support this objective because they prevent attackers from manipulating or otherwise affecting the random number generator.

Random numbers are often used by the Security IC Embedded Software to generate cryptographic keys for internal use. Therefore, the TOE must prevent the unauthorised disclosure of random numbers. Other security functional requirements which prevent inherent leakage attacks, probing and forced leakage attacks ensure the confidentiality of the random numbers provided by the TOE.

Depending on the functionality of specific TOEs the Security IC Embedded Software will have to support the objective by providing runtime-tests of the random number generator. Together, these requirements allow the TOE to provide cryptographically good random numbers and to ensure that no information about the produced random numbers is available to an attacker.

It was chosen to define **FCS_RNG.1** explicitly, because Part 2 of the Common Criteria do not contain generic security functional requirements for Random Number generation. (Note, that there are security functional requirements in Part 2 of the Common Criteria, which refer to random numbers. However, they define requirements only for the authentication context, which is only one of the possible applications of random numbers.)

Package “Authentication of the Security IC”

The justification related to the security objective “Authentication to external entities (**O.Authentication**)” is as follows:

The security objective “Authentication to external entities (O.Authentication) is directly covered by the SFR **FIA_API.1**.

Package 1: Loader dedicated for usage in secured environment only

The security objective “Capability and availability of the Loader (**O.Cap_Avail_Loader**) is directly covered by the SFR **FMT_LIM.1/Loader** and **FMT_LIM.2/Loader**.

Package 2 Lite: Loader dedicated for usage by authorized users only (Part)

The security objective Access control and authenticity for the Loader (**O.Ctrl_Auth_Loader**) is covered by the SFR as follows:

- The SFR **FDP_ACC.1/Loader** defines the subjects, objects and operations of the Loader SFP enforced by the SFR **FDP_UIT.1** and **FDP_ACF.1/Loader**.
- The SFR **FDP_UIT.1** requires the TSF to verify the integrity of the received user data.
- The SFR **FDP_ACF.1/Loader** requires the TSF to implement access control for the Loader functionality.
- The SFR **FTP_TRP.1** requires the TSF to establish a communication path with assured identification of its end points and protection of the communication data from modification.

Additional security objectives for the TOE (provided by [19] and [20])

The security objective “Protection of the confidentiality of the TSF” (**O.Prot_TSF_Confidentiality**) is directly covered by the SFR **FDP_ACC.1/Loader** and **FDP_ACF.1/Loader** which requires the TSF to implement access control for the Loader functionality. The user must be successfully authenticated before having access to the TOE.

The justification related to the security objective “Area based Memory Access Control (**O.Mem-Access**)” is as follows:

The security functional requirement “Subset access control (**FDP_ACC.1**)” with the related Security Function Policy (SFP) “Memory Access Control Policy” exactly require to implement an area based memory access control as demanded by O.Mem-Access. Therefore, FDP_ACC.1 with its SFP is suitable to meet the security objective.

Nevertheless, the developer of the Smartcard Embedded Software must ensure that the additional functions are used as specified and that the User Data processed by these functions are protected as defined for the application context.

The security functional requirement “Security Attribute based access control (**FDP_ACF.1**) with the related Security Function Policy (SFP) “Memory Access Control Policy” addresses security attributes usage and characteristics of policies. It describes the rules for the function that implements the Security Function Policy (SFP) as identified in FDP_ACC.1. Therefore, FDP_ACF.1 with its SFP is suitable to meet the security objective.

The security functional requirement “Static attribute initialisation (**FMT_MSA.3**)” requires that the TOE provides default values for security attributes. Since the TOE is a hardware platform these default values are generated by the reset procedure. Therefore FMT_MSA.3 is suitable to meet the security objective O.Mem-Access.

The security functional requirement “Management of security attributes **(FMT_MSA.1**)” requires that the ability to change the security attributes is restricted to privileged subject(s). It ensures that the access control required by O.Mem-Access can be realized using the functions provided by the TOE. Therefore FMT_MSA.1 is suitable to meet the security objective O.Mem-Access.

Finally, the security functional requirement “Specification of Management Functions (**FMT_SMF.1**)” is used for the specification of the management functions to be provided by the TOE as required by O.Mem_Access. Therefore, FMT_SMF.1 is suitable to meet the security objective O.Mem-Access.

6.3.2 Dependencies of security functional requirements

Table 11 below lists the security functional requirements defined in this Security Target, their dependencies and whether they are satisfied by other security requirements defined in this Security Target. The text following the table discusses the remaining cases.

Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
FRU_FLT.2	FPT_FLS.1	Yes
FPT_FLS.1	None	No dependency
FMT_LIM.1	FMT_LIM.2	Yes
FMT_LIM.2	FMT_LIM.1	Yes
FAU_SAS.1	None	No dependency
FPT_PHP.3	None	No dependency
FDP_ITT.1	[FDP_ACC.1 or FDP_IFC.1]	Yes (FDP_IFC.1)
FDP_IFC.1	FDP_IFF.1	See discussion below
FPT_ITT.1	None	No dependency
FDP_SDC.1	None	No dependency
FDP_SDI.2/RAM	None	No dependency
FDP_SDI.2/NVM	None	No dependency
FDP_SDI.2/Register&Bus	None	No dependency
FCS_RNG.1/PTG.2	None	No dependency
FMT_LIM.1/Loader	FMT_LIM.2	Yes (FMT_LIM.2/Loader)
FMT_LIM.2/Loader	FMT_LIM.1	Yes (FMT_LIM.2/Loader))
FIA_API.1	None	No dependency
FDP_UIT.1	[FTP_ITC.1 or FTP_TRP.1] [FDP_ACC.1 or FDP_IFC.1]	Yes (FTP_TRP.1) Yes (FDP_ACC.1/Loader)
FDP_ACC.1/Loader	FDP_ACF.1	Yes (FDP_ACF.1/Loader)
FDP_ACF.1/Loader	FMT_MSA.3	See discussion below
FTP_TRP.1	None	No dependency
FDP_ACC.1	FDP_ACF.1	Yes.
FDP_ACF.1	FDP_ACC.1 FMT_MSA.3	Yes. Yes.
FMT_MSA.1	[FDP_ACC.1 or FDP_ITC.1] FMT_SMR.1 FMT_SMF.1	FDP_ACC.1. See discussion bellow. Yes.
FMT_MSA.3	FMT_MSA.1 FMT_SMR.1	Yes. See discussion bellow.

Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
FMT_SMF.1	None	No dependency.

Table 11 : Dependencies of Security Functional Requirements

Part 2 of the Common Criteria defines the dependency of **FDP_IFC.1** (information flow control policy statement) on FDP_IFF.1 (Simple security attributes). The specification of FDP_IFF.1 would not capture the nature of the security functional requirement nor add any detail. As stated in the Data Processing Policy referred to in FDP_IFC.1 there are no attributes necessary. The security functional requirement for the TOE is sufficiently described using FDP_ITT.1 and its Data Processing Policy (FDP_IFC.1).

As Table 11 shows, all other dependencies of functional requirements are fulfilled by security requirements defined in this Security Target.

The discussion in Section 6.3.1 has shown, how the security functional requirements support each other in meeting the security objectives of this Security Target. In particular the security functional requirements providing resistance of the hardware against manipulations (e. g. FPT_PHP.3) support all other more specific security functional requirements (e. g. FCS_RNG.1) because they prevent an attacker from disabling or circumventing the latter.

The dependency of **FDP_ACF.1/Loader** on FMT_MSA.3 isn't necessary because the security attributes used to enforce the Loader SFP are fixed by the IC manufacturer and no new objects under control of the Loader SFP are created.

The dependency FMT_SMR.1 introduced by the two components **FMT_MSA.1** and **FMT_MSA.3** is considered to be satisfied because the access control specified for the intended TOE is not role-based but enforced for each subject. Therefore, there is no need to identify roles in form of a security functional requirement FMT_SMR.1.

6.3.3 Rationale for the Assurance Requirements

The assurance level EAL5 and the augmentation with the requirements ALC_DVS.2, and AVA_VAN.5 were chosen in order to meet assurance expectations explained in the following paragraphs.

EAL5

An assurance level of EAL5 with the augmentations AVA_VAN.5 and ALC_DVS.2 are required for this type of TOE since it is intended to defend against sophisticated attacks.

This evaluation assurance package was selected to permit a developer to gain maximum assurance from positive security engineering based on good commercial practices. In order to provide a meaningful level of assurance that the TOE provides an adequate level of defence against such attacks, the evaluators should have access to the low level design and source code.

ALC_DVS.2 Sufficiency of security measures

Development security is concerned with physical, procedural, personnel and other technical measures that may be used in the development environment to protect the TOE.

In the particular case of a Security IC the TOE is developed and produced within a complex and distributed industrial process which must especially be protected. Details about the implementation, (e.g. from design, test and development tools as well as Initialisation Data) may make such attacks easier. Therefore, in the case of a Security IC, maintaining the confidentiality of the design is very important.

This assurance component is a higher hierarchical component to EAL5 (which only requires ALC_DVS.1). ALC_DVS.2 has no dependencies.

AVA_VAN.5 Advanced methodical vulnerability analysis

Due to the intended use of the TOE, it must be shown to be highly resistant to penetration attacks. This assurance requirement is achieved by the AVA_VAN.5 component.

Independent vulnerability analysis is based on highly detailed technical information. The main intent of the evaluator analysis is to determine that the TOE is resistant to penetration attacks performed by an attacker possessing high attack potential.

AVA_VAN.5 has dependencies to ADV_ARC.1 “Security architecture description”, ADV_FSP.4 “Complete functional specification”, ADV_IMP.1 “Implementation representation of the TSF”, ADV_TDS.3 “Basic modular design”, AGD_OPE.1 “Operational user guidance”, AGD_PRE.1 “Preparative procedures” and ATE_DPT.1 “Testing: basic design”.

All these dependencies are satisfied by EAL5.

It has to be assumed that attackers with high attack potential try to attack Security ICs like smart cards used for digital signature applications or payment systems. Therefore, specifically AVA_VAN.5 was chosen in order to assure that even these attackers cannot successfully attack the TOE.

Note that detailed refinements for assurance requirements are given in Section 6.2.1 of PP [7].

6.3.4 Security Requirements are Internally Consistent

The discussion of security functional requirements and assurance components in the preceding sections has shown that consistency are given for both groups of requirements. The arguments given for the fact that the assurance components are adequate for the functionality of the TOE also shows that the security functional requirements and assurance requirements support each other and that there are no inconsistencies between these groups.

The security functional requirements FDP_SDC.1 and FDP_SDI.2/RAM, FDP_SDI.2/NVM and FDP_SDI.2/Register&Bus address the protection of user data in the specified memory areas against compromise and manipulation. The security functional requirement FPT_PHP.3 makes it harder to manipulate data. This protects the primary assets identified in Section 3.1 and other security features or functionality which use these data.

Though a manipulation of the TOE (refer to FPT_PHP.3) is not of great value for an attacker in itself, it can be an important step in order to threaten the primary assets identified in Section 3.1. Therefore, the security functional requirement FPT_PHP.3 is not only required to meet the security objective O.Phys-Manipulation. Instead it protects other security features or functions of both the TOE and the Security IC Embedded Software from being bypassed, deactivated or changed. In particular this may pertain to the security features or functions being specified using FDP_ITT.1, FPT_ITT.1, FPT_FLS.1, FMT_LIM.2, FCS_RNG.1/PTG.2, and those implemented in the Security IC Embedded Software.

A malfunction of TSF (refer to FRU_FLT.2 and FPT_FLS.1) can be an important step in order to threaten the primary assets identified in Section 3.1. Therefore, the security functional requirements FRU_FLT.2 and FPT_FLS.1 are not only required to meet the security objective O.Malfunction. Instead they protect other security features or functions of both the TOE and the Security IC Embedded Software from being bypassed, deactivated or changed. In particular this pertains to the security features or functions being specified using FDP_ITT.1, FPT_ITT.1, FMT_LIM.1, FMT_LIM.2, FCS_RNG.1/PTG.2, and those implemented in the Security IC Embedded Software.

In a forced leakage attack the 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 which normally do not contain significant information about secrets. Therefore, in order to avert the disclosure of primary assets identified in Section 3.1 it is important that the security functional requirements averting leakage (FDP_ITT.1, FPT_ITT.1) and those against malfunction (FRU_FLT.2 and FPT_FLS.1) and physical manipulation (FPT_PHP.3) are effective and bind well. The security features and functions against malfunction ensure correct operation of other security functions (refer to above) and help to avert forced leakage themselves in other attack scenarios. The security features and functions against physical manipulation make it harder to manipulate the other security functions (refer to above).

Physical probing (refer to FPT_PHP.3) shall directly avert the disclosure of primary assets identified in Section 3.1. In addition, physical probing can be an important step in other attack scenarios if the corresponding security features or functions use secret data. For instance the security functional requirement FMT_LIM.2 may use passwords. Therefore, the security functional requirement FPT_PHP.3 (against probing) help to protect other security features or functions including those being implemented in the Security IC Embedded Software. Details depend on the implementation.

Leakage (refer to FDP_ITT.1, FPT_ITT.1) shall directly avert the disclosure of primary assets identified in Section 3.1. In addition, inherent leakage and forced leakage (refer to above) can be an important step in other attack scenarios if the corresponding security features or functions use secret data. For instance the security functional requirement FMT_LIM.2 may use passwords. Therefore, the security functional requirements FDP_ITT.1 and FPT_ITT.1 help to protect other security features or functions implemented in the Security IC Embedded Software (FDP_ITT.1) or provided by the TOE (FPT_ITT.1). Details depend on the implementation.

The user data of the Composite TOE are treated as required to meet the requirements defined for the specific application context (refer to Treatment of user data of the Composite TOE (A.Resp-Appl)). However, the TOE may implement additional functions. This can be a risk if their interface cannot completely be controlled by the Security IC Embedded Software. Therefore, the security functional requirements FMT_LIM.1 and FMT_LIM.2 are very important. They ensure that appropriate control is applied to the interface of these functions (limited availability) and that these functions, if being usable, provide limited capabilities only.

The combination of the security functional requirements FMT_LIM.1 and FMT_LIM.2 ensures that (especially after TOE Delivery) these additional functions cannot be abused by an attacker to (i) disclose or manipulate user data of the Composite TOE, (ii) to manipulate (explore, bypass, deactivate or change) security features or services of the TOE or of the Security IC Embedded Software or (iii) to enable other attacks on the assets. Hereby the binding between these two security functional requirements is very important.

The security functional requirement Limited Capabilities (FMT_LIM.1) must close gaps which could be left by the control being applied to the function's interface (Limited Availability (FMT_LIM.2)). Note that the security feature or services which limits the availability can be bypassed, deactivated or changed by physical manipulation or a malfunction caused by an attacker. Therefore, if Limited Availability (FMT_LIM.2) is vulnerable, it is important to limit the capabilities of the functions in order to limit the possible benefit for an attacker.

The security functional requirement Limited Availability (FMT_LIM.2) must close gaps which could result from the fact that the function's kernel in principle would allow to perform attacks. The TOE must limit the availability of functions which potentially provide the capability to disclose or manipulate user data of the Composite TOE, to manipulate security features or services of the TOE or of the Security IC Embedded Software or to enable other attacks on the assets. Therefore, if an attacker could benefit from using such functions, it is important to limit their availability so that an attacker is not able to use them.

No perfect solution to limit the capabilities (FMT_LIM.1) is required if the limited availability (FMT_LIM.2) alone can prevent the abuse of functions. No perfect solution to limit the availability (FMT_LIM.2) is required if the limited capabilities (FMT_LIM.1) alone can prevent the abuse of functions. Therefore, it is correct that both requirements are defined in a way that they together provide sufficient security.

It is important to avert malfunctions of TSF and of security functions implemented in the Security IC Embedded Software (refer to above). There are two security functional requirements which ensure that malfunctions cannot be caused by exposing the TOE to environmental stress. First it must be ensured that the TOE operates correctly within some limits (Limited fault tolerance (FRU_FLT.2)). Second the TOE must prevent its operation outside these limits (Failure with preservation of secure state (FPT_FLS.1)). Both security functional requirements together prevent malfunctions. The two functional requirements must define the "limits". Otherwise there could be some range of operating conditions which is not covered so that malfunctions may occur. Consequently, the security functional requirements Limited fault tolerance (FRU_FLT.2) and Failure with preservation of secure state (FPT_FLS.1) are defined in a way that they together provide sufficient security.

The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced also protect the area based memory access control function implemented according to the security functional requirement described in the security functional requirement FDP_ACC.1 with reference to the Memory Access Control Policy and details given in FDP_ACF.1. Therefore, those security functional requirements support the secure implementation and operation of FDP_ACF.1 with its dependent security functional requirements.

7 TOE Summary Specification

7.1 Description of TSF features

The product overview is given in section 1.2. In the following the Security Mechanism are described and the relation to the security functional requirements is shown. The TOE is equipped with following Security Features to meet the security functional requirements:

7.1.1 SF_PMODE: Product Mode

The TSF implements several mode during the life cycle of the product.

- Product mode — Mechanism to manage the different step of the product life cycle. At each step, register, data and memories access are limited or not. This allows to restrict access at person using the product in each step from manufacturing step to final user. In addition, it is not possible to come back to test mode after the deployment of the product.

7.1.2 SF_IDENT: Identification

The TSF implements unique identification of the product.

- Unique Id — A unique identification of the product is stored in the One-Time-Programmable Memory (part of NVM Flash).
- Authentication — Mechanism of identification by authentication of the TOE based on cryptographic authentication. This prevent masquerade and improve the security during transport.

7.1.3 SF_CONF&INT: Confidentiality & Integrity

The TSF implements protection to keep confidentiality and integrity of data in register, memory and bus such as:

- Bus Encryption — Mechanism to mask the data on the bus. This guarantees the confidentiality of these data moving on the bus. This prevents leakage of these data.
- Register Masking — Mechanism to add confidentiality on the data in the register. This prevents leakage of these data.
- Memory & bus & register Integrity — Mechanism to add integrity on the memory and integrity from the memory to register. This prevents to modify the value of the data in RAM, in the bus and in the register. This guaranties a correct execution of the data.

- Memories Encryption — Mechanism to encrypt memories content. This brings confidentiality of the data stored in memories. This prevents to directly know the value of the data in case of reverse, probing and extraction.

7.1.4 SF_SCRA: Scrambling

The TSF implements protections against localization of the data in the product such as:

- Address Scrambling — Mechanism to scramble the addresses. CPU address is translated in physical address via a translation mechanism. This brings complexity to localize data in the memories.

7.1.5 SF_EXEC: Correct Execution

The TSF implements protection against the un-correct execution of the code such as:

- Anti Re-routing — Mechanism to detect code re-routing. Alarm is sent in case of detection. See SF_ALARM.
- Illegal opcode — Mechanism to detect illegal opcode execution. This prevents re-routing of the product. Alarm is sent in case of detection. See SF_ALARM.
- MPU — Mechanism to define access permission on different memory areas. Each areas can have an attribute read, write, execution. This prevents access to illegal memory area during operating condition and protect these memory areas.

7.1.6 SF_EM: Environment Control

The TSF implements protection against tentative of modification of the product and disturbing of environment such as:

- Active Shield: — Active mechanism to detect tentative of physical intrusion in the product (FIB…). If tentative of physical manipulation or physical probing are carried out on the product, active shield shall detect that. This prevents to modify the product and reverse it.
- Environment Sensors Monitoring — Mechanism to control the correct operating conditions.
  If the operating environment is not in the range expected by the chip manufacturer, the appropriate embedded analog sensors shall detect external perturbations and the out of range operating conditions. This prevents to put the circuit in an uncontrolled state.

7.1.7 SF_ALARM: Alarm Management

The TSF implements mechanisms to send alarm such as:

- Alarm management — Mechanism to configure alarm, either IT or HW reset. Certain Alarms are hardcoded, other can have a chosen behaviour.

7.1.8 SF_RANDOM: Randomization

The TSF implements mechanisms brought randomization of the execution such as:

- Randomized Synchronization — Mechanism to generate randomization in the synchronization of the system. This mechanism makes the execution timing unpredictable to add complexity to synchronize attacks and to observe side channel leakage.

7.1.9 SF_RNG: Random Number Generator

The TSF implements mechanisms providing random number such as:

- RNG Generator — Mechanism to generate random number. Production of random number is controlled and the quality of the random value is evaluated. Random number (PTRNG) is used for key generation or for security measure. It is compliant AIS31. A second internal DRNG will be used as well.

7.1.10 SF_DE: Design

The TSF implements mechanism to protect the design of the product such as:

- Layout — Mechanism added in the layout. Certain net are not routed in the Top, redundancy is added for certain signal. This adds complexity in the reverse engineering.

7.1.11 SF_LOAD: Loader

The TSF implements mechanism to load secure code in the product such as:

- Secure Loading — Mechanism to allow loading in the product in a secure way and mechanism to block the loading mechanism.

7.1.12 SF_NORMAL_EXEC: Control of Operating Conditions

The TSF implements mechanisms to control the correct operating conditions of the product such as:

Control of Operating Conditions: Mechanisms to ensure the correct operating conditions of the product and to prevent any malfunction using the following mechanisms:

- filters on clock and on supply voltage;
- integrity check of sensitive data on boot.

The sensors triggering occurs before the sensors functionality limits.

7.2 Rationale for TSF

The justification and overview of the mapping between security functional requirements (SFR) and the TOE's security functionality (SF) is given in section above.

SFR / SF	SF_PMODE	SF_IDENT	SF_CONF&INT	SF_SCRA	SF_EXEC	SF_EM	SF_ALARM	SF_RANDOM	SF_RNG	SF_DE	SF_LOAD	SF_NORMAL_EXEC
FRU_FLT.2					X							X
FPT_FLS.1					X	X	X					X
FMT_LIM.1	X											
FMT_LIM.2	X											
FAU_SAS.1	X	X										
FPT_PHP.3			X	X		X				X		
FDP_ITT.1			X	X						X		
FDP_IFC.1			X	X	X			X				
FPT_ITT.1			X	X						X		
FDP_SDC.1	X		X	X	X	X						
FDP_SDI.2/RAM			X				X					
FDP_SDI.2/NVM			X				X					
FDP_SDI.2/Register&Bus			X				X					
FCS_RNG.1/PTG.2									X			
FMT_LIM.1/Loader											X	
FMT_LIM.2/Loader											X	
FIA_API.1		X										
FDP_UIT.1											X	
FDP_ACC.1/Loader		X									X	
FDP_ACF.1/Loader		X									X	
FTP_TRP.1											X	
FDP_ACC.1	X				X							
FDP_ACF.1	X				X							
FMT_MSA.1					X							
FMT_MSA.3					X							
FMT_SMF.1					X							

Table 12: Mapping SFR & SF

8 Glossary

Application Data	All data managed by the Security IC Embedded Software in the application context. Application data comprise all data in the final Security IC.
Authentication reference data	Data used to verify the claimed identity in an authentication procedure.
Authentication verification data	Data used to prove the claimed identity in an authentication procedure.
Composite Product Integrator	Role installing or finalising the IC Embedded Software and the applications on platform transforming the TOE into the unpersonalised Composite Product after TOE delivery. The TOE Manufacturer may implement IC Embedded Software delivered by the Security IC Embedded Software Developer before TOE delivery (e.g. if the IC Embedded Software is implemented in ROM or is stored in the non-volatile memory as service provided by the IC Manufacturer or IC Packaging Manufacturer).
Composite Product Manufacturer	The Composite Product Manufacturer has the following roles (i) the Security IC Embedded Software Developer (Phase 1), (ii) the Composite Product Integrator (Phase 5) and (iii) the Personaliser (Phase 6). If the TOE is delivered after Phase 3 in form of wafers or sawn wafers (dice) he has the role of the IC Packaging Manufacturer (Phase 4) in addition. The customer of the TOE Manufacturer who receives the TOE during TOE Delivery. The Composite Product Manufacturer includes the Security IC Embedded Software developer and all roles after TOE Delivery up to Phase 6 (refer to Figure 2 on page 11 and Section 7.1.1 of the PP).
End-consumer	User of the Composite Product in Phase 7.
IC Dedicated Software	IC proprietary software embedded in a Security IC (also known as IC firmware) and developed by the IC Developer. Such software is required for testing purpose (IC Dedicated Test Software) but may provide additional services to facilitate usage of the hardware and/or to provide additional services (IC Dedicated Support Software).
IC Dedicated Test Software	That part of the IC Dedicated Software (refer to above) which is used to test the TOE before TOE Delivery but which does not provide any functionality thereafter.

IC Dedicated Support Software	That part of the IC Dedicated Software (refer to above) which provides functions after TOE Delivery. The usage of parts of the IC Dedicated Software might be restricted to certain phases.
Initialisation Data	Initialisation Data defined by the TOE Manufacturer to identify the TOE and to keep track of the Security IC's production and further life-cycle phases are considered as belonging to the TSF data. These data are for instance used for traceability and for TOE identification (identification data). If "Package Authentication of the Security IC" is used the Initialisation data contain the confidential authentication verification data of the IC. If the "Package 2: Loader dedicated for usage by authorized users only" may contain the authentication verification data or key material for the trusted channel between the TOE and the authorized users using the Loader.
Integrated Circuit (IC)	Electronic component(s) designed to perform processing and/or memory functions.
Pre-personalisation Data	Any data supplied by the Card Manufacturer that is injected into the non-volatile memory by the Integrated Circuits manufacturer (Phase 3). These data are for instance used for traceability and/or to secure shipment between phases. If "Package 2: Loader dedicated for usage by authorized users only" is used the Pre-personalisation Data may contain the authentication reference data or key material for the trusted channel between the TOE and the authorized users using the Loader.
Security IC	(as used in this Security Target) Composition of the TOE, the Security IC Embedded Software, user data of the Composite TOE and the package (the Security IC carrier).
Security IC Embedded Software	Software embedded in a Security IC and normally not being developed by the IC Designer. The Security IC Embedded Software is designed in Phase 1 and embedded into the Security IC in Phase 3 or in later phases of the Security IC product life-cycle. Some part of that software may actually implement a Security IC application others may provide standard services. Nevertheless, this distinction doesn't matter here so that the Security IC Embedded Software can be considered as being application dependent whereas the IC Dedicated Software is definitely not.
Security IC Product	Composite product which includes the Security Integrated Circuit (i.e. the TOE) and the Embedded Software and is evaluated as composite target of evaluation in the sense of the Supporting Document

Secured Environment	Operational environment maintains the confidentiality and integrity of the TOE as addressed by OE.Process-Sec-IC and the confidentiality and integrity of the IC Embedded Software, TSF data or user data associated with the smartcard product by security procedures of the smartcard product manufacturer, personaliser and other actors before delivery to the smartcard end-user depending on the smartcard life-cycle.
Test Features	All features and functions (implemented by the IC Dedicated Test Software and/or hardware) which are designed to be used before TOE Delivery only and delivered as part of the TOE.
TOE Delivery	The period when the TOE is delivered which is (refer to Figure 2 on page 11) either (i) after Phase 3 (or before Phase 4) if the TOE is delivered in form of wafers or sawn wafers (dice) or (ii) after Phase 4 (or before Phase 5) if the TOE is delivered in form of packaged products.
TOE Manufacturer	The TOE Manufacturer must ensure that all requirements for the TOE (as defined in Section 1.2.2) and its development and production environment are fulfilled (refer to Figure 2 on page 11). The TOE Manufacturer has the following roles: (i) IC Developer (Phase 2) and (ii) IC Manufacturer (Phase 3). If the TOE is delivered after Phase 4 in form of packaged products, he has the role of the (iii) IC Packaging Manufacturer (Phase 4) in addition.
TSF data	Data for the operation of the TOE upon which the enforcement of the SFR relies. They are created by and for the TOE, that might affect the operation of the TOE. This includes information about the TOE's configuration, if any is coded in non-volatile non-programmable memories (ROM), in non-volatile programmable memories (for instance EEPROM or flash memory), in specific circuitry or a combination thereof.
User data of the Composite TOE	All data managed by the Smartcard Embedded Software in the application context.
User data of the TOE	Data for the user of the TOE, that does not affect the operation of the TSF. From the point of view of TOE defined in this ST the user data comprises the Security IC Embedded Software and the user data of the Composite TOE.