紫光同芯
TONGXIN MICRO

ASE EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C Lite

Version 0.4

Tongxin Microelectronics Co., Ltd.
2025 – 08

Revision History

No.	Version	Date	Change	By
1	0.1	Jun. 2025	Create	Zheng Xin
2	0.2	Jul. 2025	Fix OR from A+	Zheng Xin
3	0.3	Aug. 2025	Fix OR from A+	Zheng Xin
4	0.4	Aug. 2025	Fix OR from A+	Zheng Xin

1 Contents

1. ST Introduction

This chapter presents the ST reference, the reference for the Target of Evaluation (TOE), a TOE overview description and a description of the logical and physical scope of the TOE.

1.1. ST and TOE reference

Table 1 Description of ST reference and TOE reference

ST reference:	ASE EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C, version 0.4, Aug. 2025.
TOE reference:	THC80F480C/384C/340C/280C/256C/228C/176C/150C Secure Microcontroller Version 1.0

1.2. TOE overview

1.2.1. TOE

The TOE is a secure Microcontroller is dedicated to applications such as for mobile phones (GSM SIM cards) etc.

The TOE consists of hardware. The hardware is based on a 32-bit secure CPU with NVM (Non-volatile Programmable Memory) and RAM (Volatile Memory). The hardware of the TOE also incorporates communication peripherals and true random number generator.

The TOE supports the following communication interfaces:

- ISO/IEC 7816 contact interface.
- I2C interface

The TOE has been designed to provide a platform for Security IC Embedded Software which ensures that the critical user data of the Composite TOE are stored and processed in a secure way. To this end the TOE has the following security features:

- True Random Number Generator
- Protection against power analysis
- Protection against physical attacks
- Protection against perturbation attacks

1.2.2. Non-TOE

The TOE is delivered to a composite product manufacturer. The security IC embedded software is developed by the composite product manufacturer. The security IC embedded software is not part of the TOE.

The Deterministic Random Number Generator hardware component is used internally by the TOE. However, the service provided to the user is not under the scope of the evaluation.

1.3. TOE description

This section presents the physical and logical scope of the TOE.

1.3.1. Physical architecture

The main functional blocks of the TOE hardware are depicted below.

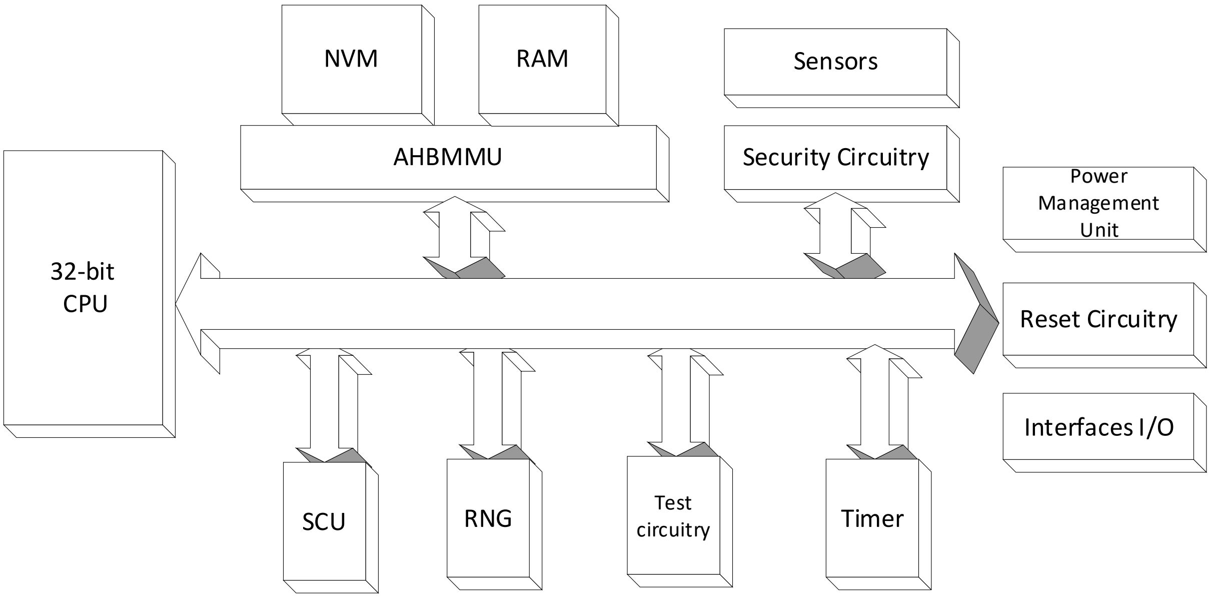


Figure 1 The block diagram of the TOE hardware

The hardware of the TOE has the following components:

- 32-bit secure core CPU
- NVM
- RAM
- AHBMMU
- Interfaces I/O
  - ISO/IEC 7816 contact interface
  - I2C interface
- True Random Number Generator
- SCU
- Test circuitry
- Timers
- Security Circuitry
- Sensors
  - Voltage sensor
  - Glitch sensor
  - Frequency sensor
  - Temperature sensor
  - Light sensor
- Power Management Unit
- Reset circuitry

The AHBMMU is a bus component which also provides user controllable bus masking.

1.3.2. Logical Scope

The TOE distinguishes three modes:

1. Test mode
2. ENG mode
3. Normal mode

Test mode is also not available for the Security IC embedded software. It is utilized to perform the TOE testing before the TOE is delivered to the end user. Test mode is strictly protected by a combination of hardware and software security features.

ENG mode is used for reading SN information for engineering analysis.

Normal mode is utilized for the end user, Security IC embedded software can be executed under this mode. Normal mode cannot switch back to and test mode.

The Memory management unit is performed by the MMU, and it also performs the access control of test mode,ENG mode and normal mode.

There are two communication interfaces available, including ISO/IEC 7816 contact interface, and I2C interface.

The TOE provides the system control functions to handle the reset, etc.

The TOE provides the timers for the security IC embedded software to abort irregular executions of the program.

The TOE provides strong security functionalities against malfunction, including the environmental sensors to monitor if environmental conditions are within the specified range.

The abnormality check of TRNG to verify the quality of the generated random data, also the integrity to monitor if the data is manipulated.

The TOE provides strong security functionalities against leakage, including memory encryption, bus masking .

The TOE provides strong security functionalities against physical manipulation and probing, including (1) the dedicated shielding techniques, (2) the memory and bus encryption.

The TOE provides strong security functionalities against abuse of functionality and identification by the means of test access control mechanism. It is implemented by a combination with hardware fuse and software access control mechanism.

The TOE provides a true random number generator, which is composed of entropy sources, self-test circuit and post-processing circuit. The self-test circuit includes the total failure test and online test. The total failure test is performed on the entropy source. The on line testing is performed on the raw random number sequence, aiming to prevent malfunctioning. The true random number also fulfils the AIS20/31 PTG.2 level.

1.3.3. TOE components

The TOE consists of the following components that are delivered to the composite product manufacturer:

Table 2 List of TOE components

Type	Name	Version	Package	Format	Delivery method	Memory Size(NVM)	Hash value
Hardware	THC80F480C	1.0	Module	Module	Courier delivery	480KB	-
	THC80F384C	1.0	Module	Module	Courier delivery	384KB	-
	THC80F340C	1.0	Module	Module	Courier delivery	340KB	-
	THC80F280C	1.0	Module	Module	Courier delivery	280KB	-
	THC80F256C	1.0	Module	Module	Courier delivery	256KB	-
	THC80F228C	1.0	Module	Module	Courier delivery	228KB	-
	THC80F176C	1.0	Module	Module	Courier delivery	176KB	-
	THC80F150C	1.0	Module	Module	Courier delivery	150KB	-
Document	AGD_OPE OG EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C [1]	0.7	Document	.pdf	Encrypted e-mail	-	See chapter 9
	AGD_OPE SG EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C [2]	1.9	Document	.pdf	Encrypted e-mail	-	See chapter 9
	AGD_PRE EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C [3]	1.1	Document	.pdf	Encrypted e-mail	-	See chapter 9

Note: The hardware modules are the same design, the difference between them is due to the factory configuration of the corresponding registers, which cannot be written or read by users and cannot be altered afterward.

1.4. Life cycle and delivery

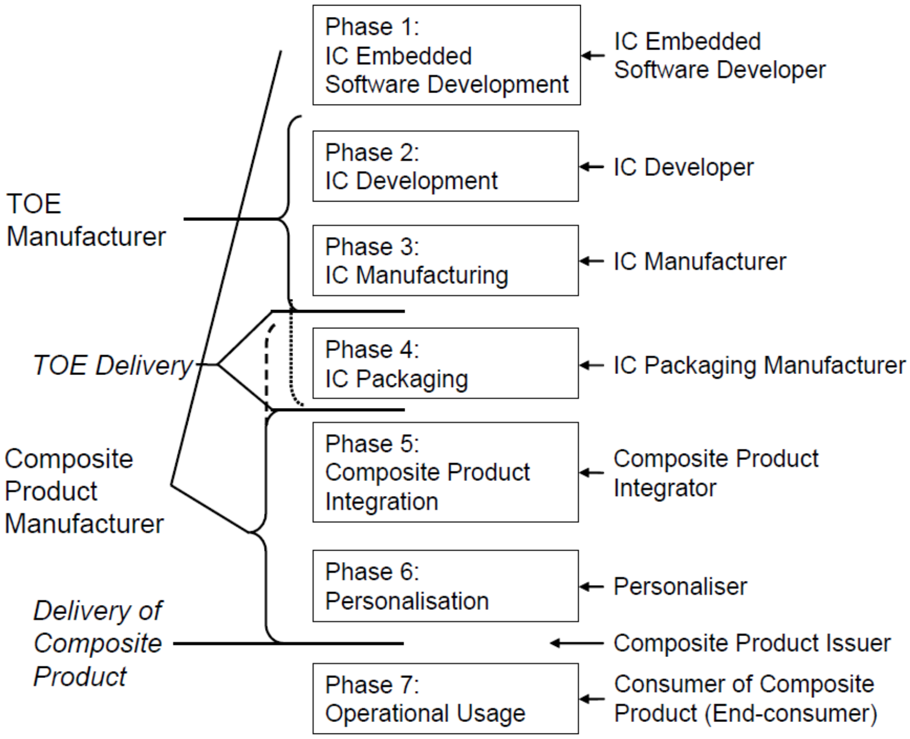


Figure 2: Definition of "TOE Delivery" and responsible Parties

The end-consumer environment of the TOE is phase 7 of the Security IC product life-cycle as defined in figure 2. In this phase the TOE is in usage by the end-consumer. Its method of use now depends on the Security IC Embedded Software. Examples of use cases are eSIM or eSE.

The scope of the assurance components referring to the TOE's life cycle is limited to phases 2, 3 and 4. These phases are under the control of the TOE manufacturer. At the end of phase 4 the TOE components described in 1.3.3 are delivered to the Composite Manufacturer.

2. Conformance claim

This chapter presents conformance claim and the conformance claim rationale.

2.1. CC Conformance

This Security Target and the TOE claim to be conformant to the Common Criteria version 3.1:

- Part 1 revision 5 [4]
- Part 2 revision 5 [5]
- Part 3 revision 5 [6]

For the evaluation will be used the methodology in Common Criteria Evaluation Methodology version 3.1 CEM revision 5 [7]

This Security Target and the TOE claim to be CC Part 2 extended and CC Part 3 conformant.

2.2. PP Claim

This ST does not claim conformance to any other PP.

2.3. Package claim

This Security Target claims conformance to the assurance package **EAL4** augmented with AVA_VAN.4 and ALC_DVS.2.

2.4. Conformance claim rationale

The TOE is a Security IC equivalent to the TOE type defined in [1] as it is composed by:

- Processing unit (32-bit secure CPU)
- Security components (e.g. sensors)
- I/O ports (ISO 7816 and I2C interfaces)
- Volatile memory (e.g. RAM)
- Non-Volatile memory (e.g. NVM)

3. Security problem definition

This chapter presents the assets,the threats, organisational security policies and assumptions for the TOE.

3.1. Description of Assets

Assets regarding the Threats

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

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

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

SC1 confidentiality of user data of the Composite TOE being stored in the TOE's protected memory areas,

SC2 correct operation of the security services provided by the TOE for the Security IC Embedded Software.

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

SC3 deficiency of random numbers.

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

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

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

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

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

3.2. Threats

The following explanations help to understand the focus of the threats and objectives defined below. For example, certain attacks are only one step towards a disclosure of assets, others may directly lead to a compromise of the application security.

- Manipulation of user data (which includes user data and code of the Composite TOE, stored in or processed by the Security IC) means that an attacker is able to alter a meaningful block of data. This should be considered for the threats T.Malfunction, T.Phys-Manipulation and T.Abuse-Func.
- Disclosure of user data (which may include user data and code of the Composite TOE, stored in protected memory areas or processed by the Security IC) or TSF data means that an attacker is realistically[1] able to determine a meaningful block of data. This should be considered for the threats T.Leak-Inherent, T.Phys-Probing, T.Leak-Forced and T.Abuse-Func.
- Manipulation of the TSF or TSF data means that an attacker is able to deliberately deactivate or otherwise change the behaviour of a specific security functionality in a manner which enables exploitation. This should be considered for the threat T.Malfunction, T.Phys-Manipulation and T.Abuse-Func.

The cloning of the functional behaviour of the Security IC on its physical and command interface is the highest level security concern in the application context.
The cloning of that functional behaviour requires to (i) develop a functional equivalent of the Security IC Embedded Software, (ii) disclose, interpret and employ the user data of the Composite TOE stored in the TOE, and (iii) develop and build a functional equivalent of the Security IC using the input from the previous steps.

The Security IC is a platform for the Security IC Embedded Software which ensures that especially the critical user data are stored and processed in a secure way (refer to below). The Security IC Embedded Software must also ensure that critical user data are treated as required in the application context. In addition, the personalisation process supported by the Security IC Embedded Software (and perhaps by the Security IC in addition) must be secure. This last step is beyond the scope of this Security Target. As a result the threat “cloning of the

[1] taking into account the assumed attack potential (and for instance the probability of errors)

functional behaviour of the Security IC on its physical and command interface" is averted by the combination of mechanisms which split into those being evaluated according to the Security IC and those being subject to the evaluation of the Security IC Embedded Software or Security IC and the corresponding personalisation process. Therefore, functional cloning is indirectly covered by the security concerns and threats described below.

The high-level security concerns are refined below by defining threats as required by the Common Criteria (refer to Figure 3). Note that manipulation of the TOE is only a means to threaten user data and is not a success for the attacker in itself.

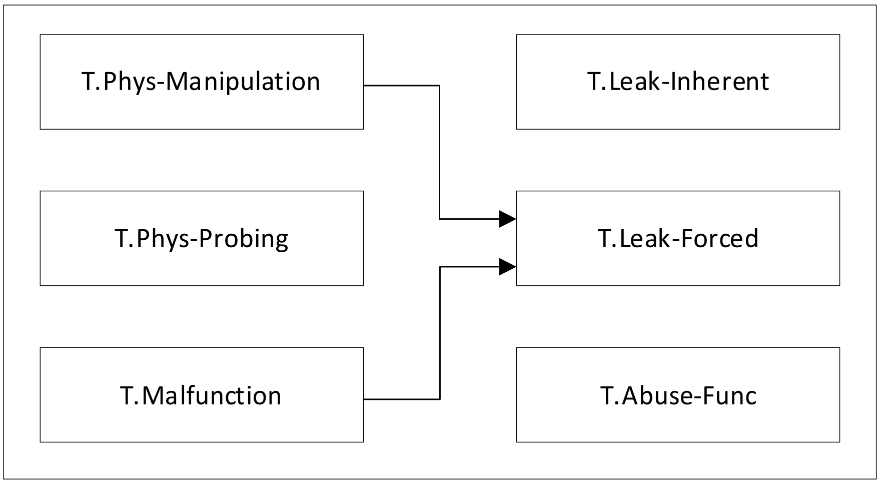


Figure 3: Standard Threats

The high-level security concern related to security service is refined below by defining threats as required by the Common Criteria (refer to Figure 4).

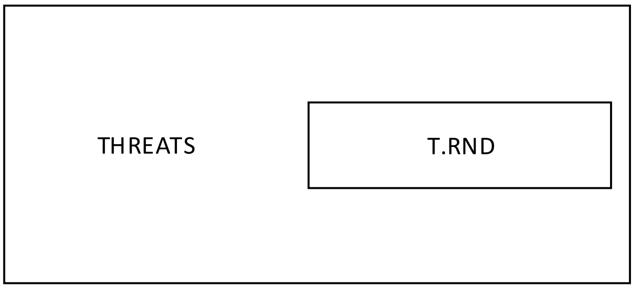


Figure 4: Threats related to security service

The Security IC Embedded Software may be required to contribute to averting the threats. At least it must not undermine the security provided by the TOE.

The above security concerns are derived from considering the operational usage by the end-consumer (Phase 7) since

- Phase 1 and the Phases from TOE Delivery up to the end of Phase 6 are covered by assumptions and
- the development and production environment starting with Phase 2 up to TOE Delivery are covered by an organisational security policy.

The TOE’s countermeasures are designed to avert the threats described below. Nevertheless, they may be effective in earlier phases (Phases 4 to 6).

The TOE is exposed to different types of influences or interactions with its outer world. Some of them may result from using the TOE only but others may also indicate an attack. The different types of influences or interactions are visualised in Figure 5. Due to the intended usage of the TOE all interactions are considered as possible.

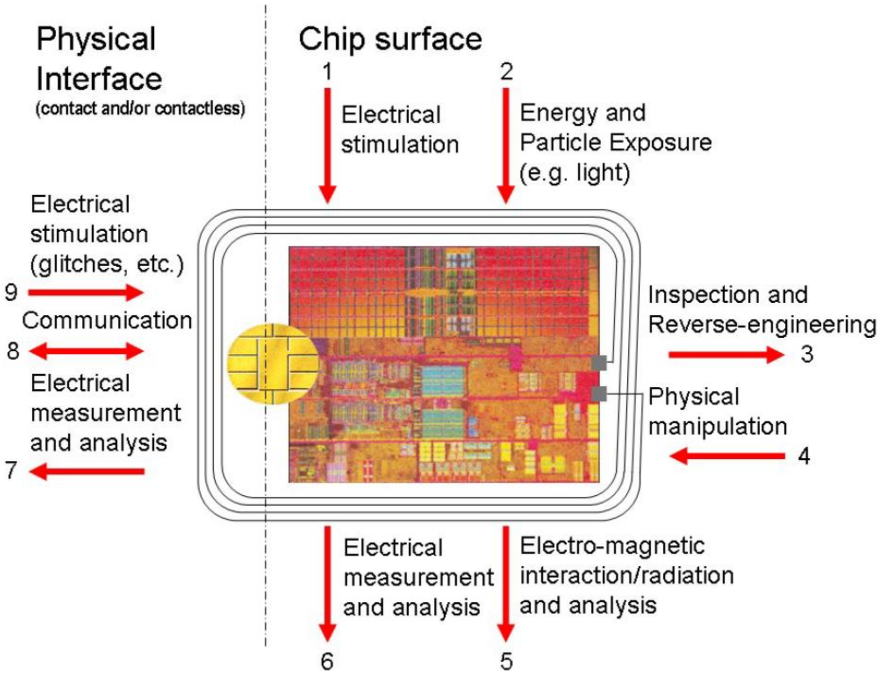

Physical Interface
(contact and/or contactless)
Chip surface
1
Electrical stimulation
2
Energy and Particle Exposure (e.g. light)
Electrical stimulation (glitches, etc.)
9
Communication
8
Electrical measurement and analysis
7
Inspection and Reverse-engineering
3
Physical manipulation
4
Electrical measurement and analysis
6
Electro-magnetic interaction/radiation and analysis
5

Figure 5: Threats related to security service

An interaction with the TOE can be done through the physical interfaces (Number 7 – 9 in Figure 5) which are realised using contacts interface. Influences or interactions with the TOE also occurs through the chip surface (Number 1 – 6 in Figure 5). In Number 1 and 6 galvanic contacts are used. In Number 2 and 5 the influence (arrow directed to the chip) or the measurement (arrow starts from the chip) does not require a contact. Number 3 and 4 refer to specific situations where the TOE and its functional behaviour is not only influenced but definite changes are made by applying mechanical, chemical and other methods (such as 1, 2). Many attacks require a prior inspection and some reverse-engineering (Number 3). This demonstrates the basic building blocks of attacks. A practical attack will use a combination of these elements.

Standard Threats

The TOE shall avert the threat “Inherent Information Leakage (T.Leak-Inherent)” as specified below.

T.Leak-Inherent Inherent Information Leakage

An attacker may exploit information which is leaked from the TOE during usage of the Security IC in order to disclose confidential user data as part of the assets.

No direct contact with the Security IC internals is required here. Leakage may occur through emanations, variations in power consumption, I/O characteristics, clock frequency, or by changes in processing time requirements. One example is Differential Power Analysis (DPA). This leakage may be interpreted as a covert channel transmission but is more closely related to measurement of operating parameters, which may be derived either from direct (contact) measurements (Numbers 6 and 7 in Figure 5) or measurement of emanations (Number 5 in Figure 5) and can then be related to the specific operation being performed.

The TOE shall avert the threat "Physical Probing (T.Phys-Probing)" as specified below.

T.Phys-Probing Physical Probing

An attacker may perform physical probing of the TOE in order (i) to disclose user data while stored in protected memory areas, (ii) to disclose/reconstruct the user data while processed or (iii) to disclose other critical information about the operation of the TOE to enable attacks disclosing or manipulating the user data of the Composite TOE or the Security IC Embedded Software.

Physical probing requires direct interaction with the Security IC internals (Numbers 5 and 6 in Figure 5). Techniques commonly employed in IC failure analysis and IC reverse engineering efforts may be used. Before that hardware security mechanisms and layout characteristics need to be identified (Number 3 in Figure 5). Determination of software design including treatment of user data of the Composite TOE may also be a pre-requisite.

This pertains to "measurements" using galvanic contacts or any type of charge interaction whereas manipulations are considered under the threat "Physical Manipulation (T.Phys-Manipulation)". The threats "Inherent Information Leakage (T.Leak-Inherent)" and "Forced Information Leakage (T.Leak-Forced)" may use physical probing but require complex signal processing in addition.

The TOE shall avert the threat "Malfunction due to Environmental Stress (T.Malfunction)" as specified below.

T.Malfunction Malfunction due to Environmental Stress

An attacker may cause a malfunction of TSF or of the Security IC Embedded Software by applying environmental stress in order to (i) modify security services of the TOE or (ii) modify functions of the Security IC Embedded Software (iii) deactivate or affect security mechanisms of the TOE to enable attacks disclosing or manipulating the user data of the Composite TOE or the Security IC Embedded Software. This may be achieved by operating the Security IC outside the normal operating conditions (Numbers 1, 2 and 9 in Figure 5).

The modification of security services of the TOE may e.g. affect the quality of random numbers provided by the random number generator up to undetected deactivation when the random number generator does not produce random numbers and the Security IC Embedded Software gets constant values. In another case errors are introduced in executing the Security IC Embedded Software. To exploit this an attacker needs information about the functional operation, e.g. to introduce a temporary failure within a register used by the Security IC Embedded Software with light or a power glitch.

The TOE shall avert the threat "Physical Manipulation (T.Phys-Manipulation)" as specified below.

T.Phys-Manipulation Physical Manipulation

An attacker may physically modify the Security IC in order to (i) modify user data of the Composite TOE, (ii) modify the Security IC Embedded Software, (iii) modify or deactivate security services of the TOE, or (iv) modify security mechanisms of the TOE to enable attacks disclosing or manipulating the user data of the Composite TOE or the Security IC Embedded Software.

The modification may be achieved through techniques commonly employed in IC fail-ure analysis (Numbers 1, 2 and 4 in Figure 5) and IC reverse engineering efforts (Number 3 in Figure 5). The modification may result in the deactivation of a security feature. Before that hardware security mechanisms and layout characteristics need to be identified. Determination of software design including treatment of user data of the Composite TOE may also be a prerequisite. Changes of circuitry or data can be permanent or temporary.

In contrast to malfunctions (refer to T.Malfunction) the attacker requires to gather significant knowledge about the TOE's internal construction here (Number 3 in Figure 5).

The TOE shall avert the threat "Forced Information Leakage (T.Leak-Forced)" as specified below:

T.Leak-Forced Forced Information Leakage

An attacker may exploit information which is leaked from the TOE during usage of the Security IC in order to disclose confidential user data of the Composite TOE as part of the assets even if the information leakage is not inherent but caused by the attacker.

This threat pertains to attacks where methods described in "Malfunction due to Environmental Stress" (refer to T.Malfunction) and/or "Physical Manipulation" (refer to T.Phys-Manipulation) are used to cause leakage from signals (Numbers 5, 6, 7 and 8 in Figure 5) which normally do not contain significant information about secrets.

The TOE shall avert the threat "Abuse of Functionality (T.Abuse-Func)" as specified below.

T.Abuse-Func Abuse of Functionality

An attacker may use functions of the TOE which may not be used after TOE Delivery in order to (i) disclose or manipulate user data of the Composite TOE, (ii) manipulate (explore, bypass, deactivate or change) security services of the TOE or (iii) manipulate (explore, bypass, deactivate or change) functions of the Security IC Embedded Software or (iv) enable an attack disclosing or manipulating the user data of the Composite TOE or the Security IC Embedded Software.

Threats related to security services
The TOE shall avert the threat "Deficiency of Random Numbers (T.RND)" as specified below.

T.RND Deficiency of Random Numbers

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

An attacker may gather information about the random numbers produced by the TOE security service. Because unpredictability is the main property of random numbers this may be a problem in case they are used to generate cryptographic keys. The entropy provided by the random numbers must be appropriate for the strength of the cryptographic algorithm, the key or the cryptographic variable is used for. Here the attacker is expected to take advantage of statistical properties of the random numbers generated by the TOE. Malfunctions or premature ageing are also considered which may assist in getting information about random numbers.

3.3. Organisational security policies

The following P.Process shows the policies applied in this Security Target.

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

An accurate identification must be established for the TOE.This requires that each instantiation of the TOE carries this unique identification.
The accurate identification is introduced at the end of the production test in phase 3.Therefore the production environment must support this unique identification.

3.4. Assumptions

The following Figure 6 shows the assumptions applied in this Security Target.

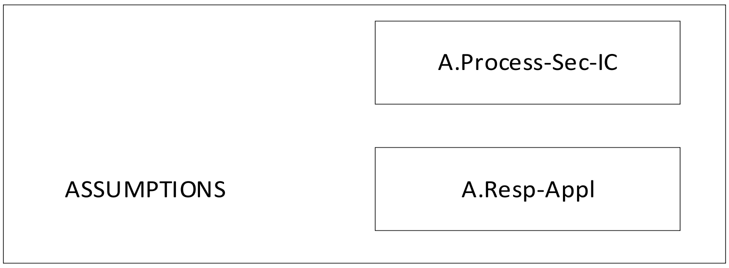


Figure 6: Assumptions

The intended usage of the TOE is twofold, depending on the Life Cycle Phase: (i) The Security IC Embedded Software developer uses it as a platform for the Security IC software being developed. The Composite Product Manufacturer (and the consumer) uses it as a part of the Security IC. The Composite Product is used in a terminal which supplies the Security IC (with power and lock) and (at least) mediates the communication with the Security IC Embedded Software.

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

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

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

This means that the Phases after TOE Delivery are assumed to be protected appropriately.

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

- the Security IC Embedded Software including specifications, implementation and related documentation,
- Pre-personalisation Data and Personalisation Data including specifications of formats and memory areas, test related data,

- the user data of the Composite TOE and related documentation, and
- material for software development support

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

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

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

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

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

4. Security objectives

This chapter provides the statement of security objectives and the security objective rationale.

4.1. Security Objectives for the TOE

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

SG1 maintain the confidentiality of user data (when being processed and when being stored in the TOE's protected memories).

SG2 maintain the correct operation of the security services provided by the TOE for the Security IC Embedded Software.

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

These standard high-level security goals in the context of the security problem definition build the starting point for the definition of security objectives as required by the Common Criteria (refer to Figure 7). Note that the integrity of the TOE is a means to reach these objectives.

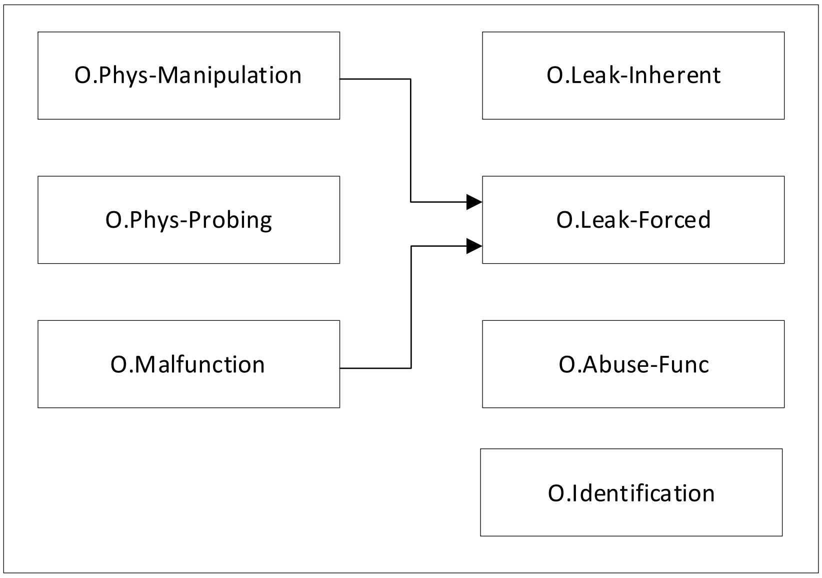


Figure 7 : Standard Security Objectives

There is the following high-level security goal related to specific functionality:

SG3 provide true random numbers.

The additional high-level security considerations are refined below by defining security objectives as required by the Common Criteria (refer to Figure 8).

Security Objectives
O.RND

Figure 8: Security Objectives related to Specific Functionality

Standard Security Objectives

The TOE shall provide "Protection against Inherent Information Leakage (O.Leak-Inherent)" as specified below.

O.Leak-Inherent Protection against Inherent Information Leakage

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

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

This objective pertains to measurements with subsequent complex signal processing whereas O.Phys-Probing is about direct measurements on elements on the chip surface. Details correspond to an analysis of attack scenarios which is not given here.

The TOE shall provide "Protection against Physical Probing (O.Phys-Probing)" as specified below.

O.Phys-Probing Protection against Physical Probing

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

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

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

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

The TOE must be designed and fabricated so that it requires a high combination of complex equipment, knowledge, skill, and time to be able to derive detailed design information or other information which could be used to compromise security through such a physical attack.

The TOE shall provide "Protection against Malfunctions (O.Malfunction)" as specified below.

O.Malfunction Protection against Malfunctions

The TOE must ensure its correct operation.

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

Remark: A malfunction of the TOE may also be caused using a direct interaction with elements on the chip surface. This is considered as being a manipulation (refer to the objective O.Phys-Manipulation) provided that detailed knowledge about the TOE´s internal construction is required and the attack is performed in a controlled manner.

The TOE shall provide "Protection against Physical Manipulation (O.Phys-Manipulation)" as specified below.

O.Phys-Manipulation Protection against Physical Manipulation

The TOE must provide protection against manipulation of the TOE (including its software and TSF data), the Security IC Embedded Software and the user data of the Composite TOE. This includes protection against

- reverse-engineering (understanding the design and its properties and functions),
- manipulation of the hardware and any data, as well as
- undetected manipulation of memory contents.

The TOE must be designed and fabricated so that it requires a high combination of complex equipment, knowledge, skill, and time to be able to derive detailed design information or other information which could be used to compromise security through such a physical attack.

The TOE shall provide “Protection against Forced Information Leakage (O.Leak-Forced)“ as specified below:

O.Leak-Forced — Protection against Forced Information Leakage

The Security IC must be protected against disclosure of confidential data processed in the Security IC (using methods as described under O.Leak-Inherent) even if the information leakage is not inherent but caused by the attacker

- by forcing a malfunction (refer to “Protection against Malfunction due to Environmental Stress (O.Malfunction)” and/or
- by a physical manipulation (refer to “Protection against Physical Manipulation (O.Phys-Manipulation)”.

If this is not the case, signals which normally do not contain significant information about secrets could become aninformation channel for a leakage attack.

The TOE shall provide “Protection against Abuse of Functionality (O.Abuse-Func)” as specified below.

O.Abuse-Func — Protection against Abuse of Functionality

The TOE must prevent that functions of the TOE which may not be used after TOE Delivery can be abused in order to (i) disclose critical user data of the Composite TOE, (ii) manipulate critical user data of the Composite TOE, (iii) manipulate Security IC Embedded Software or (iv) bypass, deactivate, change or explore security features or security services of the TOE. Details depend, for instance, on the capabilities of the Test Features provided by the IC Dedicated Test Software which are not specified here.

The TOE shall provide “TOE Identification (O.Identification)“ as specified below:

O.Identification — TOE Identification

The TOE must provide means to store Initialisation Data and Pre-personalisation Data in its non-volatile memory. The Initialisation Data (or parts of them) are used for TOE identification.

Security Objectives related to Specific Functionality (referring to SG3)

The TOE shall provide “Random Numbers (O.RND)” as specified below.

O.RND — Random Numbers

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

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

4.2. Security Objectives for the Security IC Embedded Software

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

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

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

OE.Resp-Appl Treatment of user data of the Composite TOE

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

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

4.3. Security Objectives for the operational Environment

TOE Delivery up to the end of Phase 6

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

OE.Process-Sec-IC Protection during composite product manufacturing

Security procedures shall be used after TOE Delivery up to delivery to the end-consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use).

This means that Phases after TOE Delivery up to the end of Phase 6 must be protected appropriately.

4.4. Security Objectives Rationale

Table 3 below gives an overview, how the assumptions, threats, and organisational security policies are addressed by the objectives. The text following after the table justifies this in detail.

Table 3 Security Objectives versus Assumptions, Threats or Policies

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

The justification related to the assumption “Treatment of user data of the Composite TOE (A.Resp-Appl)” is as follows:
Since OE.Resp-Appl requires the Security IC Embedded Software to implement measures as assumed in A.Resp-Appl, the assumption is covered by the objective.

The justification related to the organisational security policy “Protection during TOE Development and Production (P.Process-TOE)” is as follows:
O.Identification requires that the TOE has to support the possibility of a unique identification. The unique identification can be stored on the TOE. Since the unique identification is generated by the production environment the production environment must support the integrity of the generated unique identification. The technica and organisational security measures that ensure the security of the development environment and production environment are evaluated based on the assurance measures that are part of the evaluation. For a list of material produced and processed by the TOE Manufacturer. All listed items and the associated development and production environments are subject of the evaluation. Therefore, the organisational security policy P.Process-TOE is covered by this objective, as far as organisational measures are concerned.

The justification related to the assumption “Protection during Packaging, Finishing and Personalisation (A.Process-Sec-IC)” is as follows:
Since OE.Process-Sec-IC requires the Composite Product Manufacturer to implement those measures assumed in A.Process-Sec-IC, the assumption is covered by this objective.

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

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

5. Extended Components Definitions

This Security Target uses the extended components definitions contains: definition of the family FCS_RNG, FMT_LIM, FAU_SAS, FDP_SDC.

5.1. Definition of the Family FCS_RNG

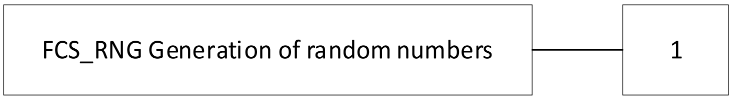
To define the IT security functional requirements of the TOE an additional family (FCS_RNG) of the Class FCS (cryptographic support) is defined here. This family describes the functional requirements for random number generation used for cryptographic purposes.

FCS_RNG Generation of random numbers

Family behaviour

This family defines quality requirements for the generation of random numbers which are intended to be use for cryptographic purposes.

Component levelling:



FCS_RNG.1 Generation of random numbers requires that random numbers meet a defined quality metric.

Management: FCS_RNG.1
There are no management activities foreseen.

Audit: FCS_RNG.1
There are no actions defined to be auditable.

FCS_RNG.1 Random number generation

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1 The TSF shall provide a [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*] random number generator that implements: [assignment: *list of security capabilities*].

FCS_RNG.1.2 The TSF shall provide [selection: *bits, octets of bits, numbers [assignment: format of the numbers]*] that meet [assignment: *a defined quality metric*].

5.2. Definition of the Family FMT_LIM

To define the IT security functional requirements of the TOE an additional family (FMT_LIM) of the Class FMT (Security Management) is defined here. This family describes the functional requirements for the Test Features of the TOE. The new functional requirements were defined in the class FMT because this class addresses the management of functions of the TSF. The examples of the technical mechanism used in the TOE appropriate to address the specific issues of preventing the abuse of functions by limiting the capabilities of the functions and by limiting their availability.

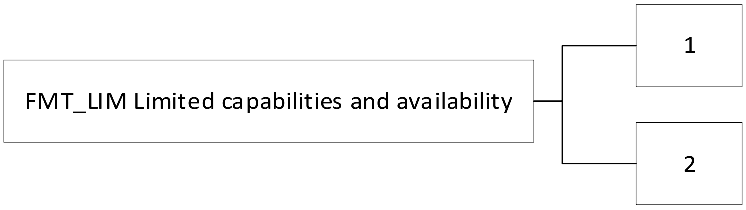
The family "Limited capabilities and availability (FMT_LIM)" is specified as follows.

FMT_LIM Limited capabilities and availability

Family behaviour

This family defines requirements that limit the capabilities and availability of functions in a combined manner. Note that FDP_ACF restricts the access to functions whereas the component Limited Capability of this family requires the functions themselves to be designed in a specific manner.

Component levelling:



FMT_LIM.1 Limited capabilities requires that the TSF is built to provide only the capabilities (perform action, gather information) necessary for its genuine purpose.

FMT_LIM.2 Limited availability requires that the TSF restrict the use of functions (FMT_LIM.1). This can be achieved, for instance, by removing or by disabling functions in a specific phase of the TOE's life-cycle.

Management: FMT_LIM.1, FMT_LIM.2

There are no management activities foreseen.

Audit: FMT_LIM.1, FMT_LIM.2

There are no actions defined to be auditable.

The TOE Functional Requirement "Limited capabilities (FMT_LIM.1)" is specified as follows.

FMT_LIM.1 **Limited capabilities**

Hierarchical to: No other components.

Dependencies: FMT_LIM.2 Limited availability.

FMT_LIM.1.1 The TSF shall be designed and implemented in a manner that limits its capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced [assignment: *Limited capability policy*].

The TOE Functional Requirement "Limited availability (FMT_LIM.2)" is specified as follows.

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 [assignment: *Limited availability policy*].

Application Note:The functional requirements FMT_LIM.1 and FMT_LIM.2 assume that there are two types of mechanisms (limitation of capabilities and limitation of availability) which together shall provide protection in order to enforce the same policy or two mutual supportive policies related to the same functionality. This allows e.g. that
(i)the TSF is provided without restrictions in the product in its userenvironment but its capabilities are so limited that the policy is enforced
or conversely
(ii)the TSF is designed with high functionality but is removed or disabled in the product in its user environment.

5.3. Definition of the Family FAU_SAS

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

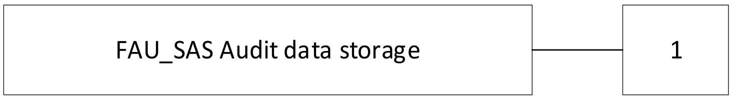
The family “Audit data storage (FAU_SAS)” is specified as follows.

FAU_SAS Audit data storage

Family behaviour

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

Component levelling



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

Management: FAU_SAS.1
There are no management activities foreseen.

Audit: FAU_SAS.1
There are no actions defined to be auditable.

FAU_SAS.1 **Audit storage**

Hierarchical to: No other components.

Dependencies: No dependencies.

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

5.4. Definition of the Family FDP_SDC

To define the security functional requirements of the TOE an additional family (FDP_SDC.1) of the Class FDP (User data protection) is defined here.

The family "Stored data confidentiality (FDP_SDC)" is specified as follows.

FDP_SDC Stored data confidentiality

Family behaviour

This family provides requirements that address protection of user data confidentiality while these data are stored within memory areas protected by the TSF. The TSF provides access to the data in the memory through the specified interfaces only and prevents compromise of their information bypassing these interfaces.

Component levelling

FDP_SDC.1 Requires the TOE to protect the confidentiality of information of the user data in specified memory areas.

Management: FDP_SDC.1
There are no management activities foreseen.

Audit: FDP_SDC.1
There are no actions defined to be auditable.

FDP_SDC.1 **Stored data confidentiality**

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP_SDC.1.1 The TSF shall ensure the confidentiality of the information of the user data while it is stored in the [assignment: *memory area*].

6. IT Security Requirements

This chapter IT Security Requirements contains the following sections:
Definitions(6.1)
Security Functional Requirements for the TOE (6.2)
Security Assurance Requirements for the TOE (6.3)
Security Requirements Rationale (6.4)

6.1. Definitions

In the next sections the following notation is used:

- The iteration operation is used when a component is claimed with varying operations, it is denoted by adding "[XXX]" to the component name.
- Selection or assignment operations are used to add details or assign specific values to components, they are indicated by italic text and explained in footnotes.
- The Refinement is pointed out by using the **bold type**.

6.2. Security Functional Requirements for the TOE

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

Malfunctions

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

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

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

FPT_FLS.1	**Failure with preservation of secure state**
Hierarchical to:	No other components.
Dependencies:	No dependencies.

[1] [assignment: list of types of failures]

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: *exposure to operating conditions which may not be tolerated according to the requirement Limited fault tolerance (FRU_FLT.2) and where therefore a malfunction could occur*[2].

Refinement: **The term "failure" above also covers "abnormal voltage, abnormal temperature, and power glitch". The TOE prevents failures for the "abnormal voltage, abnormal temperature, and power glitch" defined above.**

Abuse of Functionality

The TOE shall meet the requirement "Limited capabilities (FMT_LIM.1)" as specified below (Common Criteria Part 2 extended).

FMT_LIM.1 **Limited capabilities**

Hierarchical to: No other components.

Dependencies: FMT_LIM.2 Limited availability.

FMT_LIM.1.1 The TSF shall be designed and implemented in a manner that limits their capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced: *Deploying Test Features after TOE Delivery does not allow user data of the Composite TOE to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks*[3].

The TOE shall meet the requirement "Limited availability (FMT_LIM.2)" as specified below (Common Criteria Part 2 extended).

FMT_LIM.2 **Limited availability**

Hierarchical to: No other components.

Dependencies: FMT_LIM.1 Limited capabilities.

FMT_LIM.2.1 The TSF shall be designed and implemented in a manner that limits their availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced: *Deploying Test Features after TOE Delivery does not allow user data of the Composite TOE to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks*[4].

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

[2] [assignment: list of types of failures in the TSF]

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

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

FAU_SAS.1 **Audit storage**

Hierarchical to: No other components.

Dependencies: No dependencies.

FAU_SAS.1.1 The TSF shall provide *the test process before TOE Delivery*[5] with the capability to store *the Initialisation Data, Pre-personalisation Data*[6] in the *NVM*[7].

Application Note: The integrity and uniqueness of the unique identification of the TOE must be supported by the development, production and test environment.

Physical Manipulation and Probing

The TOE shall meet the requirement "Stored data confidentiality (FDP_SDC.1)" as specified below (Common Criteria Part 2 extended).

FDP_SDC.1 **Stored data confidentiality**

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP_SDC.1.1 The TSF shall ensure the confidentiality of the information of the user data while it is stored in the *NVM or RAM*[8].

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

FPT_PHP.3 **Resistance to physical attack**

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_PHP.3.1 The TSF shall resist *physical manipulation and physical probing*[9] to the *TSF*[10] by responding automatically such that the SFRs are always enforced.

Refinement: **The TSF will implement appropriate mechanisms to continuously counter physical manipulation and physical probing. Due to the nature of these attacks (especially manipulation) the TSF can by no means detect attacks on all of its elements. Therefore, permanent protection against these attacks is required ensuring that security functional requirements are enforced. Hence, "automatic response" means here (i) assuming that there might be an attack at any time and (ii) countermeasures are provided at any time.**

Application note: If a physical manipulation or physical probing attack is detected, an alarm will be automatically triggered by the hardware, which will cause the chip to be reset.

[5] [assignment: list of subjects]
[6] [assignment: list of audit information]
[7] [assignment: type of persistent memory]
[8] [assignment: memory area]
[9] [assignment: physical tampering scenarios]
[10] [assignment: list of TSF devices/elements]

Leakage

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

FDP_ITT.1	**Basic internal transfer protection**
Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_ITT.1.1	The TSF shall enforce the *Data Processing Policy*[11] to prevent the *disclosure*[12] of user data when it is transmitted between physically-separated parts of the TOE.
Refinement:	**The different memories, the CPU and other functional units of the TOE (e.g. a random number generator) are seen as physically-separated parts of the TOE.**

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

FPT_ITT.1	**Basic internal TSF data transfer protection**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_ITT.1.1	The TSF shall protect TSF data from *disclosure*[13] when it is transmitted between separate parts of the TOE.
Refinement:	**The different memories, the CPU and other functional units of the TOE (e.g. a random number generator) are seen as separated parts of the TOE.**

This requirement is equivalent to FDP_ITT.1 above but refers to TSF data instead of user data. Therefore, it should be understood as to refer to the *same Data Processing Policy* defined under FDP_IFC.1 below.

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

FDP_IFC.1	**Subset information flow control**
Hierarchical to:	No other components.
Dependencies:	FDP_IFF.1 Simple security attributes
FDP_IFC.1.1	The TSF shall enforce the *Data Processing Policy*[14] on all confidential data when *they are processed or transferred by the TOE or by the Security IC Embedded Software*[15].

[11] [assignment: access control SFP(s) and/or information flow control SFP(s)]
[12] [selection: disclosure, modification, loss of use]
[13] [selection: disclosure, modification]
[14] [assignment: information flow control SFP]
[15] [assignment: list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP]

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

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

Random Numbers

The TOE shall meet the requirement “Quality metric for random numbers (FCS_RNG.1[PTG.2])” as specified below (Common Criteria Part 2 extended).

FCS_RNG.1[PTG.2] Random number generation

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1[PTG.2] The TSF shall provide a *physical*[16] random number generator that implements:

- *A total failure test detects a total failure of entropy source immediately when the RNG has started. When a total failure is detected, no random numbers will be output.*
- *If a total failure of the entropy source occurs while the RNG is being operated, the RNG prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source.*
- *The online test shall detect non-tolerable statistical defects of the raw random number sequence (i) immediately when the RNG has started. And (ii) while the RNG is being operated. The TSF must not output any random numbers before the power-up online test has finished successfully or when a defect has been detected.*
- *The online test procedure shall be effective to detect non-tolerable weakness of the random numbers soon.*
- *The online test procedure checks the quality of the raw random number sequence. It is triggered applied upon specified internal events. 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.*[17]

FCS_RNG.1.2[PTG.2] The TSF shall provide 32-bit random number words[18] that meet:

- *Test procedure A and no other test suites does not distinguish the internal random numbers from output sequences of an ideal RNG.*

[16][selection: physical, non-physical true, deterministic, hybrid physical, hybrid deterministic]
[17][assignment: list of security capabilities]
[18][selection: bits, octets of bits, numbers [assignment: format of the numbers]]

- *The average Shannon entropy per internal random bit exceeds 0.997*[19].

6.3. Security Assurance Requirements for the TOE

The Security Target will be evaluated according to

Security Target evaluation (Class ASE)

The Security Assurance Requirements for the TOE for the evaluation of the TOE are those taken from the

Evaluation Assurance Level 4 (EAL4)

and augmented by taking the following components:

ALC_DVS.2, and AVA_VAN.4

The assurance requirements are:

Class ADV: Development

Architectural design	(ADV_ARC.1)
Functional specification	(ADV_FSP.4)
Implementation representation	(ADV_IMP.1)
TOE design	(ADV_TDS.3)

Class AGD: Guidance documents

Operational user guidance	(AGD_OPE.1)
Preparative user guidance	(AGD_PRE.1)

Class ALC: Life-cycle support

CM capabilities	(ALC_CMC.4)
CM scope	(ALC_CMS.4)
Delivery	(ALC_DEL.1)
Development security	(ALC_DVS.2)
Life-cycle definition	(ALC_LCD.1)
Tools and techniques	(ALC_TAT.1)

Class ASE: Security Target evaluation

Conformance claims	(ASE_CCL.1)
Extended components definition	(ASE_ECD.1)
ST introduction	(ASE_INT.1)
Security objectives	(ASE_OBJ.2)
Derived security requirements	(ASE_REQ.2)

[19] [assignment: a defined quality metric]

Security problem definition (ASE_SPD.1)

TOE summary specification (ASE_TSS.1)

Class ATE: Tests

Coverage	(ATE_COV.2)
Depth	(ATE_DPT.1)
Functional tests	(ATE_FUN.1)
Independent testing	(ATE_IND.2)

Class AVA: Vulnerability assessment

Vulnerability analysis	(AVA_VAN.4)

6.3.1. Refinements of the TOE Assurance Requirements

The CCDB, the JILWG and the certification bodies publish supporting documents and guidance documents for evaluation and certification of smartcards and similar devices mandatory under CCRA and SOG-IS or the national certification schemes, cf. [8], [9], [10], [11] and [12]. These documents are regularly updated and valid for the running evaluation in their actual versions. The "Supporting Document, Mandatory Technical Document: The Application of CC to Integrated Circuits" provides a comprehensive application of CC to smartcard technology.

The following refinements shall support the comparability of evaluations. Where refinements were not needed some background information based on such documents was provided. In all cases the background information is informative only. The mandatory documents itself shall be consulted for exact details and overrule the refinements in case of any inconsistency (e.g. due to updates).

Refinements regarding Delivery procedure (ALC_DEL)

Refinements regarding Development Security (ALC_DVS)

Refinement regarding CM scope (ALC_CMS)

Refinement regarding CM capabilities (ALC_CMC)

Refinements regarding Security Architecture (ADV_ARC)

Refinements regarding Functional Specification (ADV_FSP)

Refinements regarding Implementation Representation (ADV_IMP)

Refinement regarding Test Coverage (ATE_COV)

Refinement regarding User Guidance (AGD_OPE)

Refinement regarding Preparative User Guidance (AGD_PRE)

Refinement regarding Vulnerability Analysis (AVA_VAN)

The Refinement is pointed out by using the **bold type**. These refinements refer to some keywords within the Security Assurance Requirements that are stressed by underlining.

*Application Note:*The refinements as defined below may also be applicable to a hierarchically higher assurance component of the specific family. If a Security Target includes an additional augmentation, the author of the Security Target has to examine that the refinements as defined below are still applicable.

6.3.1.1Refinements regarding Delivery procedure （ALC_DEL)

Introduction

The Common Criteria assurance component of the family ALC_DEL (delivery procedure) refer to the delivery of (i) the TOE or parts of it (ii) to the user or user's site (Developer of the Security IC Embedded Software or the Composite TOE Manufacturer). The Common Criteria assurance component ALC_DEL.1 requires procedures and technical measures to detect modifications and prevent any compromise of the Initialisation Data and/or Pre-personalisation Data and/or assigned other data.

In the particular case of a Security IC more "material and information" than the TOE itself (which by definition includes the necessary guidance) is exchanged with "users". Therefore, considering the definition of the Common Criteria the following refinement is made regarding the items "TOE" and "to the user or user's site":

The following text reflects the requirements of the selected component ALC_DEL.1:

Developer action elements:

ALC_DEL.1.1D The developer shall document procedures for delivery of the TOE or parts of it to the consumer.

ALC_DEL.1.2D The developer shall use the delivery procedures.

Content and presentation elements:

ALC_DEL.1.1C The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to the consumer.

Evaluator action elements:

ALC_DEL.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

Refinement

For delivery of the TOE to the "Composite Product Manufacturer" as consumer, all the external interfaces of the TOE Manufacturer have to be taken into account. These are:

- **the interface with the Security IC Embedded Software Developer (Phase 1) where information about the Security IC, development software and/or tools for software development and possible information about mask options are exchanged and**

- **the interface with the Phase after TOE Delivery (Phase 4 or 5) where pre- personalisation data, information about tests, and the product in form of wafers, sawn wafers (dice) or packaged products are exchanged.**

Application Note: The consumer in the context of ALC_DEL is the Composite Product Manufacturer to which the TOE as security IC is delivered. The End-consumer is the consumer of the Composite Product which includes the TOE as platform for the IC Embedded Software.

Application Note: All identified critical information about the TOE have to be taken into account in order to avoid any tampering with the actual version or substitution of a false version (including unauthorised modification or replacement).

Application Note: Depending on whether the TOE comprises programmable non-volatile memory, in addition to IC pre-personalisation requirements, the Security IC Embedded Software and/or keys for the authorised personalisation of the programmable non-volatile memory are delivered to the Composite Product Manufacturer.

6.3.1.2Refinements regarding Development Security (ALC_DVS)

Introduction

The JILWG published the document "Joint Interpretation Library: Minimum Site Security Requirements, 2020" [12].

The Common Criteria assurance component of the family ALC_DVS refer (i) to "development environment", (ii) to the "TOE" or "TOE design and implementation". The component ALC_DVS.2 "Sufficiency of security measures" requires additional evidence for the suitability of the security measures.

The TOE Manufacturer must ensure that the development and production of the TOE(refer to Section 1.4) is secure so that no restricted, sensitive, critical or very critical information is unintentionally made available for the operational phase of the TOE which enables or support attacks (cf. [12] for details). Therefore confidentiality and integrity of design information and test data must be guaranteed, access to samples, development tools and other material must be restricted to authorised persons only, scrap must be destroyed. This not only pertains to the TOE but also to all information and material exchanged with the developer of the Security IC Embedded Software and therefore especially to the Security IC Embedded Software itself. This includes the delivery (exchange) procedures for Phase 1 and the Phases after TOE Delivery as far as they can be controlled by the TOE Manufacturer.

In the particular case of a Security IC the TOE is developed and produced within a complex industrial process which must especially be protected. Therefore, the following refinement is made regarding the items "development environment", or "TOE design and implementation" and the confirmation of the application of the security measures:

The following text reflects the requirements of the selected component ALC_DVS.2:
Developer action elements:

ALC_DVS.2.1D The developer shall produce development security documentation.

Content and presentation elements:

ALC_DVS.2.1C The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.

ALC_DVS.2.2C The development security documentation shall justify that the security measures provide the necessary level of protection to maintain the confidentiality and integrity of the TOE.

Evaluator action elements:

ALC_DVS.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ALC_DVS.2.2E The evaluator shall confirm that the security measures are being applied.

Refinement

"TOE design and implementation" must be understood as comprising all material and information related to the development and production of the TOE. Therefore, all critical information identified in Section 3.1 have to be taken into account in order to ensure integrity and – if necessary confidentiality - (including protection against unauthorised disclosure, unauthorised modification or replacement and theft). The "development security documentation" shall describe all security measures related to the "TOE design and implementation" in the development environment as defined above.

Application Note : Whenever samples, material and information is given to external partners (such as the developer of the Security IC Embedded Software) the latter must be obliged by an Non Disclosure Agreement to treat the samples, material and information as it is required for the TOE Manufacturer.

Background information

The scope of the requirement of "Development Security (ALC_DVS)" pertains to the Phase 2 up to TOE Delivery. These phases are under the control of the TOE Manufacturer. The "development environment" as referred to in the Common Criteria covers both, the development (Phase 2) and the production (at least Phase 3, e.g. Phase 4 may be included if the TOE Manufacturer delivers packaged products) of the TOE.

6.3.1.3 Refinement regarding CM scope (ALC_CMS)

Introduction

The Common Criteria assurance component of the family ALC_CMS (CM scope) refers to the tracking of specific configuration items within the developers configuration management system.

In the particular case of a Security IC it is helpful to clarify the scope of the configuration item "TOE implementation representation":

The following text reflects the requirements of the selected component ALC_CMS.4:
Developer action elements:

ALC_CMS.4.1D The developer shall provide a configuration list for the TOE.

Content and presentation elements:

ALC_CMS.4.1C The configuration list includes the following: the TOE itself; the evaluation evidence required by the SARs; the parts that comprise the TOE; the implementation representation; and security flaws reports and resolution status.

ALC_CMS.4.2C The configuration list shall uniquely identify the configuration items.

ALC_CMS.4.3C For each TSF relevant configuration item, the configuration list shall indicate the developer of the item.

Evaluator action elements:

ALC_CMS.4.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

Refinement

The "Security IC Embedded Software" is as user data not part of the TOE but the whole "Security IC Embedded Software" or part of it may be delivered together with the TOE (written by the TOE manufacturer in persistent memory). Therefore the items "Security IC Embedded Software" or "authentication data" are only relevant for the configuration list as far as the TOE manufacturer can control these items. Since the Security IC Embedded Software may be developed by another company it is only available in a specific from and is not part of the TOE though delivered together with it. Authentication data may be required for products implementing programmable non-volatile memory to enable the download of software.

Background information

Depending on the product type with programmable non-volatile memory the Security IC Embedded Software and/or authentication data for a secure loader of the programmable non-volatile memory may be considered as part of the TOE implementation representation.
The "TOE implementation representation" within the scope of the CM will include at least:

- logical design data,
- physical design data,
- IC Dedicated Software,
- final physical design data necessary to produce the photomasks, and
- photomasks.

6.3.1.4Refinement regarding CM capabilities (ALC_CMC)

Introduction

The Common Criteria assurance component of the family ALC_CMC (CM capabilities) refers to the capabilities of a CM system. The component ALC_CMC.4 "Production support, acceptance procedures and automation" refers to "configuration items" and "configuration list" and uses the term "TOE" in addition.

In the particular case of a Security IC the scope of "configuration items" and the meaning of "TOE" in this context need to be clarified:

The following text reflects the requirements of the selected component ALC_CMC.4:
Developer action elements:

ALC_CMC.4.1D The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.4.2D The developer shall provide the CM documentation.

ALC_CMC.4.3D The developer shall use a CM system.

Content and presentation elements:

ALC_CMC.4.1C The TOE shall be labelled with its unique reference.

ALC_CMC.4.2C The CM documentation shall describe the method used to uniquely identify the configuration items.

ALC_CMC.4.3C The CM system shall uniquely identify all configuration items.

ALC_CMC.4.4C The CM system shall provide automated measures such that only authorised changes are made to the configuration items.

ALC_CMC.4.5C The CM system shall support the production of the TOE by automated means.

ALC_CMC.4.6C The CM documentation shall include a CM plan.

ALC_CMC.4.7C The CM plan shall describe how the CM system is used for the development of the TOE.

ALC_CMC.4.8C The CM plan shall describe the procedures used to accept modified or newly created configuration items as part of the TOE.

ALC_CMC.4.9C The evidence shall demonstrate that all configuration items are being maintained under the CM system.

ALC_CMC.4.10C The evidence shall demonstrate that the CM system is being operated in accordance with the CM plan.

Evaluator action elements:

ALC_CMC.4.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

Refinement

"Configuration items" comprise all items defined and refined under ALC_CMS (see above) to be tracked under CM.

A production control system has to be applied to guarantee the traceability and completeness of different production charges or lots. The number of wafers, dies and chips must be tracked by this system. Appropriate administration procedures have to be provided for managing wafers, dies or complete chips, which are being removed from the production-process in order to verify and to control predefined quality standards and production parameters. It has to be controlled that these wafers, dies or assembled devices are returned to the same production stage from which they are taken or they have to be securely stored or destroyed otherwise.

6.3.1.5Refinements regarding Security Architecture (ADV_ARC)

Introduction

The "Supporting Document Guidance Security Architecture requirements (ADV_ARC) for smart cards and similar devices" provides further guidance on how to apply the assurance requirements for the security architecture to security integrated circuits.

The refinement of the Common Criteria assurance component ADV_ARC.1 refers to the following text:

Developer action elements:

ADV_ARC.1.1D The developer shall design and implement the TOE so that the security features of the TSF cannot be bypassed.

ADV_ARC.1.2D The developer shall design and implement the TSF so that it is able to protect itself from tampering by untrusted active entities.

ADV_ARC.1.3D The developer shall provide a security architecture description of the TSF.

Content and presentation elements:

ADV_ARC.1.1C The security architecture description shall be at a level of detail commensurate with the description of the SFR-enforcing abstractions described in the TOE design document.

ADV_ARC.1.2C The security architecture description shall describe the security domains maintained by the TSF consistently with the SFRs.

ADV_ARC.1.3C The security architecture description shall describe how the TSF initialisation process is secure.

ADV_ARC.1.4C The security architecture description shall demonstrate that the TSF protects itself from tampering.

ADV_ARC.1.5C The security architecture description shall demonstrate that the

TSF prevents bypass of the SFR-enforcing functionality.

Evaluator action elements:

ADV_ARC.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence

Refinement

The Security Architecture description of the TSF initialisation process shall include the procedures to establish full functionality after power-up, state transitions from the secure state as required by FPT_FLS.1 and any state transitions of power save modes if provided by the TOE.
The Security Architecture shall describe how the security architecture design and implementation prevents bypass of SFR limiting the availability of the Test Features as required by the Limited capability and availability policy defined in FMT_LIM.2. This includes any configuration of the availability of the Test Features performed by the TOE Manufacturer before TOE Delivery.

6.3.1.6Refinements regarding Functional Specification (ADV_FSP)

Introduction

The Common Criteria assurance component of the family ADV_FSP (functional pecification) refer to the user-visible interface and behaviour of the TSF. It is an instantiation of the TOE security functional requirements. The functional specification has to show that all the TOE security functional requirements are addressed. It is a basis for the Test Coverage Analysis.

In the particular case of a Security IC specific design mechanisms, which are non-functional in nature, provide security and additionally, a test tool is delivered to the user as a part of the TOE. Therefore, refinements are provided.

The intended user of the TOE is the Developer of the Security IC Embedded Software and the Composite TOE Manufacturer.
The following text reflects the requirements of the selected component ADV_FSP.4:

Developer action elements:

ADV_FSP.4.1D The developer shall provide a functional specification.

ADV_FSP.4.2D The developer shall provide a tracing from the functional specification to the SFRs.

Content and presentation elements:

ADV_FSP.4.1C The functional specification shall completely represent the TSF.

ADV_FSP.4.2C The functional specification shall describe the purpose and method of use for all TSFI.

ADV_FSP.4.3C The functional specification shall identify and describe all parameters associated with each TSFI.

ADV_FSP.4.4C The functional specification shall describe all operations associated with each TSFI.

ADV_FSP.4.5C The functional specification shall describe all direct error messages that may result from security enforcing effects and exceptions associated with an invocation of each TSFI.

ADV_FSP.4.6C The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

Evaluator action elements:

ADV_FSP.4.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.4.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

Refinement

Although the IC Dedicated Test Software is a part of the TOE, the test functions of the IC Dedicated Test Software are not described in the Functional Specification because the IC Dedicated Test Software is considered as a test tool delivered with the TOE but not providing security functionality for the operational phase of the TOE.

The Functional Specification shall trace also security features that do not provide any external interface but that contribute to fulfil the SFRs e.g. like physical protection. Thereby they are part of the complete instantiation of the SFRs.

The Functional Specification is expected to refer to mechanisms against physical attacks in a more general way only, but detailed enough to be able to support Test Coverage Analysis also for those mechanisms where inspection of the layout is of relevance or tests beside the TSFI may be needed.

The Functional Specification shall specify operating conditions of the TOE. These conditions include but are not limited to the frequency of the clock, the power supply, and the temperature.

Background information

All functions and mechanisms which control access to the functions provided by the IC Dedicated Test Software (refer to the security functional requirement (FMT_LIM.2) will at least be referred to within the Functional Specification. Details will be given in the document for ADV_ARC”, refer to Section 6.3.1.5. In addition, all these functions and mechanisms will subsequently be refined according to all relevant requirements of the Common Criteria assurance class ADV because these functions and mechanisms are active after TOE Delivery and need to be part of the assurance aspects Tests (class ATE) and Vulnerability Assessment (class AVA). Therefore, all necessary information will be provided to allow tests and vulnerability assessment.

6.3.1.7 Refinements regarding Implementation Representation (ADV_IMP)

Introduction

The Common Criteria assurance component of the family ADV_IMP (implementation representation) refers to the implementation representation of the TSF. Since most parts of the Security IC are security enforcing it is expected that the complete implementation representation is available for the evaluators.

This requirement is supported by the application notes of CC part 3, paragraph 250, stating "The entire implementation representation is made available to ensure that analysis activities are not curtailed due to lack of information. This does not, however, imply that all of the representation is examined when the analysis activities are being performed."

The following text reflects the requirements of the selected component ADV_IMP.1:

Developer action elements:

ADV_IMP.1.1D The developer shall make available the implementation representation for the entire TSF.

ADV_IMP.1.2D The developer shall provide a mapping between the TOE design description and the sample of the implementation representation.

Content and presentation elements:

ADV_IMP.1.1C The implementation representation shall define the TSF to a level of detail such that the TSF can be generated without further design decisions.

ADV_IMP.1.2C The implementation representation shall be in the form used by the development personnel.

ADV_IMP.1.3C The mapping between the TOE design description and the sample of the implementation representation shall demonstrate their correspondence.

Evaluator action elements:

ADV_IMP.1.1E The evaluator shall confirm that, for the selected sample of the implementation representation, the information provided meets all requirements for content and presentation of evidence.

Refinement

It must be checked that the provided implementation representation is complete and sufficient to ensure that analysis activities are not curtailed due to lack of information.

6.3.1.8Refinements regarding Test Coverage (ATE_COV)

Introduction

The Common Criteria assurance component of the family ATE_COV (test coverage) “addresses the extent to which the TSF is tested, and whether or not the testing is sufficiently extensive to demonstrate that the TSF operates as specified.”

The following text reflects the requirements of the selected component ATE_COV.2:
Developer action elements:

ATE_COV.2.1D The developer shall provide an analysis of the test coverage. Content and presentation of evidence elements:

ATE_COV.2.1C The analysis of the test coverage shall demonstrate the correspondence between the tests in the test documentation and the TSFIs in the functional specification.

ATE_COV.2.2C The analysis of the test coverage shall demonstrate that all TSFIs in the functional specification have been tested.

Evaluator action elements:

ATE_COV.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

Refinement

The TOE must be tested under different operating conditions within the specified ranges. These conditions include but are not limited to the frequency of the clock, the power supply, and the temperature. This means that “Fault tolerance (FRU_FLT.2)” must be proven for the complete TSF. The tests must also cover functions which may be affected by “ageing” (such as FLASH writing).

The existence and effectiveness of mechanisms against physical attacks (as specified by the functional requirement FPT_PHP.3) cannot be tested in a straightforward way. Instead the TOE Manufacturer shall provide evidence that the TOE actually has the particular physical characteristics (especially layout design principles). This can be done by checking the layout (implementation or actual) in an appropriate way. The required evidence pertains to the existence of mechanisms against physical attacks (unless being obvious).

Background information

The IC Dedicated Test Software is seen as a “test tool” being delivered as part of the TOE. However, the Test Features do not provide security functionality. Therefore, Test Features need not to be covered by the Test Coverage Analysis but all functions and mechanisms which limit the capability of the functions (cf. FMT_LIM.1) and control access to the functions (cf. FMT_LIM.2) provided by the IC Dedicated Test Software must be part of the Test Coverage Analysis.

6.3.1.9Refinement regarding User Guidance (AGD_OPE)

Introduction

The Common Criteria assurance components of the families AGD_OPE (Operational user guidance) and AGD_PRE (Preparative user guidance) "describe all relevant aspects for the secure application of the TOE."

The Operational User Guidance documents should provide only the information which is necessary for using the TOE. Depending on the recipient of that guidance documentation Operational and Preparative User Guidance can be given in the same document.

After production the TOE is tested where communication is performed by directly contacting the pads that mostly become part of the interface during packaging. Here no guidance document according to Common Criteria class AGD is required (provided that the tests are performed by the TOE Manufacturer). Note that test procedures are described under the Common Criteria assurance component of the family ATE_FUN.

The following text reflects specific requirements of the selected component AGD_OPE.1:

Developer action elements:

AGD_OPE.1.1D The developer shall provide the operational user guidance. Content and presentation elements:

AGD_OPE.1.1C The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2C The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3C The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4C The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user- accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5C The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AGD_OPE.1.6C The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7C The operational user guidance shall be clear and reasonable.

Evaluator action elements:

AGD_OPE.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

Refinement

The TOE serves as a platform for the Security IC Embedded Software. Therefore the role of the developer of the Security IC Embedded Software is the main focus of the guidance.

If the TOE provides security functionality which can or need to be administrated (i)by the Security IC Embedded Software or (ii) if the IC Dedicated Support Software provides additional services (refer to Section 1.2.2), these aspects must be described in Guidance. This may also comprise specific functionality that must be provided by the Security IC Embedded Software to support the security of the platform and configuration options of the TOE.

Guidance documents must not contain security relevant details which are not necessary for the usage or administration of the security functionality of the TOE.

Background information

Most of the security functionality will already be effective before TOE Delivery. However, guidance to determine the behaviour of security functionality, to disable, to enable or to modify the behaviour of security functionality must be given if a configuration is possible after TOE Delivery (that means either by the Developer of the Security IC Embedded Software or by the Composite Product Manufacturer). This guidance is delivered by the TOE Manufacturer.

6.3.1.10 Refinement regarding Preparative User Guidance (AGD_PRE)

Introduction

Preparative user guidance is intended to be used by those persons responsible for secure acceptance and installation of the TOE as well as the secure preparation of the operational environment in a correct manner for maximum security.

The following text reflects specific requirements of the selected component AGD_PRE.1:

Developer action elements:

AGD_PRE.1.1D The developer shall provide the TOE including its preparative procedures.

Content and presentation elements:

AGD_PRE.1.1C The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with developer's delivery procedures.

AGD_PRE.1.2C The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

Evaluator action elements:

AGD_PRE.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2E The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

Refinement

The Family AGD_PRE addresses the activities of the delivery acceptance procedures. For the hardware platform this comprises procedures that can be applied to identify the TOE and eventually to verify the authenticity of that part of the TOE using e.g. the security functionality provided according to FAU_SAS.1.

The TOE may be configured after production before the Composite Product is delivered to the consumer. In this case, these configuration aspects have to be considered. Differences between the TOE before first use (normally done during wafer test) and Phase 7 must be summarised. Guidance to change that behaviour must exist.

The preparation may include the download of Security IC Embedded Software if parts of the Security IC Embedded Software are stored in the programmable non-volatile memory. If the TOE includes software that is delivered separately the preparation includes integration of the IC Dedicated Support Software. The preparation also includes the configuration of the TOE according to the options described in the Security Target that can be changed after TOE delivery. The guidance documentation shall describe all relevant procedures.

6.3.1.11 Refinement regarding Vulnerability Analysis (AVA_VAN)

Introduction

The Common Criteria assurance component of the family AVA_VAN (Advanced methodical vulnerability analysis) addresses "A methodical vulnerability analysis is performed by the evaluator to ascertain the presence of potential vulnerabilities.”

Since [7] does not describe a specific methodical approach available guidance for this product type shall be used for the vulnerability analysis. Especially supporting documents available as part of the Common Criteria for this product type must be considered.

The following text reflects the requirements of the selected component AVA_VAN.4:

Developer action elements:

AVA_VAN.4.1D The developer shall provide the TOE for testing.

Content and presentation elements:

AVA_VAN.4.1C The TOE shall be suitable for testing.

Evaluator action elements:

AVA_VAN.4.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.4.2E The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.4.3E The evaluator shall perform an independent, methodical vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design, security architecture description and implementation representation to identify potential vulnerabilities in the TOE.

AVA_VAN.4.4E The evaluator shall conduct penetration testing based on the identified potential vulnerabilities to determine that the TOE is resistant to attacks performed by an attacker possessing Moderate attack potential.

Refinement

The vulnerability analysis shall include a justification for the rating of information on the TOE available to the attacker and the usage of Open Samples since the protection of such information is demanded according to refinement regarding “Development Security (ALC_DVS)”, section 6.3.1.2

Application Note: Evaluator may assess the NVM content protection in addition to the vulnerability analysis related to the SFR FDP_SDC.1 in order to assess effectiveness of the security architecture if relevant security features of the TOE are identified and to support composite evaluation of the smartcard.

Application Note: The attack potential quotation as part of the vulnerability analysis shall use the Mandatory Technical Document “Application of Attack Potential to Smartcards”, which current version is [11]. It is expected that this document will be updated as attacks on smart cards are developing rapidly. Therefore the ST writer should indicate the version of this document used for the vulnerability analysis.

Application Note: The Vulnerability Analysis will assess the resistance against Side Channel Attacks to meet the SFP “Data Processing Policy” defined for the SFR “Subset information flow control (FDP_IFC.1)” and the security architecture aspect non-bypassability of the SFR “Stored data confidentiality (FDP_SDC.1)”.

Application Note: The vulnerability analysis will assess that the functions provided by the IC Dedicated Test Software cannot be abused after TOE Delivery (refer to the security functional requirements FMT_LIM.1 and FMT_LIM.2 in section 6.2). The Vulnerability Analysis shall examine that the capability and availability of Test Features is limited so that they do not allow software to be reconstructed and/or substantial information about construction of TSF to be gathered which may enable other attacks.

6.4. Security Requirements Rationale

Rationale for the security functional requirements

Table 4 below gives an overview, how the security functional requirements are combined to meet the security objectives. The detailed justification follows after the table.

Table 4 Security Requirements versus Security Objectives

Objective	TOE Security Functional and Assurance Requirements
O.Leak-Inherent	- FDP_ITT.1 “Basic internal transfer protection” - FPT_ITT.1 “Basic internal TSF data transfer protection” - FDP_IFC.1 “Subset information flow control”
O.Phys-Probing	- FDP_SDC.1 “Stored data confidentiality” - FPT_PHP.3 “Resistance to physical attack”
O.Malfunction	- FRU_FLT.2 “Limited fault tolerance - FPT_FLS.1 “Failure with preservation of secure state”
O.Phys-Manipulation	- FPT_PHP.3 “Resistance to physical attack”
O.Leak-Forced	All requirements listed for O.Leak-Inherent - FDP_ITT.1, FPT_ITT.1, FDP_IFC.1 plus those listed for O.Malfunction and O.Phys-Manipulation - FRU_FLT.2, FPT_FLS.1, FPT_PHP.3
O.Abuse-Func	- FMT_LIM.1 “Limited capabilities” - FMT_LIM.2 “Limited availability” plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1
O.Identification	- FAU_SAS.1 “Audit storage”
OE.Resp-Appl	Not applicable
OE.Process-Sec-IC	Not applicable
O.RND	- FCS_RNG.1[PTG.2] “Quality metric for random numbers” plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1

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

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

It is possible that the TOE needs additional support by the Security IC Embedded Software (e.g. timing attacks are possible if the processing time of algorithms implemented in the software depends on the content of secret).

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

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

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

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

The definition of this objective shows that it covers a situation, where malfunction of the TOE might be caused by the operating conditions of the TOE (while direct manipulation of the TOE is covered O.Phys-Manipulation). There are two possibilities in this situation: Either the operating conditions are inside the tolerated range or at least one of them is outside of this range. The second case is covered by FPT_FLS.1, because it states that a secure state is preserved in this case. The first case is covered by FRU_FLT.2 because it states that the TOE operates correctly under normal (tolerated) conditions. The functions implementing FRU_FLT.2 and FPT_FLS.1 must work independently so that their operation cannot affected by the Security IC Embedded Software (refer to the refinement). Therefore, there is no possible instance of conditions under O.Malfunction, which is not covered.

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

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

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

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

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

Other security functional requirements which prevent attackers from circumventing the functions implementing these two security functional requirements (for instance by manipulating the hardware) also support the objective. The relevant objectives are also listed in Table 4.

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

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

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

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

The justification related to the security objective "Protection during Packaging, Finishing and Personalisation (OE.Process-Sec-IC) " is as follows:

The Composite Product Manufacturer has to use adequate measures to fulfil OE.Process-Sec-IC. Depending on the security needs of the application, the Security IC Embedded Software may have to support this for instance by using appropriate authentication mechanisms for personalisation functions.

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

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

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

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

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

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

6.4.1. Dependencies of security functional requirements

Table 5 below lists the security functional requirements defined in this Security Target, their dependencies and whether they are satisfied by other security requirements defined in this Security Target. The text following the table discusses the remaining cases.

Table 5: Dependencies of the Security Functional Requirements

Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
FRU_FLT.2	FPT_FLS.1	Yes
FPT_FLS.1	None	No dependency
FMT_LIM.1	FMT_LIM.2	Yes
FMT_LIM.2	FMT_LIM.1	Yes
FAU_SAS.1	None	No dependency
FPT_PHP.3	None	No dependency
FDP_ITT.1	FDP_ACC.1 or FDP_IFC.1	Yes
FDP_IFC.1	FDP_IFF.1	See discussion below
FPT_ITT.1	None	No dependency
FDP_SDC.1	None	No dependency
FCS_RNG.1[PTG.2]	None	No dependency

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

In particular the security functional requirements providing resistance of the hardware against manipulations (e. g. FPT_PHP.3) support all other more specific security functional requirements (e. g. FCS_RNG.1[PTG.2]) because they prevent an attacker from disabling or circumventing the latter.

6.4.2. Rationale for the Assurance Requirements

The assurance level EAL4 and the augmentation with the requirements ALC_DVS.2, and AVA_VAN.4 were chosen in order to meet assurance expectations explained in the following paragraphs.

An assurance level of EAL4 is required for this type of TOE since it is intended to defend against attacks with moderate level of resistance. The TOE is dedicated to mobile phones (GSM SIM cards) so it is intended to defend against sophisticated attacks for meeting AVA_VAN.4 level.

In order to provide a meaningful level of assurance that the TOE provides an adequate level of defence against such attacks, the evaluators should have access to the low level design and source code.

6.4.3. ALC_DVS.2 Sufficiency of security measures

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

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

This assurance component is a higher hierarchical component to EAL4 (which only requires ALC_DVS.1). ALC_DVS.2 has no dependencies.

6.4.4. AVA_VAN.4 Advanced methodical vulnerability analysis

Due to the intended use of the TOE, it must be shown to be highly resistant to penetration attacks. This assurance requirement is achieved by the AVA_VAN.4 component.

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

AVA_VAN.4 has dependencies to ADV_ARC.1 “Security architecture description”, ADV_FSP.4 “Security enforcing 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 :security enforcing modules”.

All these dependencies are satisfied by EAL4.

It has to be assumed that attackers with moderate attack potential try to attack Security ICs like smart cards used for SIM cards. Therefore, specifically AVA_VAN.4 was chosen in order to assure that even these attackers cannot successfully attack the TOE.

6.4.5. Security Requirements are Internally Consistent

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

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

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

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

In a forced leakage attack the methods described in “Malfunction due to Environmental Stress” (refer to T.Malfunction) and/or “Physical Manipulation” (refer to T.Phys-Manipulation) are used to cause leakage from signals which normally do not contain significant information about secrets. Therefore, in order to avert the disclosure of primary assets identified in Section 3.1 it is important that the security functional requirements averting leakage (FDP_ITT.1, FPT_ITT.1) and those against malfunction (FRU_FLT.2 and FPT_FLS.1) and physical manipulation (FPT_PHP.3) are effective and bind well. The security features and functions against malfunction ensure correct operation of other security functions (refer to above) and help to avert forced leakage themselves in other attack scenarios. The security features and functions against physical manipulation make it harder to manipulate the other security functions (refer to above).

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

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

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

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

The security functional requirement Limited Capabilities (FMT_LIM.1) must close gaps which could be left by the control being applied to the function’s interface (Limited Availability (FMT_LIM.2)). Note that the security feature or services which limits the availability can be bypassed, deactivated or changed by physical manipulation or a malfunction caused by an attacker. Therefore, if Limited Availability (FMT_LIM.2) is vulnerable18F18, it is important to limit the capabilities of the functions in order to limit the possible benefit for an attacker.

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

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

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

7. TOE summary specification

This chapter provides general information to potential users of the TOE on how the TOE implements the Security Functional Requirements in terms of “Security Functionality”.

7.1. Malfunction

Malfunctioning relates to the security functional requirements FRU_FLT.2 and FPT_FLS.1. The TOE meets these SFRs by a group of security measures that guarantee correct operation of the TOE.

The TOE ensures its correct operation and prevents any malfunction while the security IC embedded software is executed by implementation of the following security features:

- Environmental sensors

7.2. Leakage

Leakages relates to the security functional requirements FDP_ITT.1, FDP_IFC.1 and FPT_ITT.1. The TOE meets these SFRs by implementing several measures that provide logical protection against leakage:

- Bus masking
- Random OSC clock jitter

7.3. Physical manipulation and probing

Physical manipulation and probing relates to the security functional requirements FPT_PHP.3, FDP_SDC.1. The TOE meets this SFR by implementing security measures that provides physical protection against physical probing and manipulation.

The security measures protect the TOE against manipulation of
(i) The hardware.
(ii) The application data in the NVM including the configuration data.
(iii) The special function register.

It also protects User Data or TSF data against disclosure by physical probing when stored or while being processed by the TOE.

The protection of the TOE comprises different features within the design and construction, which make reverse-engineering and tamper attacks more difficult. These features comprise of

- Active shielding
- Critical registers protection
- Memory encryption

7.4. Abuse of functionality and Identification

Abuse of functionality and Identification relates to the security functional requirements FMT_LIM.1, FMT_LIM.2 and FAU_SAS.1 by implementation of a test mode access control mechanism that prevents abuse of test functionality delivered as part of the TOE.

7.5. Random numbers

Random numbers relate to the security requirement FCS_RNG.1[PTG.2]. The TOE meets this SFR by providing a random number generator.

8. References

Ref	Title	Version	Date
[1]	AGD_OPE OG EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C	Version 0.7	August 2025
[2]	AGD_OPE SG EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C	Version 1.9	May 2025
[3]	AGD_PRE EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C	Version 1.1	August 2025
[4]	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model CCMB-2012-09-001	Version 3.1 Revision 5	April 2017
[5]	Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements CCMB-2012-09-002	Version 3.1 Revision 5	April 2017
[6]	Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements CCMB-2012-09-003	Version 3.1 Revision 5	April 2017
[7]	Common Methodology for Information Technology Security Evaluation (CEM), Evaluation Methodology CCMB-2012-09-004	Version 3.1 Revision 5	April 2017
[8]	Joint Interpretation Library The Application of CC to Integrated Circuits	Version 3.0	February 2009
[9]	Joint Interpretation Library Guidance for smartcard evaluation	Version 2.0	February 2010
[10]	Supporting Document Guidance Security Architecture requirements (ADV_ARC) for smart cards and similar devices, CCDB-2012-04-003 Joint Interpretation Library Security Architecture requirements (ADV_ARC) for smart cards, and similar devices,extend to Secure Sub-System in SoC	Version 2.1	July 2021
[11]	Joint Interpretation Library: Application of Attack Potential to Smartcards	Version 3.2.1	February 2024
[12]	Joint Interpretation Library: Minimum Site Security Requirements	Version 3.1	December2023

9. Appendix

This section records the hash values of the guidance documents of THC80F480C/384C/340C/280C/256C/228C/176C/150C:

Name	Version	Hash value(SHA-256)
AGD_OPE OG EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C [1]	0.7	840c9cf9bc5ea94b8227d01ccba228e0384ce5d981eee62d6b5068d903e9f3d7
AGD_OPE SG EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C [2]	1.9	4ea483dd910c4c499bb87d25378488b8d57c2bdf844705b627788566d5c90183
AGD_PRE EAL4+ for TMC THC80F480C/384C/340C/280C/256C/228C/176C/150C [3]	1.1	493d1ea82e654af4aeccb0ab358381618fad8a3f296a7e6473517927daef39ff