IFX_CCI_000Dh, IFX_CCI_0020h, IFX_CCI_0031h, IFX_CCI_0032h, IFX_CCI_0034h, IFX_CCI_0037h T31 and M31
Security Target Lite

Revision: 5.4

Table of Contents

Contents

1 Security Target Introduction (ASE_INT)

1.1 ST reference

The Security Target Lite has the revision 5.4 and is dated 2025-08-14. The title of this document is IFX_CCI_000Dh, IFX_CCI_0020h, IFX_CCI_0031h, IFX_CCI_0032h, IFX_CCI_0034h, IFX_CCI_0037h T31 and M31 Security Target Lite.

1.2 TOE Reference

The Security Target Lite comprises an Infineon Technologies Security Controller named IFX_CCI_000Dh, IFX_CCI_0020h, IFX_CCI_0031h, IFX_CCI_0032h, IFX_CCI_0034h, IFX_CCI_0037h design step T31 and M31 with optional HSL v2.62.7626, optional SCL version v2.04.003, UMSLC lib v01.00.0234 with specific IC-dedicated firmware identifier 80.301.05.1 and user guidance in the following called TOE (Target of evaluation).

The Security Target Lite is based on the Protection Profile “Smartcard IC Platform Protection Profile” [1].

The Protection Profile is built in compliance to Common Criteria v3.1. The Security Target Lite is conformant to CC:2022.

The targeted assurance level is EAL5+.

Table 1 Identification

Hardware	Version	Method of identification
IFX_CCI_000Dh, IFX_CCI_0020h, IFX_CCI_0031h, IFX_CCI_0032h, IFX_CCI_0034h, IFX_CCI_0037h (each of the comma separated term is a Common Criteria Certification Identifier)	T31 and M31 (design step)	Non-ISO ATR
firmware		
BOS	80.301.05.1	Non-ISO ATR: firmware identifier
Flash-loader	v8.07.006	Flash-loader function
Software		
HSL	v2.62.7626	HSL function
UMSLC	v01.00.0234	UMSLC function
SCL	v2.04.003	SCL function
User Guidance		
32-bit Security Controller – V02, Hardware Reference Manual	V9.2, 2020-02-06	document
32-bit ARM-based Security Controller, SLC 37/40-nm Technology, Programmer's Reference Manual	V4.6, 2020-10-13	document
32-bit Security Controller – V02, Security Guidelines	v1.00-3053, 2025-02-14	document
Production and personalization 32-bit ARM-based security controller	v.3.6, 2024-12-16	document
HSL library for SLCx7 in 40nm	v02.62.7626, Rev. 1.2, 2020-12-17	document
UMSLC library for SLCx7 in 40nm, Version 01.00.0234	V1.1, 2018-05-23	document
SCL37-uSCP-v3-C40 Symmetric Crypto Library for uSCP-v3 DES /AES	v2.04.003, 2025-06-17	document

A customer shall identify the TOE. The TOE hardware and its configuration (for details see chapter 1.4.7) using the Non-ISO ATR. The Non-ISO ATR outputs a Common Criteria Certification Identifier and firmware identifier, which links the TOE to this ST. Specific firmware functions can be used to determine the exact configuration of a device from the certified range defined in Table 3

1.3 TOE Overview

1.3.1 TOE Definition and Usage

The TOE consists of smart card ICs (Security Controllers), firmware and user guidance meeting high requirements in terms of performance and security designed by Infineon Technologies AG. This TOE is intended to be used in smart cards for security-relevant applications and as developing platform for smart card operating systems according to the lifecycle model from [1]

The term Smartcard Embedded Software is used in the following for all operating systems and applications stored and executed on the TOE. The TOE is the platform for the Smartcard Embedded Software. The Smartcard Embedded Software itself is not part of the TOE. The TOE does not require any non-TOE hardware/software/firmware.

1.3.2 TOE major security features

- Cryptographic support: TDES, AES, RNG (PTG.2 according to [6])
- Memory protection unit supporting different memory access levels
- Memory encryption
- Robust set of sensors and detectors for the purpose of monitoring proper chip operating conditions
- Redundant alarm propagation and system deactivation principle
- Register protection
- Security life control
- Program flow integrity protection
- Peripheral access control
- Bus encryption for security peripherals
- Tearing safe NVM programming
- Security optimized wiring
- Leakage control of data dependent code execution
- Device phase management supporting isolation of test features and flash loader accessibility
- Detection of NVM single and multi bit errors

1.4 TOE description

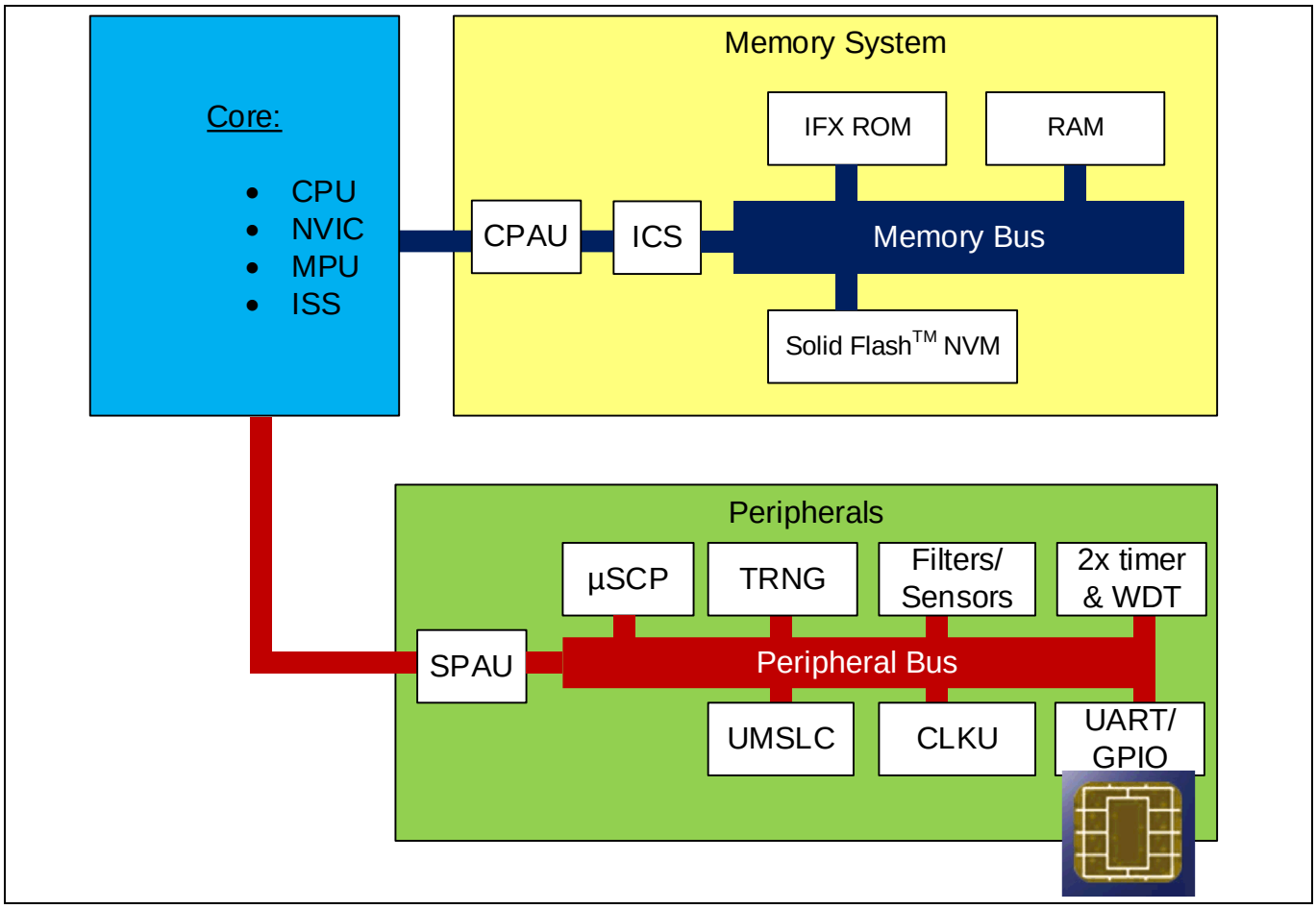
1.4.1 TOE components

1.4.1.1 Hardware components

Figure 1 shows a block diagram of the TOE hardware:

Figure 1 Block diagram of TOE hardware

Security Target Introduction (ASE_INT)



The TOE hardware consists of a core, a memory system and peripherals.

The major components of the core system are a 32-bit CPU (Central Processing Unit), an MPU (Memory Protection Unit), a Nested Vectored Interrupt Controller (NVIC) and an Instruction Stream Signature Checking (ISS).

The MPU of the core stores code and data in a linear 4-GByte memory space (32-bit range), allowing direct access without the need to swap memory segments in and out of memory using a memory protection unit.

There are two separate bus entities: a memory bus and a peripheral bus for high-speed communication with the peripherals.

The SPAU can be configured by the user to block or allow peripheral access. It can also be used to block RAM areas (For keeping Figure 1 simple, the connection between SPAU and RAM is not shown). The CPAU enables the user to block or allow unprivileged level access to NVM and specific registers of ICS and NVM.

The CPU accesses memory via the Internal Ciphering System (ICS), which encrypts/decrypts memory content. All data of the memory block is encrypted. The NVM is equipped with an error correction code (ECC). Security modules manage the alarms. Alarms may be triggered when the environmental conditions are outside the specified operational range.

A set of sensors (temperature sensor, backside light detector, glitch sensor, low frequency sensor) is used to detect excessive deviations from the specified operational range and serve for robustness of the TOE. The UMSLC function can be used to test the alarm lines.

A True Random Number Generator (TRNG) specially designed for smart card applications is implemented. The TRNG a class PTG.2 random number generator of [6] and produces genuine random numbers which can be used internally or by the user software.

The micro Symmetric Cryptographic Processor (µSCP) supports calculation of dual-key or triple-key triple-DES and AES.

The implemented sleep mode logic (clock stop mode per ISO/IEC 7816-3) is used to reduce overall power consumption.

The UART- or GPIO-controlled I/O interface allows the smart card controller and the terminal interface to be operated independently.

The UMSLC enables the user software to check the activity and proper function of the system's security features.

The Clock Unit (CLKU) supplies the clocks for all components of the TOE. The Clock Unit can work in internal and external clock mode. When operating the internal clock mode the system frequency is derived from an oscillator, whereas in external clock mode, the system clock is derived from an externally supplied interface clock.

The watchdog timer triggers an event in case of a counter overflow. The timers are general purpose up-counting timers.

The ROM is used by IFX only. The user software has to be implemented in SOLID FLASH™ memory. The user can choose, whether the software is loaded into the SOLID FLASH™ memory by Infineon Technologies AG or by the user.

The TOE uses Special Function Registers (SFRs). These SFRs are used for general purposes and chip configuration; they are located in SOLID FLASH™ memory in a configuration area page. The Online Configuration Check (OCC) function is used for register protection, i.e. controls the modification of relevant SFR settings.

In case a security violation is detected, secure state is entered by the hardware.

1.4.1.2 Firmware and software components

The TOE provides low-level firmware components: the Boot Software (BOS) and the Flash Loader (FL).

The BOS firmware is used for test purposes during start-up and the FL allows downloading of user software to the NVM during the manufacturing process. All mandatory functions for start-up and internal testing are protected by a dedicated hardware firewall with two levels "BOS" and "user".

The flash loader allows downloading of User Software into the NVM during the manufacturing process. It uses the µSCP to download encrypted user data.

The software of the TOE consists of packages:

- Optional Symmetric Crypto Library (SCL): The optional SCL is used to provide a high level interface to the TDES and AES cryptography, which is partly implemented on the hardware component µSCP and includes countermeasures against SPA, DPA and DFA attacks. The SCL is delivered as object code and in this way integrated into the user software.
- Optional Hardware Support Library (HSL): provides functionality via APIs to the Smartcard Embedded Software . which contains SOLID FLASH™ NVM service routines and functionality for tearing safe programming of SOLID FLASH™ NVM.

- UMSLC lib: this library provides a wrapper around the UMSLC hardware functionality with measures to counter fault attacks.

1.4.1.3 User Guidance components

The user guidance consists of the components as follows:

- 32-bit Security Controller – V02, Hardware Reference Manual: description of hardware features and user interfaces
- 32-bit ARM-based Securtity Controller, SLC 37/40-nm Technology, Programmer's Reference Manual: description of firmware principles relevant for IC embedded software.
- Production and personalization 32-bit ARM-based security cvontroller:contains detailed information about the usage of the Flash Loader
- 32-bit Security Controller – V02, Security Guidelines: provides the guidance and recommendations to develop secure software for and secure usage of this TOE.
- HSL library for SLCx7 in 40nm: provides an application interface (API) description and security guidelines for the optional HSL software part.
- UMSLC library for SLCx7 in 40nm, Version 01.00.0234: provides some guidelines, how to use the UMSLC library
- SCL37-uSCP-v3-C40 Symmetric Crypto Library for uSCP-v3 DES / AES (optional): User Interface, contains all interfaces of the SCL. This document is only delivered to the user in case the SCL is part of the delivered TOE.

1.4.2 Physical scope of the TOE

The physical scope of the TOE is defined by the TOE components described in chapter 1.4.1

1.4.3 Logical scope of the TOE

The logical scope of the TOE consists of the logical security features provided by the TOE. These features are listed in chapter 1.3.2. More details are provided in this chapter:

- Cryptographic support: TDES, AES (block cipher modes ECB, CBC, CFB, CTR and CMAC); RNG (PTG.2 according to [6])
- Memory protection unit supporting up to eight memory regions with different access rights and two privilege levels "privileged" and "user". "User" level is more restricted in using TOE resources compared to "privileged"
- Memory encryption: all data of memories ROM, RAM and NVM are encrypted. Addresses are scrambled to disguise the location of data
- Robust set of sensors and detectors for the purpose of monitoring proper chip operating conditions consisting of a temperature sensor, backside light detector, glitch sensor and low frequency sensor.
- Redundant alarm propagation and system deactivation principle, which decreases the risk of manipulation and tampering.
- Register protection: protection of security relevant registers against fault attacks using OCC.
- Security life control: a life test on specific security features can be used by the IC embedded software to detect manipulation of these security features
- Program flow integrity protection: The Instruction Stream Signature Checking (ISS) can be employed by the IC embedded software to detect illegal program flows and trigger an alarm. The TOE also contains a watchdog, which may be used to detect program flow manipulations.

- Peripheral access control: The TOE allows the IC embedded software to lock certain peripherals dynamically.
- Bus encryption for security peripherals: All data transfers to and from dedicated peripherals are encrypted dynamically.
- Tearing safe NVM programming: the HSL provides specific routines provided for tearing safe programming. These routines prevent an unspecified interim state by either propagating the pre- or post-programming condition.
- Security optimized wiring: shield lines in combination with layout measures reduce the risk of successful manipulative attacks.
- Leakage control of data dependant code execution: dedicated measures allow the user to reduce such leakage.
- Device phase management supporting isolation of test features and flash loader accessibility: dedicated test features employed during production are switched off before customer delivery. The flash loader usage to download flash data requires a mutual authentication. The flash loader supports permanent deactivation.
- Detection of NVM single and multi bit errors: Single bit errors are detected and corrected and multi bit errors detected.

1.4.4 Interfaces of the TOE

- The physical interface of the TOE to the external environment is the entire surface of the IC.
- The electrical interface of the TOE to the external environment is constituted by the pads of the chip:
    - The five ISO 7816 pads consist particularly of the contacted RES, I/O, CLK lines and supply lines VCC and GND. The contact based communication is according to ISO 7816/ETSI/EMV.
    - The I2C communication can be driven via the ISO 7816 pads. In this case no other communication using the ISO 7816 pads is possible.
- The data-oriented I/O interface of the TOE is represented by the I/O pad.
- The interface between firmware and hardware consists of special registers used for hardware configuration and control (Special Function Registers, SFR).
- Optional: The interface of the TOE to the operating system is covered by the optional HSL routines and by the instruction set of the TOE.
- The interface of the UMSLC lib defined by the UMSLC lib
- Optional: The interface to the SCL calculations is defined by the SCL

1.4.5 Forms of Delivery

The TOE can be delivered in the form of complete modules, as plain wafers in an IC case (e.g. DSO20) or in bare dies. The delivery can therefore be at the end of phase 3 or at the end of phase 4 which may also include pre-personalization steps according to [1]. This means phase 4 is also part of the evaluation process. In any case the testing of the TOE is finished and the extended test features are removed. From a security policy point of view the different forms of delivery do not have any impact.

The delivery to the software developer (phase 2 ➔ phase 1) contains the documents as described above.

Part of the software delivery is the Flash Loader program, provided by Infineon Technologies AG, running on the TOE and controlling the download of user software onto the TOE via the UART interface. The download is only possible after successful authentication. The user software and data must be encrypted before download. In addition, the user can permanently block further use of the Flash Loader.

The table as follows provides an overview about form and method of TOE deliveries:

Table 2 TOE deliveries: forms and methods

TOE Component	Delivered Format	Delivery Method	Comment
Hardware			
IFX_CCI_000Dh, IFX_CCI_0020h, IFX_CCI_0031h, IFX_CCI_0032h, IFX_CCI_0034h, IFX_CCI_0037h T31 and M31	Wafer, IC case, packages	Postal transfer in cages	All materials are delivered to distribution centers in cages, locked.
Firmware			
All (see Table 1 “firmware”)	–	–	stored on the delivered hardware.
Software			
All software libraries (see Table 1 “Software”)	L251 Library File (object code)	Secured download[1]	–
Guidance Documentation			
HSL library for SLCx7 in 40nm	Compiled html help (chm)	Secured download[1]	–
UMSLC library for SLCx7 in 40nm, Version 01.00.0234	Compiled html help (chm)	Secured download[1]	–
All other User Guidance documents (see Table 1 “User Guidance”)	PDF	Secured download[1]	–

1.4.6 Production sites

The TOE may be handled at different production sites but the silicon is produced at Global Foundries fab 7 in Singapore only. The production site can be determined by the non-ISO ATR.

The delivery measures are described in the ALC_DVS aspect.

1.4.7 TOE Configuration

This TOE is represented by various configurations called products.

The module design, layout and footprint, of all products are identical.

The degree of freedom for configuring the TOE is predefined by Infineon Technologies AG. The tables as follows show the TOE hardware configurations:

Table 3 TOE hardware configuration options

Memory	Values	Identification
SOLID FLASH™	up to 800 kBytes	IFX-Mailbox, see [7] where this value is encoded

[1] Secured download is a way of delivery of documentation and TOE related software using a secure ishare connected to Infineon customer portal. The TOE user needs a DMZ Account to login (authenticate) via the Internet.

Memory	Values	Identification
RAM	up to 20 kBytes	IFX-Mailbox, see [7] where this value is encoded

Table 4 TOE operating temperature range configuration options

Min value	Max value	Identification
-25 C	+85 C	Design step T31
-40 C	+105 C	Design step M31

Further the flash-loader can be configured in different ways as explained in the following section.

1.4.8 TOE initialization with Customer Software

This TOE is equipped with Flash Loader software (FL) to download user software, i.e. an operating system and applications. Various options can be chosen by the user to store software onto the SOLID FLASH™ NVM:

Table 5 Options to initialize the TOE with customer software

Case	Option	Flash loader status
1	The user or/and a subcontractor downloads the software into the SOLID FLASH™ memory. Infineon Technologies does not receive any user software.	The Flash Loader can be activated or reactivated by the user or subcontractor to download software into the SOLID FLASH™ memory.
2	The user provides software to download into the SOLID FLASH™ memory to Infineon Technologies AG. The software is loaded into the SOLID FLASH™ memory during chip production.	There is no Flash Loader present.
3	The user provides software to download into the SOLID FLASH™ memory to Infineon Technologies AG. The software is loaded into the NVM memory during chip production.	The Flash Loader is blocked by Infineon but can be activated or reactivated by the user or subcontractor to download software into the SOLID FLASH™ memory. The user is required to provide a reactivation procedure as part of the software to Infineon Technologies AG.

2 Conformance Claims (ASE_CCL)

2.1 CC Conformance Claim

This ST and TOE claim conformance to CC:2022. The ST claims conformance to [CCBook3]. It is [CCBook2] extended.

2.2 PP Claim

This Security Target Lite claims **strict conformance** to [1].

The Security IC Platform Protection Profile with Augmentation Packages is registered and certified by the Bundesamt für Sicherheit in der Informationstechnik[1] (BSI) under the reference:

BSI-CC-PP-0084-2014, Version 1.0, dated 2014-01-13.

The security assurance requirements of the TOE are according to the Security IC Platform Protection Profile [1] and [CCBook3].

The augmentations of the PP [1] are listed below.

Table 6 Augmentations of the assurance level of the TOE

Assurance Class	Assurance components	Description
Life-cycle support	ALC_DVS.2	Sufficiency of security measures
Vulnerability assessment	AVA_VAN.5	Advanced methodical vulnerability analysis

2.3 Package Claim

This Security Target Lite implements the functional packages from [1] as follows:

- Packages “TDES” augmented and “AES” augmented; sections 7.4.1 and 7.4.2 in case the SCL is part of the TOE
- Package “Loader dedicated for usage in secured environment only” conformant; section 7.3.1
- Package “Loader dedicated for usage by authorized users only” conformant, in case of an active flash loader; section 7.3.2
- Package “Authentication of the Security IC” conformant in case of an active flash loader; section 7.2

The assurance level for the TOE is EAL5 augmented with the components ALC_DVS.2 and AVA_VAN.5. Therefore this ST is **package-augmented** to the packages in [1].

2.4 Conformance Rationale:

The TOE is a typical security IC as defined in [1] chapter 1.2.2 comprising:

- the circuitry of the IC (hardware including the physical memories),
- configuration data, initialization data related to the IC Dedicated Software and the behaviour of the security functionality
- the IC Dedicated Software with the parts

[1] Bundesamt für Sicherheit in der Informationstechnik (BSI) is the German Federal Office for Information Security

Conformance Claims (ASE_CCL)

- the IC Dedicated Test Software,
- the IC Dedicated Support Software.

The TOE is designed, produced and/or generated by the TOE Manufacturer.

With CC:2022 several SFR changes are introduced. Due to this ST claiming conformance to CC:2022 and [1], rationales are provided that these changes do not affect the conformance claim to [1]:

- FCS_COP.1: for this SFR dependencies are changed in CC:2022. FCS_CKM.4 is removed and instead FCS_CKM.6 added. Further FCS_CKM.5 is added for key derivation as an alternative.
- FCS_CKM.1: for this SFR dependencies are changed in CC:2022. Additionally to FCS_CKM.2 and FCS_COP.1, one further SFR is introduced as alternative: FCS_CKM.5. This SFR targets key derivation, subsequent to FCS_CKM.1. In CC:2022 key derivation would have been part of FCS_CKM.1 and thus conformancy to [1] can still be claimed. FCS_CKM.4 is removed and instead FCS_CKM.6 added. All other dependencies (i.e. FCS_RNG.1 or FCS_RBG.1) are in addition to the already existing ones, i.e. add stricter requirements.
- FCS_CKM.6 replaces FCS_CKM.4 and adds further requirements on the timing of key destruction. As an alternative dependency to FCS_CKM.1, FCS_CKM.5 (key derivation) can be used. As FCS_CKM.5 is neither used within [1] nor within this ST, it has no relevance in this context.
- FCS_RNG.1: this SFR is taken from [CCBook2] rather than [1]. The SFR is identical in [CCBook2]
- FMT_LIM.1 and FMT_LIM.2 are taken from [CCBook2] rather than [1]. There is a slightly different phrasing (i.e. removing redundancy from FMT_LIM.1) and availability and capability policy mentioned in both SFR's. The meaning though is the same and therefore conformancy can still be claimed.
- FIA_API.1: this SFR is taken from [CCBook2] rather than [1]. The SFR requires additional information.
- FDP_SDC.1: this SFR is taken from [CCBook2] rather than [1]; An assignment from [1] is changed to a selection [CCBook2], which means the requirement is more stringent and thus can be considered a subset of the requirement from [1].

Further with CC:2022 some SAR changes were introduced. Rationales are provided that these changes do not affect the conformance claim to [1]:

- ASE_CCL.1: for CC:2022 several extensions were introduced (e.g. exact conformance to PP), which add to the already existing assurance requirements. No relaxation was introduced.
- ASE_INT.1: introduction of multi-assurance in combination with PP-configuration: not relevant for [1]
- ASE_REQ.2: extended for multi assurance: not relevant for [1]
- AVA_VAN.5: extension about third party components introduced. No relaxation was introduced.
- ALC_TAT.1: extension with guidance on the minimum content for an implementation standards description and rules with ADV_COMP.1. No relaxation was introduced.

3 Security Problem Definition (ASE_SPD)

The content of [1] applies to this chapter completely.

3.1 Threats

The threats are directed against the assets and/or the security functions of the TOE. For example, certain attacks are only one step towards a disclosure of assets while others may directly lead to a compromise of the application security. The more detailed description of specific attacks is given later on in the process of evaluation and certification. An overview on attacks is given in PP [1] section 3.2.

The threats to security are defined and described in PP [1] sections 3.2 and 7.2.1.

Table 7 Threats according to [1]

T.Phys-Manipulation	Physical Manipulation
T.Phys-Probing	Physical Probing
T.Malfunction	Malfunction due to Environmental Stress
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 (only considered in case flash loader is active or during delivery of the TOE to the customer)	Masquerade the TOE

3.1.1 Additional Threat due to TOE specific Functionality

The additional functionality of introducing sophisticated privilege levels and access control allows the secure separation between the operation system(s) and applications, the secure downloading of applications after personalization and enables multitasking by separating memory areas and performing access controls between different applications. Due to this additional functionality “area based memory access control” a new threat is introduced.

The TOE shall avert the threat “Memory Access Violation (T.Mem-Access)” as specified below:

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) or privilege levels. Any restrictions are defined by the security policy of the specific application context and must be implemented by the Smartcard Embedded Software.

“Diffusion of open samples” threat:

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.

Note: this threat is only relevant, if the flash-loader is not blocked, i.e. additional software can be loaded onto the TOE.

Table 8 Additional threats due to TOE specific functions and augmentations

T.Mem-Access	Memory Access Violation
T.Open_Samples_Diffusion	Diffusion of Open Samples

3.1.2 Assets regarding the Threats

The asset description from PP [1] section 3.1 applies.

3.2 Organizational Security Policies

The organizational policies from [1] sections 3.3, 7.3.1, 7.3.2 and 7.4 are applicable.

Table 9 Organizational Security Policies according PP [1]

P.Process-TOE	Protection during TOE Development and Production
P.Crypto-Service (only available, if SCL is part of the TOE)	Cryptographic services of the TOE
P.Lim_Block_Loader	Limiting and Blocking the Loader Functionality
P.Ctlr_Loader (only available , if flash loader active)	Controlled usage to Loader Functionality

3.3 Assumptions

The TOE assumptions about the operational environment are defined and described in PP [1] section 3.4.

Table 10 Assumption according PP [1]

A.Process-Sec-IC	Protection during Packaging, Finishing and Personalization
A.Resp-Appl	Treatment of User Data

4 Security objectives (ASE_OBJ)

This section shows the security objectives, which are relevant to the TOE.

4.1 Security objectives of the TOE

The security objectives of the TOE are defined and described in PP [1] sections 4.1, 7.2.1, 7.3.1, 7.3.2, 7.4.1 and 7.4.2

Table 11 Objectives for the TOE according to PP [1]

O.Phys-Manipulation	Protection against Physical Manipulation
O.Phys-Probing	Protection against Physical Probing
O.Malfunction	Protection against Malfunction
O.Leak-Inherent	Protection against Inherent Information Leakage
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.Ctlr_Auth_Loader (only available , if flash loader active)	Access control and authenticity for the Loader
O.Authentication (only available , if flash loader active)	Authentication to external entities
O.TDES (only available, if SCL is part of the TOE)	Cryptographic service Triple-DES
O.AES (only available, if SCL is part of the TOE)	Cryptographic service AES

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 and privilege levels is controlled as required, for example, in a multi-application environment.

The TOE shall provide TSF confidentiality protection as specified below:

O.Prot_TSF_Confidentiality **Protection of confidentiality of 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.

Security objectives (ASE_OBJ)

Table 12 Additional objectives due to TOE specific functions and augmentations

O.Mem-Access	Area based Memory Access Control
O.Prot_TSF_Confidentiality	Protection of confidentiality of TSF

4.2 Security Objectives for the development and operational Environment

The security objectives from [1] section 4.2, 4.3, 7.2.1, 7.3.1, 7.3.2, 7.4.1 and section 7.4.2 are applicable for this TOE.

The table below lists the environmental security objectives.

Table 13 Security objectives for the environment according to [1]

Environmental objective	description
OE.Resp-Appl	Treatment of User Data
OE.Process-Sec-IC	Protection during composite product manufacturing
OE.Lim_Block_Loader	Limitation of capability and blocking the Loader
OE.Loader_Usage (only relevant , if flash loader active)	Secure communication and usage of the Loader
OE.TOE_Auth (only relevant , if flash loader active)	External entities authenticating of the TOE

Table 14 Additional Security objectives for the environment

Environmental objective	Description
OE.Secure_Delivery (only applicable if Flash Loader deactivated (i.e. cases 2 and 3 from Table 5))	When the TOE is ordered with a disabled Flash Loader, it does not provide transport protection. Therefore, technical and / or organisational security procedures (e.g. a custom mutual authentication mechanism or a security transport) should be put in place by the customer to secure the personalized TOE during delivery as required by the security needs of the loaded IC Embedded Software.

4.3 Security Objectives Rationale

The security objectives rationale of the TOE is defined and described in PP [1] section 4.4, 7.3.1, 7.3.2, 7.4.1 and section 7.4.2.

Compared to [1] an enhancement regarding memory area protection has been established. The clear definition of privilege levels for operated software establishes the clear separation of different restricted memory areas for running the firmware, downloading and/or running the operating system and to establish a clear separation between different applications. Nevertheless, it is also possible to define a shared memory section where separated applications may exchange defined data. The privilege levels clearly define by using a hierarchical

model the access right from one level to the other. These measures ensure that the threat T.Mem-Access is clearly covered by the security objective O.Mem-Access.

The objectives O.Authentication, O.Ctrl_Auth_Loader and the organizational policy P.Ctlr_Loader and the environmental objective OE.TOE_Auth as described in [1] chapter 7.2 and 7.3.2 apply only to TOE products with Flash Loader enabled for software or data download by the user. In other cases the Flash Loader is not available anymore and the user software or data download is completed.

The objective O.Prot_TSF_Confidentiality counters the threat T.Open_Samples_Diffusion. In addition T.Open_Samples_Diffusion is countered by O.Leak-Inherent and O.Leak-Forced.

The objective OE.Secure_Delivery requires the customers to provide transport protection in case the TOE is delivered with flash loader deactivated (i.e. cases 2, 3 from Table 5). In that case O.Authentication is not available and needs to be compensated by customer measures. OE.Secure_Delivery and O.Authentication counter T.Masquerade_TOE.

5 Extended Component Definition (ASE_ECD)

There are several extended components defined and described for the TOE:

- the family FCS_RNG at the class FCS Cryptographic Support
- the family FMT_LIM at the class FMT Security Management
- the family FAU_SAS at the class FAU Security Audit
- the family FDP_SDC at the class FDP User Data Protection
- the family FIA_API at the class FIA Identification and Authentication
- the component FPT_TST.2 at the class FPT Protection of the TSF

The extended families FCS_RNG, FMT_LIM, FAU_SAS, FDP_SDC and FIA_API are defined and described in PP [1] section 5. The component FPT_TST.2 is defined in the following sections.

5.1 Component “Subset TOE security testing (FPT_TST.2)”

The security is strongly dependent on the correct operation of the security functions. Therefore, the TOE shall support that particular security functions or mechanisms are tested in the operational phase (Phase 7). The tests can be initiated by the Smartcard Embedded Software and/or by the TOE or is done automatically and continuously.

Part 2 of the Common Criteria provides the security functional component “TSF testing (FPT_TST.1)”. The component FPT_TST.1 provides the ability to test the TSF’s correct operation.

For the user it is important to know which security functions or mechanisms can be tested. The functional component FPT_TST.1 does not mandate to explicitly specify the security functions being tested. In addition, FPT_TST.1 requires verification of the integrity of TSF data and of the stored TSF executable code which might violate the security policy. Therefore, the functional component **”Subset TOE security testing (FPT_TST.2)”** of the family TSF self test has been newly created. This component allows that particular parts of the security mechanisms and functions provided by the TOE are tested.

5.2 Definition of FPT_TST.2

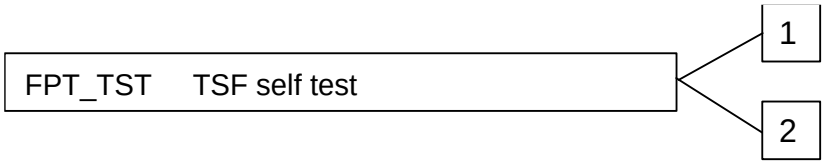
The functional component “Subset TOE security testing (FPT_TST.2)” has been newly created (Common Criteria Part 2 extended). This component allows that particular parts of the security mechanisms and functions provided by the TOE can be tested after TOE Delivery or are tested automatically and continuously during normal operation transparent for the user.
This security functional component is used instead of the functional component FPT_TST.1 from Common Criteria Part 2. For the user it is important to know which security functions or mechanisms can be tested. The functional component FPT_TST.1 does not mandate to explicitly specify the security functions being tested. In addition, FPT_TST.1 requires verifying the integrity of TSF data and stored TSF executable code which might violate the security policy.

The functional component “Subset TOE testing (FPT_TST.2)” is specified as follows (Common Criteria Part 2 extended).

5.3 TSF self test (FPT_TST)

Family Behavior The Family Behavior is defined in [CCBook2] section 15.17.1.

Component levelling



FPT_TST.1: The component FPT_TST.1 is defined in [CCBook2] section 15.17.5.

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

Management: FPT_TST.2

The following actions could be considered for the management functions in FMT:

- management of the conditions under which subset TSF self testing occurs, such as during initial start-up, regular interval or under specified conditions
- management of the time of the interval appropriate.

Audit: FPT_TST.2

There are no auditable events foreseen.

FPT_TST.2 Subset TOE testing

Hierarchical to: No other components.

Dependencies: No dependencies

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

6 Security Requirements (ASE_REQ)

For this section [1] section 6 can be applied completely.

6.1 TOE Security Functional Requirements

The security functional requirements (SFR) for the TOE are defined and described in [1] and in the following description.

Table 15 provides an overview of the functional security requirements of the TOE, defined in [1] section 6.1, 7.2.3, 7.3.1, 7.3.2, 7.4.1 and 7.4.2.

Table 15 Security functional requirements of the TOE defined in PP [1]

Security Functional Requirement	
FRU_FLT.2	“Limited fault tolerance“
FPT_FLS.1	“Failure with preservation of secure state”
FMT_LIM.1	“Limited capabilities”
FMT_LIM.2	“Limited availability”
FAU_SAS.1	“Audit storage”
FDP_SDC.1	“Stored data confidentiality
FDP_SDI.2	“Stored data integrity monitoring and action”
FPT_PHP.3	“Resistance to physical attack”
FDP_ITT.1	“Basic internal transfer protection”
FPT_ITT.1	“Basic internal TSF data transfer protection
FDP_IFC.1	“Subset information flow control”
FCS_RNG.1	“Random number generation”
FCS_COP.1/TDES	“Cryptographic operation - TDES”
FCS_CKM.6/TDES	“Cryptographic key destruction”
FCS_COP.1/AES	“Cryptographic operation - AES”
FCS_CKM.6/AES	“Cryptographic key destruction”
FMT_LIM.1/Loader “Limited Capabilities – Loader”	
FMT_LIM.2/Loader “Limited availability – Loader”	
FTP_ITC.1 “Inter-TSF trusted channel”	
FDP_UCT.1 “Basic data exchange confidentiality”	
FDP_UIT.1 “Data exchange integrity”	
FDP_ACC.1/Loader “Subset access control – Loader”	
FDP_ACF.1/Loader “Security attribute based access control – Loader”	
FIA_API.1 “Authentication Proof of Identity”	

Table 16 provides an overview about security functional requirements, which are added to the TOE. All requirements are taken from [CCbook3] Part 2, with the exception of requirement FPT_TST.2, which is defined in this ST completely.

Table 16 Additional security functional requirements of the TOE

Security Requirements (ASE_REQ)

Security Functional Requirement	
FPT_TST.2	“Subset TOE security testing“
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”

6.1.1 Definition required by [1]

According to [1] Application Note 14 the term “secure state” used by FPT_FLS.1 shall be described and a definition should be provided.

Definition of secure state:

Secure state describes three different conditions of the TOE:

1. the controller ceases operation. This condition can only be resolved by a cold or warm start of the controller. It is triggered by a security reset.
2. the controller enters a security trap. The trap handler can be defined by the user. In case no trap handler is provided the first condition is entered.
3. in case of a sudden power loss of the TOE during NVM programming (tearing): the TOE is in a condition to either restore the old NVM content or to start with the new programmed value. This condition of security state is only provided in case the HSL is part of the TOE and one of the tearing-safe functions of the HSL is used.

Note: a security reset invalidates the RAM content.

According to [1] Application Note 15, “The Common Criteria suggest that the TOE generates audit data for the security functional requirements Limited fault tolerance (FRU_FLT.2) and Failure with preservation of secure state (FPT_FLS.1).” In case of the first two conditions no Audit data are collected, because the effect entering the secure state is immediately visible. For the third condition indirect audit data is available, i.e. the user can check, whether new or old NVM data is available.

6.1.2 Extended Component FAU_SAS.1

6.1.2.1 FAU_SAS

The [1] defines additional security functional requirements with the family FAU_SAS of the class FAU (Security Audit). 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 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 dependencies

Dependencies: No dependencies.

Security Requirements (ASE_REQ)

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 not changeable configuration page area and non-volatile memory.

6.1.3 Support of Cipher Schemes

FCS_RNG.1 Random Number Generation

Hierarchical to: No other components

Dependencies: No dependencies

FCS_RNG.1 Random numbers generation Class PTG.2 according to [6]

FCS_RNG.1.1 The TSF shall provide a physical random number generator that implements:

PTG.2.1A 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.2.2If 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.2.3The 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.2.4The 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 The TSF shall provide numbers in the format 32-bit that meet

PTG.2.6Test procedure A, as defined in [6] does not distinguish the internal random numbers from output sequences of an ideal RNG.

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

Note: The physical random number generator implements total failure testing of the random source data and a continuous random number generator test according to:
National Institute of Standards and Technology, Security Requirements for Cryptographic Modules, Federal Information Processing Standards Publication (FIPS) 140-2, 2002-03-12, chapter 4.9.2

The following additional specific security functionality is implemented in the TOE:

Security Requirements (ASE_REQ)

FCS_COP.1 Cryptographic operation requires a cryptographic operation to be performed in accordance with a specified algorithm and with a cryptographic key of specified sizes. The specified algorithm and cryptographic key sizes can be based on an assigned standard; dependencies are discussed in Section 6.3.1.1.

The TOE implements the packages “TDES” and “AES” from [1].

Triple-DES Operation

The TDES Operation of the TOE shall meet the requirement “Cryptographic operation (FCS_COP.1)” as specified below.

FCS_COP.1/TDES	Cryptographic operation
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation] FCS_CKM.6 Timing and event of cryptographic key destruction
FCS_COP.1.1/TDES	The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm TDES in ECB mode, CBC mode, CFB mode, CTR mode, CMAC mode and cryptographic key sizes 112 bit and 168 bit that meet the following: [19], [20], [22]

Note: The TOE can be delivered with or without the SCL. If the user decides not to use the SCL, the SFR above is not part of the TOE.

FCS_CKM.6/TDES	**Timing and event of cryptographic key destruction**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation]
FCS_CKM.6.1/TDES	The TSF shall destroy cryptographic keys when the user requests it.
FCS_CKM.6.2/TDES	The TSF shall destroy cryptographic keys and keying material specified by FCS_CKM.6.1 in accordance with a specified cryptographic key destruction method overwriting or zeroing that meets the following: None

The SCL offers a function to wipe the key with random numbers.

Note: The TOE can be delivered with or without the SCL. If the user decides not to use the SCL, the SFR above is not part of the TOE.

AES Operation

The AES Operation of the TOE shall meet the requirement “Cryptographic operation (FCS_COP.1)” as specified below.

Security Requirements (ASE_REQ)

FCS_COP.1/AES	Cryptographic operation
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation] FCS_CKM.6 Timing and event of cryptographic key destruction
FCS_COP.1.1/AES	The TSF shall perform decryption and encryption in accordance with a specified cryptographic algorithm AES in ECB mode, CBC mode, CFB mode, CTR mode, CMAC mode, and cryptographic key sizes 128 bit, 192 bit and 256 bit that meet the following: [21], [20], [22]

Note: The TOE can be delivered with or without the SCL. If the user decides not to use the SCL, the SFR above is not part of the TOE.

FCS_CKM.6/AES	**Timing and event of cryptographic key destruction**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation]
FCS_CKM.6.1/AES	The TSF shall destroy cryptographic keys when the user requests it.
FCS_CKM.6.2/AES	The TSF shall destroy cryptographic keys and keying material specified by FCS_CKM.6.1 in accordance with a specified cryptographic key destruction method overwriting or zeroing that meets the following: None

The SCL offers a function to wipe the key with random numbers.

Note: The TOE can be delivered with or without the SCL. If the user decides not to use the SCL, the SFR above is not part of the TOE.

6.1.4 Subset of TOE testing

The security is strongly dependent on the correct operation of the security functions. Therefore, the TOE shall support that particular security functions or mechanisms are tested in the operational phase (Phase 7). The tests can be initiated by the Smartcard Embedded Software and/or by the TOE.

The TOE shall meet the requirement “Subset TOE testing (FPT_TST.2)” as specified below (Common Criteria Part 2 extended).

FPT_TST.2 Subset TOE testing

Hierarchical to: No other components.

Dependencies: No dependencies

Security Requirements (ASE_REQ)

FPT_TST.2.1 The TSF shall run a suite of self tests at the conditions request of the Security IC Embedded Software to demonstrate the correct operation of the alarm lines and/or the environmental sensor mechanisms*:*

Please refer to the confidential Security Target

6.1.5 Memory access control

Usage of multiple applications in one Smartcard often requires code and data separation in order to prevent one application from accessing code and/or data of another application. For this reason the TOE provides Area based Memory Access Control. The underlying memory protection unit (MPU) is documented in [7].

The security service being provided is described in the Security Function Policy (SFP) Memory Access Control Policy. The security functional requirement “Subset access control (FDP_ACC.1)” requires that this policy is in place and defines the scope were it applies. The security functional requirement “Security attribute based access control (FDP_ACF.1)” defines security attribute 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. The decision whether an access is permitted or not is taken based upon attributes allocated to the software. The Smartcard Embedded Software defines the attributes and memory areas. The corresponding permission control information is evaluated “on-the-fly” by the hardware so that access is granted/effective or denied/inoperable.

The security functional requirement “Static attribute initialisation (FMT_MSA.3)” claims that the default values of security attributes are appropriate either permissive or restrictive in nature. Alternative values can be specified by any subject provided that the Memory Access Control Policy allows that. This is described by the security functional requirement “Management of security attributes (FMT_MSA.1)”. The attributes are determined during TOE manufacturing (FMT_MSA.3) or set at run-time (FMT_MSA.1).

From TOE’s point of view the different roles in the Smartcard Embedded Software can be distinguished according to the memory based access control. However the definition of the roles belongs to the user software.

The following Security Function Policy (SFP) Memory Access Control Policy is defined for the requirement “Security attribute based access control (FDP_ACF.1)”:

6.1.6 Memory Access Control Policy

The TOE shall support the standard ARMv7 Protected Memory System Architecture model.
The MPU provides full support for:

- Protection regions.
- Overlapping protection regions, with ascending region priority:
  - Region 7 = highest priority.
  - Region 0 = lowest priority.
- Access permissions.
- MPU mismatches and permission violations invoke the programmable-priority MemManage fault handler.

The MPU can be used to:

- Enforce privilege rules, preventing user applications from corrupting operating system data.
- Separate processes, blocking the active task from accessing other tasks’ data.
- Enforce access rules, allowing memory regions to be defined as read-only or detecting unexpected memory accesses.

Subjects, Objects and Operations of the policy

- Subjects: privilege or non-privilege level of the ARM processor
- Objects: memory/code addresses
- Operations: Read a/o write a/o execute access

Attributes of the policy:

- MPU enable/disable bit.
- 8 regions with the following attributes
  - A unique priority
  - The enable bit
  - the start address and size
  - an access matrix which defines if an Operation of a Subject to an Object lying in the region is allowed or denied
- The default region with the following security attribute:
  - A bit which defines if an Operation for the Subject (privilege level) is allowed or if no Operation is allowed for any Subject.

Roles of the policy:
The roles correspond 1-1 to the subjects.

Properties of the policy:

- If an address is contained in multiple enabled regions, then the region with the highest priority defines the access rights.
- If an address is contained in no region then the default region defines the access rights.
- The region defining the access rights checks in the access matrix if the Subject has access to the Object with respect to the desired Operation. In case the access is denied the MPU throws an access violation exception.

Access rules between privilege level and non-privilege level:

- the privilege level has access to regions which are defined for non-privilege level access
- the non-privilege level has no access to the regions which are defined for privilege level access

Table 17 access control rules

Privileged Mode Permissions	User Mode Permissions	Description
No access	No access	All accesses generate a permission fault
Read/write	No access	Privileged mode access only
Read/write	Read only	Writes in user mode generate a permission fault
Read/write	Read/write	Full access
Read only	No access	Privileged mode read only
Read only	Read only	Privileged and user mode read only

The TOE shall meet the requirement “Subset access control (FDP_ACC.1)” as specified below.

FDP_ACC.1 Subset access control

Security Requirements (ASE_REQ)

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 on all Subjects, all Objects and all Operations.

The TOE shall meet the requirement “Security attribute based access control (FDP_ACF.1)” as specified below.

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 to objects based on the following:

As specified in the definition of the memory access control policy.

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

As specified in the definition of the memory access control policy.

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none.

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: none.

The TOE shall meet the requirement “Static attribute initialisation (FMT_MSA.3)” as specified below.

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 to provide restrictive[1] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the privilege level, to specify alternative initial values to override the default values when an object or information is created.

The TOE shall meet the requirement “Management of security attributes (FMT_MSA.1)” as specified below:

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]

[1] The static definition of the access rules is documented in [7]

FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

FMT_MSA.1.1 The TSF shall enforce the Memory Access Control Policy to restrict the ability to modify the security attributes "Attributes of the policy" from memory access control policy to the privilege level.

The TOE shall meet the requirement "Specification of management functions (FMT_SMF.1)" as specified below:

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 security management functions: access the configuration registers of the MPU.

6.1.7 Data Integrity

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

FDP_SDI.2 Stored data integrity monitoring and action

Hierarchical to: FDP_SDI.1 stored data integrity monitoring

Dependencies: No dependencies

FDP_SDI.2.1 The TSF shall monitor user data stored in containers controlled by the TSF for data integrity and one- and/or more-bit-errors on all objects, based on the following attributes: error correction ECC for the SOLID FLASH™ NVM.

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall correct 1 bit errors in the SOLID FLASH™ NVM automatically and inform the user about other bit errors.

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 all user data while it is stored in the any memory.

6.1.8 Limited Capabilities and Limited Availability

The SFR's FMT_LIM.1 and FMT_LIM.2 from [1] are adapted due to CC:2022.

FMT_LIM.1	**Limited Capabilities**
Hierarchical to:	No other components.

Security Requirements (ASE_REQ)

Dependencies:	FMT_LIM.2: Limited availability
FMT_LIM.1.1	The TSF shall limit its 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[1].

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 in a manner that limits its 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[2].

[1] [assignment: Limited capability and availability policy]

[2] [assignment: Limited capability and availability policy]

6.1.9 Support of Flash Loader

The TOE provides a Flash Loader to download user data into the SOLID FLASH™ NVM, either during production of the TOE or at customer site. The Flash Loader is dedicated for usage by authorized users only in secured and insecure environment during the production up to “Phase 6 Security IC Personalisation”. The Flash Loader has to be permanently deactivated before entering “Phase 7 Security IC end-usage”. For this reason the TOE shall meet the requirements as defined and described in the PP [1] section “7.3 Packages for Loader” and “7.2 Package “Authentication of the Security IC”:

6.1.10 Flash Loader Policy

The Flash Loader supports the following security function policy (SFP)::

- Limited capabilities (FMT_LIM.1/Loader),
- Limited availability – Loader (FMT_LIM.2/Loader),
- Authentication Proof of Identity (FIA_API.1),
- Inter-TSF trusted channel (FTP_ITC.1),
- Basic data exchange confidentiality (FDP_UCT.1),
- Data exchange integrity (FDP_UIT.1),
- Subset access control – Loader (FDP_ACC.1/Loader),
- Security attribute based access control – Loader (FDP_ACF.1/Loader)

as defined in the PP [1], section 7.2 and 7.3.

The Flash Loader supports the following security function policy (SFP):

- Loader SFP: provides the mutual authentication between the TOE and the Administrator user or Download operator user, the management of keys (Kc, Kd, Kfdi) and the download of the User data into the memory of the TOE. The Loader SFP protects the downloaded data against unauthorized disclosure, modification, deletion and insertion by transferring data always in encrypted form by using Kfdi and including signature values in the data string which are checked during the download process.

The Flash Loader supports the following subjects defined by the roles:

- Administrator user.
- Download operator user.

Deployment of loader, which covers the following Flash Loader functionality:

- The Administrator user is enabled performing mutual authentication with the key Kc, to manage (set, exchange, delete) the keys Kc, Kd and Kfdi and to process the download of the User data into the memory of the TOE.
- Download operator user is enabled performing mutual authentication with Kd, to exchange the key Kd and to perform the download of the User data into the memory of the TOE. He can also delete Kfdi.

The Flash Loader supports the following object:

- user data: Data loaded into the memory of the TOE.
- The Flash Loader supports the following security attributes:
- Keys Kc and Kd used for the mutual authentication process.
- Key Kfdi used to encrypt/decrypt the user data.

Security Requirements (ASE_REQ)

The SFR's FMT_LIM.1 and FMT_LIM.2 from [1] are adapted due to CC:2022.

The TOE shall meet the requirements "Limited capabilities (FMT_LIM.1/Loader)" as specified below:

FMT_LIM.1/Loader	Limited Capabilities
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 permanent deactivation does not allow stored user data to be disclosed or manipulated by unauthorized user.

The TOE shall meet the requirement "Limited availability – Loader (FMT_LIM.2/Loader)" as specified below:

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 permanent deactivation.

Regarding FMT_LIM.1.1/Loader the User Guidance requires the Flash Loader to be permanently deactivated prior delivery to the end user (Phase 7).

The TOE shall meet the requirement "Inter-TSF trusted channel (FTP_ITC.1)" as specified below.

FTP_ITC.1	Inter-TSF trusted channel
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FTP_ITC.1.1	The TSF shall provide a communication channel between itself and administrator user, or Download operator user, that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2	The TSF shall permit another trusted IT product to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for deploying Loader for downloading user data.

The TOE Functional Requirement "Basic data exchange confidentiality (FDP_UCT.1)" is specified as follows.

FDP_UCT.1	Basic data exchange confidentiality

Security Requirements (ASE_REQ)

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_UCT.1.1	The TSF shall enforce the Loader SFP to receive user data in a manner protected from unauthorised disclosure.

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 Administrator User and Download Operator User,
	(2) the objects user data in SOLID FLASH™ NVM memory of the TOE,
	(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:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1/Loader	FDP_ACF.1.1 The TSF shall enforce the Loader SFP to objects based on the following:

(1) the subjects Administrator user and the Download operator user with security attributes Kc, Kd and Kfdi

(2) the objects user data in data loaded into the SOLID FLASH™ NVM memory of the TOE with security attributes Kfdi.

FDP_ACF.1.2/Loader	FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: (1) evaluate the corresponding access control information of the relevant subject, Administrator user and Download operator user, before the access, so that accesses to be denied cannot be utilized by the subject attempting to perform the operation. The subsequent download is then protected by the key Kfdi.
FDP_ACF.1.3/Loader	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/Loader	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: none.

The security functional requirements FIA_API.1, FTP_ITC.1, FDP_UCT.1, FDP_UIT.1, FDP_ACC.1/Loader and FDP_ACF.1/Loader apply only to TOE products with activated Flash Loader. In other cases the Flash Loader is not available anymore and the user data download is completed. Depending on the capabilities of the user software these security functional requirements may then reoccur as subject of the composite TOE.

The permanent locking of the Flash Loader after finalizing the download and prior delivery to the end-user is covered by FMT_LIM1/Loader and FMT_LIM.2/Loader.

6.1.11 Support of Authentication of the Security IC

The flash loader provides the security IC authentication service.

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 an authentication mechanism according to [25] section 6.2.2 Mechanism 4: Three-pass authentication based on the security attributes Kc and Kd to prove the identity of the TOE by including the following properties proof of knowledge of Flash Loader administrator or download operator credentials to an external entity.

This security functional requirement applies only to TOE products with Flash Loader activated.

6.2 TOE Security Assurance Requirements

The evaluation assurance level is EAL 5 augmented with ALC_DVS.2 and AVA_VAN.5. In the following table, the security assurance requirements are given.

Table 18 Assurance components

Security Requirements (ASE_REQ)

Aspect	Acronym	Description	Refinement
Development	ADV_ARC.1	Security Architecture Description	[1]
	ADV_FSP.5	Complete semi-formal functional specification with additional error information	[1]
	ADV_IMP.1	Implementation representation of the TSF	[1]
	ADV_INT.2	Well-structured internals	
	ADV_TDS.4	Semi-formal modular design	
Guidance Documents	AGD_OPE.1	Operational user guidance	[1]
	AGD_PRE.1	Preparative procedures	[1]
Life-Cycle Support	ALC_CMC.4	Production support, acceptance procedures and automation	[1]
	ALC_CMS.5	Development tools CM coverage	[1]
	ALC_DEL.1	Delivery procedures	[1]
	ALC_DVS.2	Sufficiency of security measures	[1]
	ALC_LCD.1	Developer defined life-cycle model	
	ALC_TAT.2	Compliance with implementation standards	
Security Target Evaluation	ASE_CCL.1	Conformance claims	
	ASE_ECD.1	Extended components definition	
	ASE_INT.1	ST introduction	
	ASE_OBJ.2	Security objectives	
	ASE_REQ.2	Derived security requirements	
	ASE_SPD.1	Security problem definition	
	ASE_TSS.1	TOE summary specification	
Tests	ATE_COV.2	Analysis of coverage	[1]
	ATE_DPT.3	Testing: modular design	
	ATE_FUN.1	Functional testing	
	ATE_IND.2	Independent testing - sample	
Vulnerability Assessment	AVA_VAN.5	Advanced methodical vulnerability analysis	[1]

6.2.1 Refinements

Some refinements are taken unchanged from [1]. Table 18 provides an overview.

Two refinements from [1] have to be discussed here in the Security Target Lite, as the assurance level is increased.

6.2.1.1 Life cycle support (ALC_CMS)

The refinement from [1] can also be applied to the assurance level EAL 5 augmented with ALC_CMS.5. The assurance package ALC_CMS.4 is extended to ALC_CMS.5 with aspects regarding the configuration control system for the TOE. The refinement is still valid.

6.2.1.2 Functional Specification (ADV_FSP)

The refinement from [1] can also be applied to the assurance level EAL 5 augmented with ADV_FSP.5. The assurance package ADV_FSP.4 is extended to ADV_FSP.5 with aspects regarding the level of description. ADV_FSP.5 requires a semi-formal description in addition. The refinement is still valid.

For refinement details see [1].

6.3 Security Requirements Rationale

6.3.1 Rationale for the Security Functional Requirements

While the security functional requirements rationale of the TOE are defined and described in [1] section 6.3.1, 7.2.3, 7.3.1, 7.3.2, 7.4.1 and 7.4.2, the additional introduced SFR are discussed below:

Table 19 Rational for additional SFRs in the ST

Objective	TOE Security Functional Requirements
O.Phys-Manipulation	- FPT_TST.2 „ Subset TOE security testing “
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”
O.Prot_TSF_Confidentiality	- FTP_ITC.1 Inter-TSF-trusted channel - FDP_ACC.1/Loader Subset access control –Loader - FDP_ACF.1/Loader Security attribute based access control – Loader

The table above gives an overview, how the security functional requirements are combined to meet the security objectives (this table has to be read in addition to [1] table 2 “Security Requirements versus Security Objectives”. The detailed justification is given in the following:

The security functional component Subset TOE security testing (FPT_TST.2) has been newly created (Common Criteria Part 2 extended). This component allows that particular parts of the security mechanisms and functions provided by the TOE can be tested after TOE Delivery. This security functional component is used instead of the functional component FPT_TST.1 from Common Criteria Part 2. For the user it is important to know which security functions or mechanisms can be tested. The functional component FPT_TST.1 does not mandate to explicitly specify the security functions being tested. In addition, FPT_TST.1 requires verification of the integrity of TSF data and stored TSF executable code which might violate the security policy.

The security functional requirement FPT_TST.2 detects attempts to conduce a physical manipulation on the monitoring functions of the TOE. The objective of FPT_TST.2 is O.Phys-Manipulation.

The security functional requirement “Subset access control (FDP_ACC.1)” with the related Security Function Policy (SFP) “Memory Access Control Policy” exactly require the implementation of an area based memory

access control as required by O.Mem-Access. The related TOE security functional requirements FDP_ACC.1, FDP_ACF.1, FMT_MSA.3, FMT_MSA.1 and FMT_SMF.1 cover this security objective. The implementation of these functional requirements is represented by the dedicated privilege level concept.

The objective O.Prot_TSF_Confidentiality is met by the loader access control (FDP_ACC.1/Loader and FDP_ACF.1/Loader) and trusted channel (FTP_ITC.1) for loading code. This prevents unauthorized users from generating open samples.

The justification of the security objective and the additional requirements show that they do not contradict the rationale already given in [1] for the assumptions, policy and threats defined there.

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 TOE only provides the tool to implement the policy defined in the context of the application.

6.3.1.1 Dependencies of Security Functional Requirements

The dependencies of security functional requirements are defined and described in [1] section 6.3.2, 7.2.3, 7.3.1, 7.3.2, 7.4.1 and section 7.4.2 for the following security functional requirements: FDP_SDC.1, FDP_SDI.2, FDP_ITT.1, FDP_IFC.1, FPT_ITT.1, FPT_PHP.3, FPT_FLS.1, FRU_FLT.2, FMT_LIM.1, FMT_LIM.2, FCS_RNG.1. FAU_SAS.1, FIA_API.1, FMT_LIM.1/Loader, FMT_LIM.2/Loader, FTP_ITC.1, FDP_UCT.1, FDP_UIT.1, FDP_ACC.1/Loader, FDP_ACF.1/Loader.

The dependencies of the additional security functional requirements (the functional requirements in addition to the ones defined in [1]) are analysed in the following description.

Table 20 Dependency for cryptographic operation requirement

Security Functional Requirement	Dependencies	Fulfilled by security requirements
FPT_TST.2	None	n.a.
FDP_ACC.1	FDP_ACF.1	Yes
FDP_ACF.1	FDP_ACC.1 FMT_MSA.3	Yes Yes
FMT_MSA.3	FMT_MSA.1 FMT_SMR.1	Yes Not required, see comment 1
FMT_MSA.1	FDP_ACC.1 or FDP_IFC.1 FMT_SMR.1 FMT_SMF.1	Yes see comment 1 Yes
FMT_SMF.1	None	n.a.
FCS_COP.1	FCS_CKM.6 [FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1, FCS_CKM.5]	Yes No, see comment 2
FCS_CKM.6	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5]	No, see comment 2
FDP_ACF.1/Loader	FMT_MSA.3	No, see comment 3

Comment 1:
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 (user). Therefore, there is no need to identify roles in form of a security functional requirement FMT_SMR.1.

End of comment.

Comment 2:
The security functional requirement “Cryptographic operation (FCS_COP.1)” met by the TOE has the following dependencies:

- [FDP_ITC.1 Import of user data without security attributes, or
- FDP_ITC.2 Import of user data with security attributes or
- FCS_CKM.1 Cryptographic key generation or
- FCS_CKM.5 Cryptographic key derivation]
- FCS_CKM.6 Timing and event of cryptographic key destruction.

These requirements all address the appropriate management of cryptographic keys used by the specified cryptographic function and are not part of [1]. Most requirements concerning key management shall be fulfilled by the environment since the Smartcard Embedded Software is designed for a specific application context and uses the cryptographic functions provided by the TOE.

For the security functional requirement FCS_COP.1/TDES and FCS_COP.1/AES the respective dependency [FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.5] has to be fulfilled by the environment.

The same applies for FCS_CKM.6. Its dependencies differ slightly from FCS_COP.1, however the rationale is the same.

End of comment.

Comment 3:

The inter-TSF trusted channel SFR FTP_ITC.1 has no dependency and is provided as main purpose by the Flash Loader. The Flash Loader provides a distinct and independent communication channel with authenticated end points and protection from modification or disclosure.

The dependency FMT_MSA.3 introduced by the component FDP_ACF.1/Loader is considered to be not required, because the security attributes enforcing the Loader SFP are fixed by the IC manufacturer and no new objects under the control of the Loader SFP are created. The Loader SFP also does not create any new security attributes and the security attributes are fixed during the download process. Claim 371 of [1] applies.

End of comment.

6.3.2 Rationale of the Assurance Requirements

The chosen assurance level EAL5 and the augmentation with the requirements ALC_DVS.2 and AVA_VAN.5 were chosen in order to meet the assurance expectations explained in the following paragraphs. In Table 18 the different assurance levels are shown as well as the augmentations. The augmentations are in compliance with the Protection Profile.

An assurance level EAL5 with the augmentations ALC_DVS.2 and AVA_VAN.5 are required for this type of TOE since it is intended to defend against highly sophisticated attacks without protective environment. 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 all information regarding the TOE including the TSF internals, the low level design and source code including

the testing of the modular design. Additionally the mandatory technical document [11] shall be taken as a basis for the vulnerability analysis of the TOE.

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 Initialization 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 AVA_VAN.5.

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_TDS.3 “Basic modular design”, ADV_IMP.1 “Implementation representation of the TSF”, 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 smartcards 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.

7 TOE Summary Specification (ASE_TSS)

The product overview is given in Section 1.3.1. The Security Features are described below and the relation to the security functional requirements is shown.

The TOE is equipped with the following security features to meet the security functional requirements:

Table 21 TOE Security Features

SF_DPM	Device Phase Management
SF_PS	Protection against Snooping
SF_PMA	Protection against Modification Attacks
SF_PLA	Protection against Logical Attacks
SF_CS	Cryptographic Support

The following description of the security features is a complete representation of the TSF.

The product overview is given in Section 1.3.1. The Security Features are described below and the relation to the security functional requirements is shown.

The TOE is equipped with the following security features to meet the security functional requirements:

Table 22 TOE Security Features

SF_DPM	Device Phase Management
SF_PS	Protection against Snooping
SF_PMA	Protection against Modification Attacks
SF_PLA	Protection against Logical Attacks
SF_CS	Cryptographic Support

The following description of the security features is a complete representation of the TSF.

7.1 SF_DPM: Device Phase Management

The life cycle of the TOE is split up into several phases. Different operation modes help to protect the TOE during each phase of its lifecycle.

7.2 SF_PS: Protection against Snooping

The TOE uses various means to protect from snooping of memories and busses and prevents single stepping.

7.3 SF_PMA: Protection against Modifying Attacks

This TOE implements protection against modifying attacks of memories, alarm lines, sensors and instruction execution order.

7.4 SF_PLA: Protection against Logical Attacks

Memory access of the TOE is controlled by a Memory Protection Unit (MPU), which implements different priviledge levels. The MPU decides, whether access to a physical memory location is allowed based on access rights.

7.5 SF_CS: Cryptographic Support

The TOE is equipped with a hardware accelerator and symmetric cryptographic library (SCL) to support the standard symmetric cryptographic operations TDES and AES. It further provides random numbers to meet FCS_RNG.1

7.6 Assignment of Security Functional Requirements to TOE's Security Functionality

The justification and overview of the mapping between security functional requirements (SFR) and the TOE's security functionality (SF) is given in the sections above. The results are shown in Table 23. The security functional requirements are addressed by at least one related security feature.

Table 23 Mapping of SFR and SF

SFR	SF_DPM	SF_PS	SF_PMA	SF_PLA	SF_CS
FRU_FLT.2			x		
FPT_FLS.1		x	x		x
FMT_LIM.1	x				
FMT_LIM.2	x				
FAU_SAS.1	x				
FDP_SDC.1		x			
FDP_SDI.2			x		
FPT_PHP.3		x	x		x
FDP_ITT.1	x	x	x		x
FPT_ITT.1	x	x	x		x
FDP_IFC.1		x	x		
FCS_RNG.1					x
FCS_COP.1/TDES					x
FCS_CKM.6/TDES					x
FCS_COP.1/AES					x
FCS_CKM.6/AES					x
FMT_LIM.1/Loader	x				
FMT_LIM.2/Loader	x				
FTP_ITC.1	x				
FDP_UCT.1	x				
FDP_UIT.1	x				
FDP_ACC.1/Loader	x				
FDP_ACF.1/Loader	x				
FIA_API.1	x				
FPT_TST.2			x		
FDP_ACC.1				x	

FDP_ACF.1				x	
FMT_MSA.1				x	
FMT_MSA.3				x	
FMT_SMF.1				x	

7.7 Security Requirements are internally consistent

For this chapter [1] section 6.3.4 can be applied completely.

The functional requirement FPT_TST.2 requires further protection to prevent manipulation of test results, while checking the security functions of the TOE. An attacker could aim to switch off or disturb certain sensors or filters and prevent the detection of distortion by blocking the correct operation of FPT_TST.2. 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 security functional requirement FPT_TST.2. Therefore, the related security functional requirements support the secure implementation and operation of FPT_TST.2.

The requirement FPT_TST.2 allows testing of some security mechanisms by the Smartcard Embedded Software after delivery.

The implemented privilege level concept represents the area based memory access protection enforced by the MPU. As an attacker could attempt to manipulate the level concept as defined and present in the TOE, the functional requirement FDP_ACC.1 and the related other requirements have to be protected. The security functional requirements necessary 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.

8 References

8.1 Literature

[CCBook2] Common Criteria for Information Technology Security Evaluation, CC:2022, revision 1, November 2022 — Part 2: Security functional components

[CCBook3] Common Criteria for Information Technology Security Evaluation, CC:2022, revision 1, November 2022 — Part 3: Security assurance components

[CCBook5] Common Criteria
for Information Technology Security Evaluation, Part 5: Pre-defined packages of security requirements, November 2022, CC:2022, Revision 1

[CCTrans] CCMC-2023-04-001, Transition Policy to CC:2022 and CEM:2022, 2023-04-20

[CCErrata] Errata and Interpretation for CC:2022 (Release 1) and CEM:2022 (Release 1), V1.1, 2024-07-22

[1] Security IC Platform Protection Profile with Augmentation Packages, Version 1.0, 13.01.2014, BSI-CC-PP-0084-2014

[5] Status report, List of all available user guidance

[6] Functionality classes and evaluation methodology for physical random number generators AIS31, Version 3.0, 05.15.2013

[7] 32-bit Security Controller –V02 Controller, Hardware Reference Manual, V9.2, 2020-02-06

[11] Joint Interpretation Library, Application of Attack Potential to Smartcards, Version 2.9, January 2013

[19] NIST SP 800-67, Revision 2, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, Nov 2017, National Institute of Standards and Technology

[20] NIST SP 800-38A Recommendation for Block Cipher Modes of Operation, 2001, with Addendum Recommendation for Block Cipher Modes of Operation: Three Variants of Ciphertext Stealing for CBC Mode, October 2010

[21] Federal Information Processing Standards Publication 197, ADVANCED ENCRYPTION STANDARD (AES), U.S. DEPARTMENT OF COMMERCE/National Institute of Standards and Technology, November 26, 2001

[22] NIST SP 800-38B Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication, 2005-05

[24] Act on the Federal Office for Information Security (BSI-Gesetz – BSIG), Bundesgesetzblatt I, p.2821; BSIG Section 9, Para.4, Clause 2, 2009-08-14

[25] ISO/IEC 9798-2, Information technology—Security techniques—Entity authentication, part 2: Mechanisms using symmetric encipherment algorithms, ISO/IEC, third edition, 2008-12-15

Note that the versions of these documents are listed in the certification report.

9 Appendix: hash signatures of the HSL

HSL-02.62.7626-SLCx7V14.lib:

MD5	8b193e15625b170eea1501e0bd6a4c1c
SHA-1	7ff5fb82070375d317b41e9fa3b210bdf18c5731
SHA-256	9f9463a1f88a3e5e113327c6bfe7d38ad3c7d5c5383303c6bf4b3adac4e562e3

10 Appendix: hash signatures of the SCL

SCL37-uSCP-v3-C40-cipher.lib:
MD5=1b15f6984c5d30e1ea20655b6fc4aa2b
SHA1=f345cf5368d1bdaf7bd073d7e2bacb83ec9fd45e
SHA256=a3b3db265e6c955621953055ccb94a960b3cb62c96e6850cf4b011a7ed6a2d0c

SCL37-uSCP-v3-C40-mac.lib:
MD5=a51407d783119041306ce7b8d1203fd6
SHA1=1cade3ac1ab63cf89f1086be295a74913d6a620c
SHA256=c96a7b67453988701b3bc36e4e292609a6125d19f3b1713b2f3ddf4b4d00d542

SCL37-uSCP-v3-C40-des.lib:
MD5=92de0c028dadfc70fb584c4b19900435
SHA1=559299be365eb0009b6069610738fd83b7b021ca
SHA256=19c328a961b872081d22ee7219ecbfe3fe0ebff1198c64744f7bb395451f8a95

SCL37-uSCP-v3-C40-aes.lib:
MD5=e9a2f95897882cad929d00db790c8f98
SHA1=8b1bde0324532f9d6b5b4a7455e6a56b28b1cfb2
SHA256=14147f477de102c9bdaf281183073d1563ef1ab02abaedcbbeface02827b0fcf

11 Appendix: hash signatures of UMSLC lib

UMSLC/UMSLC.lib:

MD5	2abc04d0b3711052db0aa531fe4e3f03
SHA-1	2365e551be7b79e5f5d89fea3ee80a78b613eef5
SHA-256	ec2c75184add66bbc89d825f5480fc4040720d560508ed4c618ab1f83d0e2e1f

12 List of Abbreviations

AES Advanced Encryption Standard

AIS31 “Anwendungshinweise und Interpretationen zu ITSEC und CC Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren”

API Application Programming Interface

ATR Answer to Reset

BLD Backside Light Detector

CC Common Criteria

CI Chip Identification Mode (STS-CI)

CPAU Codem Peripheral Access Unit

CPU Central Processing Unit

Crypto2304T Asymmetric Cryptographic Processor

DPA Differential Power Analysis

DFA Differential Failure Analysis

ECC Error Correction Code

EDC Error Detection Code

EMA Electro magnetic analysis

Flash Flash Memory

FSE Frequency Sensor

IC Integrated Circuit

ICO Internal Clock Oscillator

ID Identification

IMM Interface Management Module

ITP Interrupt and Peripheral Event Channel Controller

I/O Input/Output

ITSEC Information Technology Security Evaluation Criteria

MED Memory Encryption and Decryption

MMU Memory Management Unit

O Object

OCC Online Configuration Check

OS Operating system

PEC Peripheral Event Channel

List of Abbreviations

PRNG	Pseudo Random Number Generator
RAM	Random Access Memory
RNG	Random Number Generator
ROM	Read Only Memory
SAM	Service Algorithm Minimal
SCL	Symmetric Cryptographic Library
SCP	Symmetric Cryptographic Processor
SPAU	System Peripheral Access Unit
TSC	TOE Security Functions Control
TSE	Temperatrure Sensor
TSF	TOE Security Functionality
UART	Universal Asynchronous Receiver/Transmitter
UM	User Mode (STS)
UMSLC	User mode Security Life Control
VSE	Voltage Sensor
WDT	Watch Dog Timer
TDES	Triple DES Encryption Standard

13 Glossary

Application Program/Data	Software which implements the actual TOE functionality provided for the user or the data required for that purpose
Central Processing Unit	Logic circuitry for digital information processing
Chip Identification Data	Data to identify the TOE
CPAU	Code Peripheral Access Unit
Generic Chip Identification Mode	Operational status phase of the TOE, in which actions for identifying the individual chip by transmitting the Chip Identification Data take place
Memory Encryption and Decryption	Method of encoding/decoding data transfer between CPU and memory
Memory	Hardware part containing digital information (binary data)
Microprocessor	CPU with peripherals
Object	Physical or non-physical part of a system which contains information and is acted upon by subjects
Operating System	Software which implements the basic TOE actions necessary for operation
Programmable Read Only Memory	Non-volatile memory which can be written once and then only permits read operations
Random Access Memory	Volatile memory which permits write and read operations
Random Number Generator	Hardware part for generating random numbers
Read Only Memory	Non-volatile memory which permits read operations only
Resource Management System	Part of the firmware containing NVM programming routines, AIS31 testbench etc.
Self Test Software	Part of the firmware with routines for controlling the operating state and testing the TOE hardware
Security Function	Part(s) of the TOE used to implement part(s) of the security objectives
Security Target	Description of the intended state for countering threats
SmartCard	Plastic card in credit card format with built-in chip
Software	Information (non-physical part of the system) which is required to implement functionality in conjunction with the hardware (program code)
SPAU	System Peripheral Access Unit
Subject	Entity, generally in the form of a person, who performs actions
Target of Evaluation	Product or system which is being subjected to an evaluation
Test Mode	Operational status phase of the TOE in which actions to test the TOE hardware take place
Threat	Action or event that might prejudice security

Glossary

User Mode	Operational status phase of the TOE in which actions intended for the user takes place

14 Revision History

Major changes since the last revision

Version	Description of change
0.2	Initial draft version
5.4	Final version

Edition 2025-08-14

Published by
Infineon Technologies AG
81726 Munich, Germany

Do you have a question about this document?
Email: erratum@infineon.com

<DOC_Number>
Document reference

IMPORTANT NOTICE
The information contained in this Security Target is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this Security Target.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (**www.infineon.com**).

WARNINGS
Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.