EUROSMART
European Smart Card Industry Association

Security IC Platform Protection Profile with Augmentation Packages

Version 1.0

developed by

Inside Secure
Infineon Technologies AG
NXP Semiconductors Germany GmbH
STMicroelectronics

Registered and Certified by
Bundesamt für Sicherheit in der Informationstechnik (BSI)
under the reference BSI-CC-PP-0084-2014.

This page is intentionally left blank.

Table of Contents

1 PP Introduction

1 This chapter PP Introduction contains the following sections:

PP Reference (1.1)

TOE Overview (1.2)

1.1 PP Reference

2

Title:	Security IC Platform Protection Profile with Augmentation Packages
Version number:	Version 1.0
Provided by:	Inside Secure, Infineon Technologies AG, NXP Semiconductors, and STMicroelectronics
Technical editors:	T-Systems GEI GmbH, Vorgebirgsstr. 49, 53119 Bonn, Germany in co-operation with the above mentioned IC manufacturers
Certified by:	Bundesamt für Sicherheit in der Informationstechnik (BSI) under registration number BSI-CC-PP-0084-2014

1.2 TOE Overview

1.2.1 Introduction

3 This *Security IC Platform Protection Profile* is the work of the following Integrated Circuits manufacturers:

- Infineon Technologies AG,
- Inside Secure,
- NXP Semiconductors Germany GmbH, and
- STMicroelectronics.

in co-operation with T-Systems GEI GmbH as technical editor. This protection profile has been developed on the basis of

[13] Eurosmart Smartcard IC Platform Protection Profile, Version 1.0, July 2007, BSI-PP-0035

It describes almost the same security problem definition (except the removed A.Plat-Appl), the almost same security objectives (except the removed OE.Plat-Appl) and add the security functional requirements Stored data confidentiality (FDP_SDC.1) and

Stored data integrity monitoring and action (FDP_SDI.2) compared to [13]. Therefore the current protection profile includes all security requirements of [13].

4 This document consists of

- the core protection profile defining mandatory security requirements for all TOE in the scope of this document and
- the packages of security requirements for TOE with extended security functionality:
  - Package “Authentication of the Security IC”, cf. section 7.2,
  - Package “Loader dedicated for usage in secured environment only”, cf. section 7.3.1,
  - Package “Loader dedicated for usage by authorized users only”, cf. section 7.3.2,
  - Package “TDES”, cf. section 7.4.1,
  - Package “AES”, cf. section 7.4.2,
  - Package “Hash functions”, cf. section 7.4.3.

5 The increase in the number and complexity of applications in the market of Security Integrated Circuit (hereafter “Security IC”) is reflected in the increase of the level of data security required. The security needs for a Security IC can be summarised as being able to counter those who want to defraud, gain unauthorised access to data and control a system using a Security IC. Therefore it is mandatory to:

- maintain the integrity of the content of the Security IC memories and the confidentiality of the content of protected memory areas as required by the application(s) the Security IC is built for and
- maintain the correct execution of the software residing on the Security IC.

The Security IC shall implement security features to protect the confidentiality of data in the protected memory areas and may protect data in other memory areas even if not required by SFR, e.g. in ROM. Effective user data protection requires that the Security IC especially maintains the integrity of its TSF and TSF data and their confidentiality if necessary.

6 Protected information are in general secret or integrity sensitive data as Personal Identification Numbers, Balance Value (Stored Value Cards), and Personal Data Files. Other protected information are the data representing the access rights; these include any cryptographic algorithms and keys needed for accessing and using the services provided by the system through use of the Security IC.

7 The Security IC can be used as part of products like a smart card, an USB token or other devices. The intended environment is very large; and generally once issued the Security IC can be stored and used anywhere in the world, at any time, and no control can be applied to the Security IC and its operational environment.

1.2.2 TOE Definition

8 The typical Security IC is composed of a processing unit, security components, I/O ports and volatile and non-volatile memories as depicted in Figure 1. The countermeasures against physical tampering (e.g. shields), environmental stress (e.g. sensors) and other attacks (cf. section 3.2 Threats) provided by the security IC but not directly related to other blocks are shown in a block security circuitry.

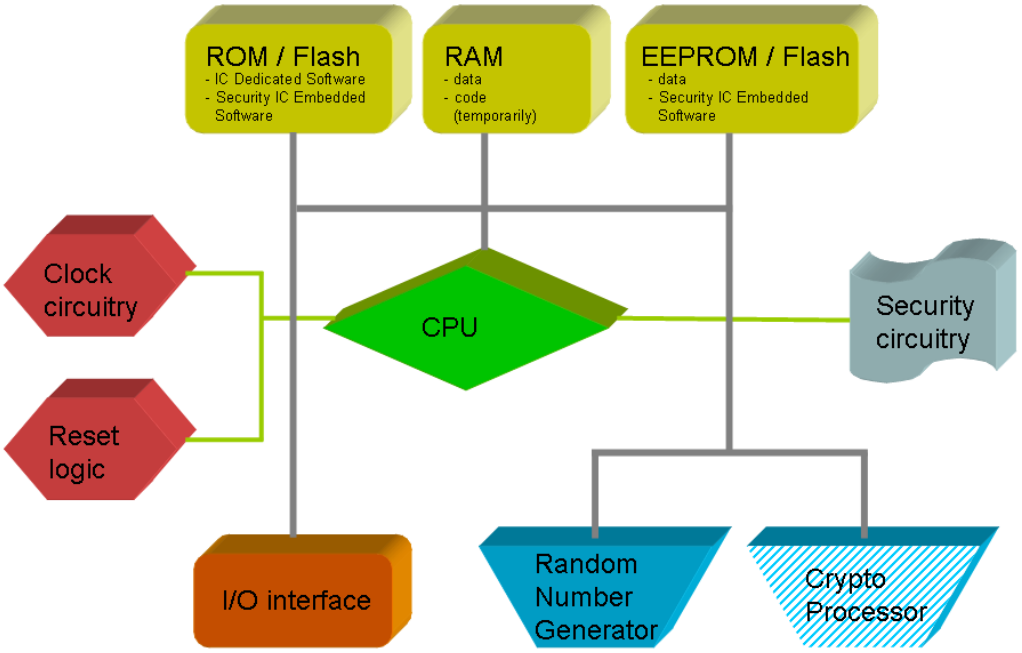


Figure 1: Typical Security IC
(Note the Cryptographic Processors are optional and the IC Dedicated Software may be stored in EEPROM / Flash as well)

9 The Target of Evaluation (TOE) is a *security integrated circuit* (security IC) which is composed of a processing unit, security components, I/O ports (contact, contactless, or similar interfaces like USB, MMC) and volatile and non-volatile memories (*hardware*). The TOE may also include IC Developer/Manufacturer proprietary *IC Dedicated Software* as long as it is delivered by the IC Manufacturer. Such software (also known as IC firmware) is often used for testing purposes during production only but may also provide additional services to facilitate usage of the hardware and/or to provide additional services (for instance in the form of a library). In addition to the IC Dedicated Software the Security IC may also comprise hardware to perform testing. All other software running on the Security IC is called Security IC Embedded Software and is not part of the TOE. Refer to Figure 1.

10 Therefore, the TOE comprises

- the circuitry of the IC (hardware including the physical memories),

- configuration data and initialisation data related to the behaviour of the security functionality[1],
- the associated guidance documentation

and, if delivered,

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

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

11 The configuration data and initialisation data related to the IC Dedicated Software and the behaviour of the security functionality are coded in non-volatile non-programmable memories (ROM), in non-volatile programmable memories (for instance EEPROM or Flash Memory), in specific circuitry or a combination thereof.

12 The “IC Dedicated Test Software” is only used to support testing of the TOE during production and does not provide security functionality to be used after TOE Delivery. Therefore, this software (or parts of it) is seen only as a “test tool” though being delivered as part of the TOE but not for usage after TOE Delivery. However, it must be verified that it cannot be abused after TOE Delivery: this is evaluated according to the Common Criteria assurance family AVA_VAN.

13 In contrast, the “IC Dedicated Support Software" does provide functions after TOE Delivery. Therefore, during evaluation it is treated as all other parts of the TOE. The IC Dedicated Support Software may be stored in the ROM, or in the non-volatile programmable memories. It may be delivered as source code or libraries in addition to the hardware.

14 The TOE is intended to be used for a Security IC product, independent of the physical interface and the way it is packaged. Note that the Security IC is usually packaged. However the way it is packaged is not specified here. Generally, a Security IC product may include other optional elements (such as specific hardware components, batteries, capacitors, antennae, ...) but these are not in the scope of this Protection Profile and can be defined in the Security Target.

15 The Composite Product comprises

- the Security IC and, if delivered and available for the Composite Product user, the IC Dedicated Support Software, i.e. the parts of the TOE of this PP,
- the Security IC Embedded Software comprising
  - Hard-coded Security IC Embedded Software (normally stored in ROM)
  - Soft-coded Security IC Embedded Software (normally stored in EEPROM or Flash Memory) and

[1] which may also be coded in specific circuitry of the IC; for a definition refer to the Glossary 7.7

- user data of the Composite TOE (especially personalisation data and other data generated and used by the Security IC Embedded Software).

16 Typically, the TOE manufacturer neither designs the Security IC Embedded software nor generates the user data of the Composite TOE. They are user data from the TOE point of view.

17 The Security IC Embedded Software can be stored in non-volatile non-programmable memories (ROM) and/or in non-volatile programmable memories (EEPROM or Flash Memory), refer to Section 7.1. The core protection profile assumes that typically

(i) the IC manufacturer installs all or the main part of the Security IC Embedded Software in ROM, EEPROM or Flash Memory during manufacturing process of the TOE and

(ii) the Composite Product Integrator installs only supplements for the Security IC Embedded Software by means of the Security IC Embedded Software itself in EEPROM or Flash Memory by (refer to the next section for details).

The installation of Security IC Embedded Software and user data of the composite product in EEPROM or Flash Memory by means of IC Dedicated Support Software

(a) until and including Personalisation (Phase 6) is addressed in the Loader Package 1 in Annex Package 1: Loader dedicated for usage in secured environment only and

(b) in Operational Usage (Phase 7) of the TOE is addressed in the Loader Package 2 in Annex 7.3.2.

18 All data managed by the Security IC Embedded Software is called user data of the Composite TOE. In addition, Pre-personalisation Data (refer to Section 7.1) may contain TSF data and user data of the TOE.

19 Further terms are explained in the Glossary (refer to Section 7.7).

1.2.3 TOE life cycle

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

- IC Development (Phase 2):
  - IC design,
  - IC Dedicated Software development,
- the IC Manufacturing (Phase 3):
  - integration and photomask fabrication,
  - IC production,
  - IC testing,

- Initialisation, and
- Pre-personalisation if necessary

The Composite Product life cycle phase 4 can be included in the evaluation of the IC as an option:

- the IC Packaging (Phase 4):
  - Security IC packaging (and testing),
  - Pre-personalisation if necessary.

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

- Security IC Embedded Software Development (Phase 1),
- the Composite Product finishing process, preparation and shipping to the personalisation line for the Composite Product (Composite Product Integration Phase 5),
- the Composite Product personalisation and testing stage where the user data of the Composite TOE is loaded into the Security IC's memory (Personalisation Phase 6),
- the Composite Product usage by its issuers and consumers (Operational Usage Phase 7) which may include loading and other management of applications in the field.

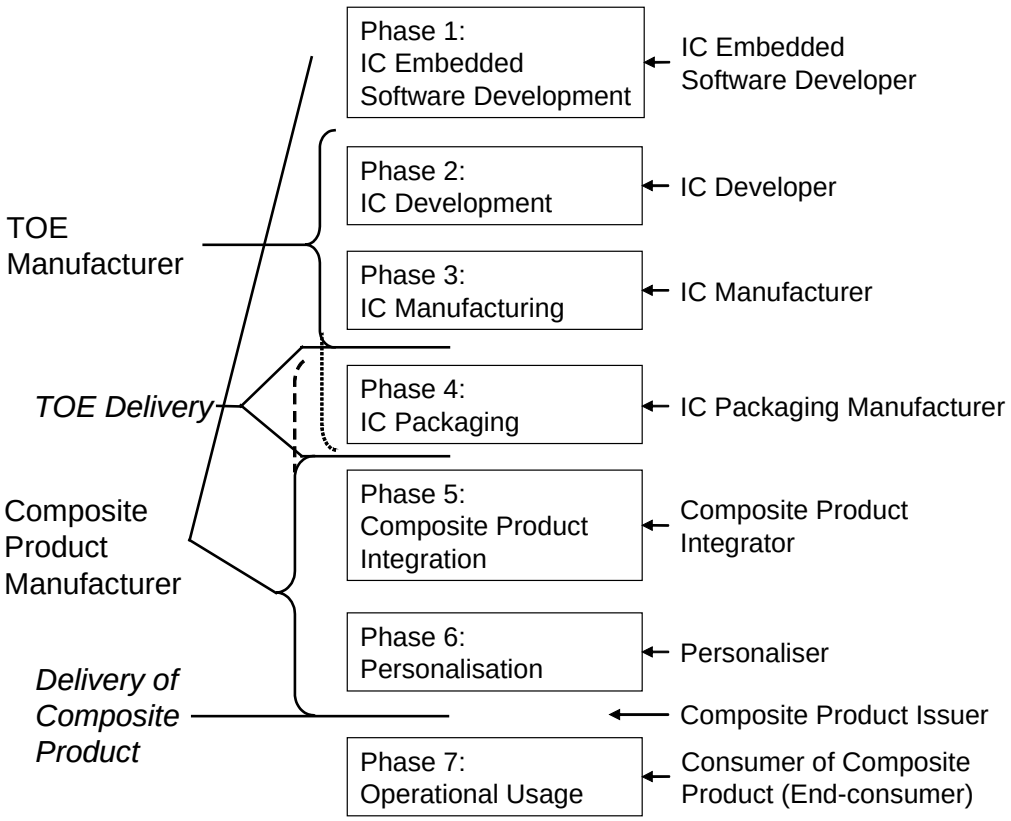


Figure 2: Definition of “TOE Delivery” and responsible Parties

22 The Security IC Embedded Software is developed outside the TOE development in Phase 1. The TOE is developed in Phase 2 and produced in Phase 3. Then the TOE can be delivered in form of wafers or sawn wafers (dice). The TOE can also be delivered in form of packaged products. In this case the corresponding assurance requirements of this Protection Profile for the development and production of the TOE not only pertain to Phase 2 and 3 but to Phase 4 in addition. Refer to the life cycle description in Section 7.1.1.

23 In the following the term “TOE Delivery” (refer to Figure 2) is uniquely used to indicate

- after Phase 3 (or before Phase 4) if the TOE is delivered in form of wafers or sawn wafers (dice) or
- after Phase 4 (or before Phase 5) if the TOE is delivered in form of packaged products.

24 The Protection Profile uniquely uses the term “TOE Manufacturer” (refer to Figure 2) which includes the following roles:

- the IC Developer (Phase 2) and
  the IC Manufacturer (Phase 3)

if the TOE is delivered after Phase 3 in form of wafers or sawn wafers (dice) or

- the IC Developer (Phase 2),
  the IC Manufacturer (Phase 3) and
  the IC Packaging Manufacturer (Phase 4)

  if the TOE is delivered after Phase 4 in form of packaged products.

25 Hence the "TOE Manufacturer" comprise all roles beginning with Phase 2 and before "TOE Delivery". Starting with "TOE Delivery" another party takes over the control of the TOE. This Protection Profile defines assurance requirements for the TOE's development and production environment up to "TOE Delivery". Refer to Figure 2.

26 The Protection Profile uniquely uses the term "Composite Product Manufacturer" which includes all roles (outside TOE development and manufacturing) except the End-consumer as user of the Composite Product (refer to Figure 2) which are the following:

- Security IC Embedded Software development (Phase 1)
- the IC Packaging Manufacturer (Phase 4)
  if the TOE is delivered after Phase 3 in form of wafers or sawn wafers (dice)
- the Composite Product Manufacturer (Phase 5) and
  the Personaliser (Phase 6).

Application Note 1: The Security Target must explicitly state whether (i) TOE Delivery is after Phase 3 only or (ii) after Phase 4 as well. This can be done by using the relevant information from the paragraphs above. A detailed description of the life-cycle is given in Section 7.1.

Application Note 2: If the TOE provides functionality to be used after TOE Delivery this is part of the IC Dedicated Support Software. Then such functions must be specified in the Security Target of the actual TOE. Revise the above paragraphs in the Security Target to make clear if the TOE comprises IC Dedicated Support Software (e.g. a loader for the Flash Memory).

1.2.4 Life-Cycle versus Scope and Organisation of this Protection Profile

27 The whole life-cycle of the Composite Product will be considered during evaluations using this Protection Profile as far as the TOE Manufacturer is directly involved. Complex details are given in terms of refinements of the Common Criteria assurance components since they are built to cover the development and production processes of the TOE.

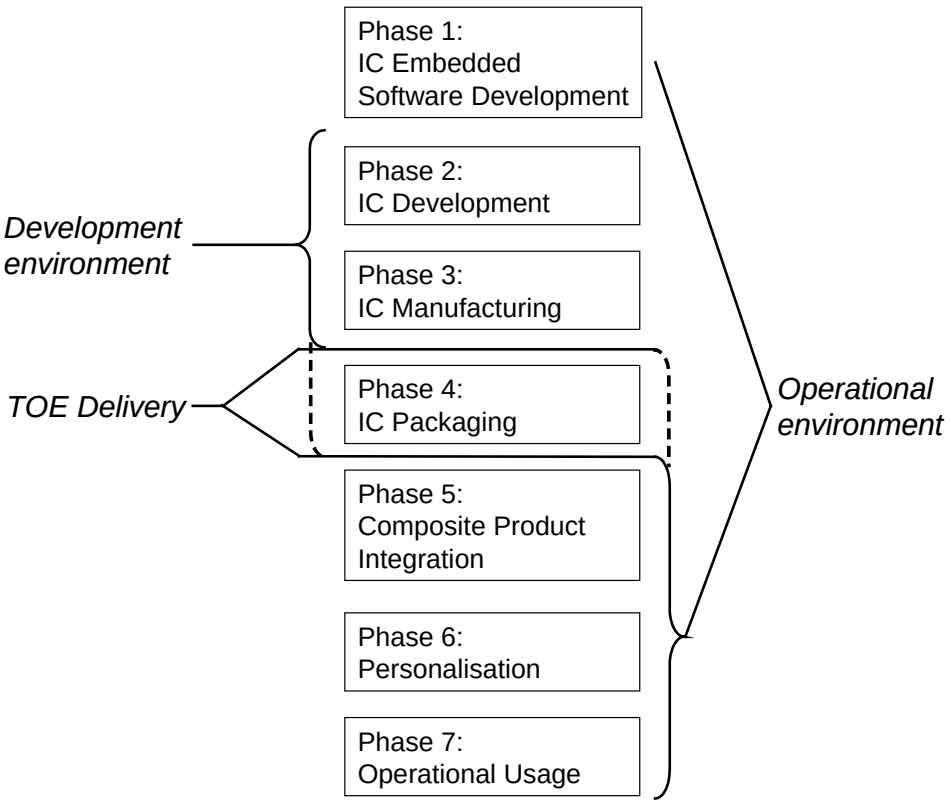


Figure 3: Development environment and Operational environment

28 The scope of those assurance components referring to the TOE's life-cycle is limited to Phases 2 and 3. These phases are under the control of the TOE Manufacturer.

29 The IC Packaging and Testing in Phase 4 may be included in the scope if the TOE Manufacturer delivers packaged TOE, refer to the dashed line in figure 4 below.

30 All procedures within these phases are addressed by the Protection Profile. This includes the interfaces to the other phases where information and material is being exchanged. The IC designer, the developer of the IC Dedicated Software, the mask manufacturer and the IC manufacturer are summarised under the term TOE Manufacturer. The Composite Product Manufacturer and the TOE Manufacturer interact and may exchange critical information. Therefore, Common Criteria assurance requirements will be refined in Section 6.2.1 to ensure that this Protection Profile exactly reflects the requirements for the exchange of information and material between the TOE Manufacturer and the Composite Product Manufacturer.

31 In particular, the Common Criteria assurance requirements ALC_DEL (delivery) and AGD_PRE are refined. So, the details regarding development of the Security IC Embedded Software, secure delivery and receipt of TOE are addressed.

32 It may be necessary to state security objectives for other parties in the ST if they use security critical information of the TOE manufacturer. However, it cannot be assessed during an evaluation of the TOE whether these security objectives for the TOE

environment are met. Consequently, these requirements must be taken into account during the evaluation of the Security IC Embedded Software or Composite Product, respectively.

33 For assumptions regarding the usage of the TOE (its environment) made in this Protection Profile refer to Section 3.4.

Application Note 3: The TOE may provide functions supporting the Security IC's life-cycle (for instance secure/authentic delivery). In this case the corresponding requirements will be specified in the Security Target in terms of security objectives and functional requirements. This is visualised in Figure 4.

34 This approach of Security IC Life-Cycle versus PP Requirements is visualised in Figure 4. Additional requirements may be chosen to correctly interface to a Protection Profile for the Security IC Embedded Software.

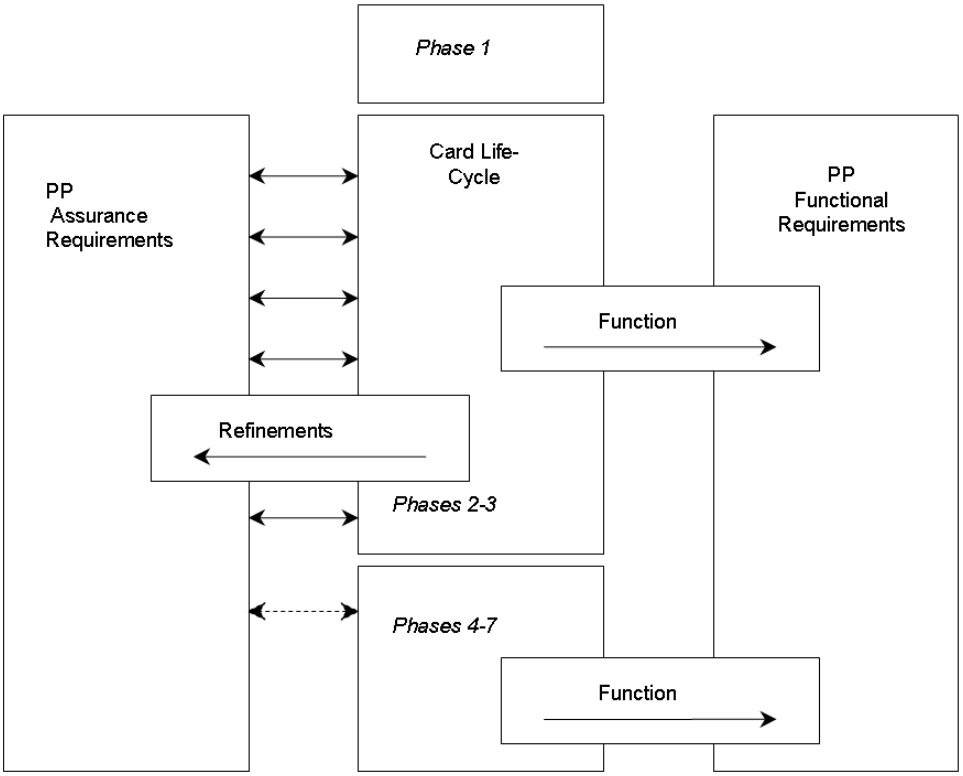


Figure 4: Security IC Life-Cycle versus PP Requirements

1.2.5 Specific Issues of Security IC Hardware and the Common Criteria

35 The Security IC is a platform to be used by the Security IC Embedded Software. The Security IC itself provides security services for the Security IC Embedded Software. All user data of the Composite TOE are assets of the Security IC Embedded Software. However, the hardware platform must

- maintain the integrity and the confidentiality of the content of the Security IC memories as required by the context of the Security IC Embedded Software and
- maintain the correct execution of the Security IC Embedded Software.

This requires that the Security IC especially maintains the integrity of the TSF and the confidentiality of at least the critical TSF parts.

36 The Security IC Embedded Software shall keep the security functionality and features of the Security IC i.e.

- store the confidential data in protected memory areas only and
- use the functionality and the security services of the Security IC in a secure way.

37 The TOE security mechanisms (cf. [6], sec. 2.11.4) need to work together in different combinations to counter attacks. Owing to complex dependencies, these combinations are only apparent in the context of a specific attack scenario. Often the composition of a security feature (cf. [1], sec. B.4.2, [6], sec. 2.11.4) only becomes clear when considering a specific attack path during vulnerability analysis (AVA_VAN). A security mechanism may be needed in different security features depending on the attack path. This has to be considered during the TOE evaluation.

38 Detailed specification of the (implementation dependent) security mechanisms and the associated integration of these security mechanisms are beyond the scope of this Protection Profile.

39 This Protection Profile will define the security problems related to Security ICs (and the corresponding security objectives and requirements) in a more general way though addressing all important issues. Attack scenarios will be mentioned whenever appropriate but only to illustrate the corresponding security problem. The information about attack scenarios cannot be considered as being complete.

40 It is not possible (because of differences between the security ICs) nor desirable (confidentiality; do not instruct the attackers) to specify all the specific attack scenarios and all the security features in this Protection Profile. The Security Target may describe the Security IC in more detail without necessarily disclosing construction details.

41 This Protection Profile will highlight some specific security features or functions though breaking them would not necessarily affect the primary assets in a direct way.

42 Hardware and software together shall build an integrated secure whole. There can be a lot of interdependencies between the two. Requirements for the Security IC Embedded Software should normally be described as Security Objectives for the operational Environment (Section 4.3) if they are necessary to ensure secure operation of the TOE (here: the Security IC).

43 However, particular requirements for the software are often not clear before considering a specific attack scenario during vulnerability analysis (AVA_VAN). Therefore, such results from the evaluation of the Security IC must be given to the developer of the Security IC Embedded Software in the guidance referenced in the certification report and be taken into account during the evaluation of the software.

44 In consequence, the Security Objectives for the operational Environment (Section 4.3) cannot be expected to exactly specify all requests for the Security IC Embedded Software. The guidance document must give all the TOE specific information

supporting the Embedded Software developer to use the Security IC in a secure way. In this way modularity for evaluations is supported without making vulnerabilities of the Security IC public or giving details about the implementation.

45 The evaluation of the Security IC according to this Protection Profile is independent of the evaluation of the composite product which includes the Security IC and the Security IC Embedded Software. The developer of the Security IC Embedded Software decides if the platform (evaluated Security IC) is suitable for the composite product. The composite evaluation process of the composite product may focus on the evaluation of the Security IC Embedded Software running on the security IC and reuse results of the product evaluation of the Security IC with the Dedicated Software (for details of the composite evaluation refer to the Supporting Documents [10]).

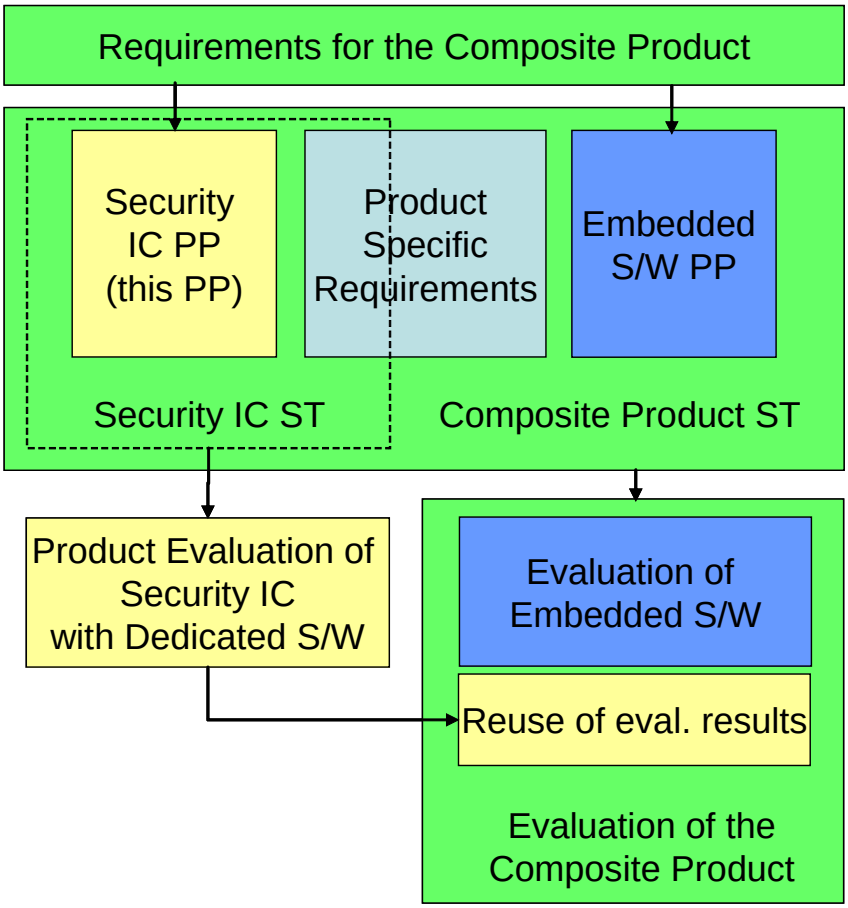


Figure 5: Relationship between the evaluations of Security IC with Dedicated Software and Composite Product including Embedded Software

46 The TOE Manufacturer delivers the TOE to the Composite Product Manufacturer. The interfaces available after delivery by TOE Manufacturer are different from the interfaces for the operational usage of the composite product by End-consumer (Phase 7). This interface for the End-consumer is determined by the developer of the Security IC Embedded Software. Therefore, the guidance documentation delivered by the TOE Manufacturer is intended for the developer of the Security IC Embedded

Software and the Composite Product Manufacturer. The term "End-consumer" is used in this Protection Profile for the user of the Composite Product in Phase 7.

47 In addition, for a Security IC the knowledge of the design information shall be considered in the vulnerability analysis and critical design information shall be protected (as reflected by the Common Criteria assurance component of the family ALC_DVS). As an example cryptographic attacks are not only possible taking a purely theoretical (mathematical) approach but also by recording and interpreting information related to the execution of cryptographic operations. Details about the implementation may make such attacks easier. If details of the design and layout of the Security IC are freely available this would considerably reduce the effort to mount an attack, since reverse-engineering would not be required. Therefore, in the case of a Security IC, maintaining the confidentiality of the design is very important. This is in contrast to Kerckhoff's principle, where the security of a cryptographic algorithm should rely solely on the secrecy of the keys and not on the secrecy of the algorithm itself.

2 Conformance Claims

48 This chapter contains the following sections:

CC Conformance Claim (2.1)

PP Claim (2.2)

Package Claim (2.3)

PP Application Notes (2.4)

2.1 CC Conformance Claim

49 This Protection Profile claims to be conformant to the Common Criteria version 3.1.

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

51 This Security IC Platform Protection Profile has been built with the Common Criteria for Information Technology Security Evaluation; Version 3.1

which comprises

[1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; September 2012, Version 3.1, Revision 4, CCMB-2012-09-001,

[2] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; September 2012, Version 3.1, Revision 4, CCMB-2012-09-002

[3] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; September 2012, Version 3.1, Revision 4, CCMB-2012-09-003

52 The

[4] Common Methodology for Information Technology Security Evaluation (CEM), Evaluation Methodology; September 2012, Version 3.1, Revision 4, CCMB-2012-09-004

has been taken into account.

2.2 PP Claim

53 The Protection Profile requires strict conformance of the ST or PP claiming conformance to this PP.

54 This PP does not claim conformance to any other PP.

2.3 Package Claim

55 The minimum assurance level for this Protection Profile is EAL4 augmented with AVA_VAN.5 and ALC_DVS.2 (refer to Section 6.2 for more detail).

2.4 PP Application Notes

56 In this Protection Profile operations are completed for all security functional components except the component FCS_RNG.1 (Generation of random numbers), FAU_SAS.1 (Audit storage), FDP_SDC.1 (Memory protection) and FDP_SDI.2 (Stored data integrity monitoring and action). To complete the latter is left to the ST writer.

57 This Protection Profile contains other application notes distributed through the paper. The application notes are separated paragraphs which are marked with “Application Note” followed by a number.

3 Security Problem Definition

58 This chapter contains the following sections:

Description of Assets (3.1)

Threats (3.2)

Organisational Security Policies (3.3)

Assumptions (3.4)

3.1 Description of Assets

Assets regarding the Threats

59 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.

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

SC1 integrity of user data of the Composite TOE,

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

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

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

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

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

63 This Protection Profile requires the TOE to provide at least one security service: the generation of random numbers by means of a physical Random Number Generator. The annex 7 provides packages for typical additional security services. The Security Target may require additional security services as described in these packages or define TOE specific security services. It is essential that the TOE ensures the correct operation of all security services provided by the TOE for the Security IC Embedded Software.

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

SC4 deficiency of random numbers.

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

- logical design data, physical design data, IC Dedicated Software, and configuration data,

- Initialisation Data and Pre-personalisation Data, specific development aids, test and characterisation related data, material for software development support, and photomasks.

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

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

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

69 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. Explanations can be found in Section 7.1.2.

3.2 Threats

70 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[2] 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.

71 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.

72 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.

73 The Security IC is a platform for the Security IC Embedded Software which ensures that especially the critical user data of the Composite TOE are stored and processed in a secure way (refer to below). The Security IC Embedded Software must also ensure that critical user data of the Composite TOE are treated as required in the application context (refer to Section 3.4). In addition, the personalisation process supported by the Security IC Embedded Software (and perhaps by the Security IC in addition) must be secure (refer to Section 3.4). This last step is beyond the scope of this Protection Profile. As a result the threat “cloning of the 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 this Protection Profile (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.

74 The high-level security concerns are refined below by defining threats as required by the Common Criteria (refer to Figure 6). Note that manipulation of the TOE is only a means to threaten user data and is not a success for the attacker in itself.

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

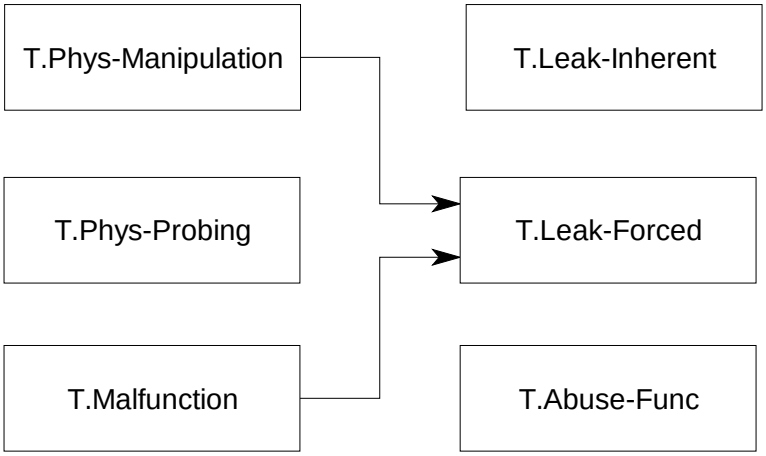


Figure 6: Standard Threats

75 The high-level security concern related to security service is refined below by defining threats as required by the Common Criteria (refer to Figure 7).

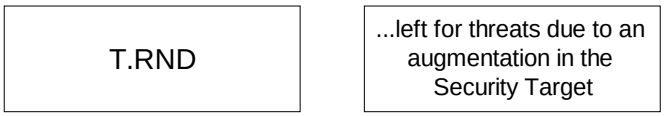


Figure 7: Threats related to security service

Application Note 4: If the TOE provides further security functions or security services to the Security IC Embedded Software (such as described in the packages) this would result in having additional security services to be protected in the Security Target. If these services are implemented due to security objectives for the TOE covering additional threats in the operational environment the ST author should add the appropriate text to the above paragraph.

76 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. For detail refer to the assumptions regarding the Security IC Embedded Software specified in Section 3.4.

77 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.

78 The TOE's countermeasures are designed to avert the threats described below. Nevertheless, they may be effective in earlier phases (Phases 4 to 6, refer to Figure 2 on page 11).

79 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 8. Due to the intended usage of the TOE all interactions are considered as possible.

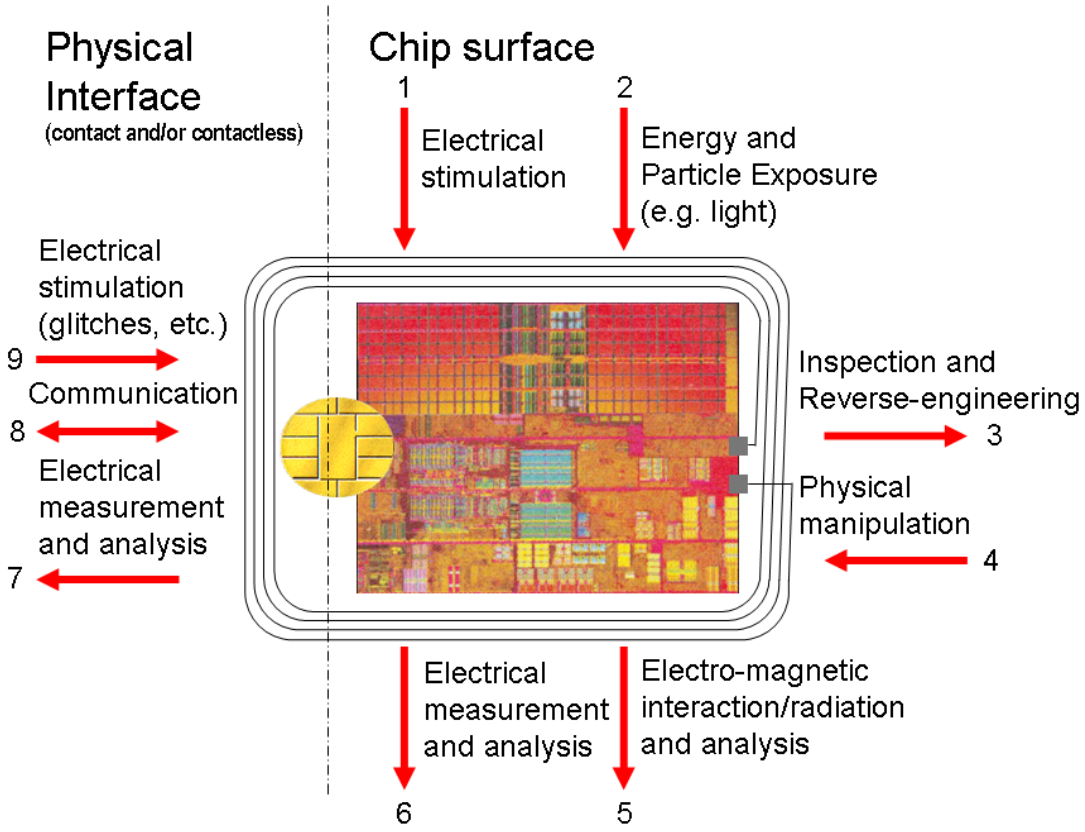


Figure 8: Interactions between the TOE and its outer world

80 An interaction with the TOE can be done through the physical interfaces (Number 7 – 9 in Figure 8) which are realised using contacts and/or a contactless interface. Influences or interactions with the TOE also occurs through the chip surface (Number 1 – 6 in Figure 8). 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.

81 Examples for specific attacks are given in Section 7.2.

Standard Threats

82 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 8) or measurement of emanations (Number 5 in Figure 8) and can then be related to the specific operation being performed.

83 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 8). Techniques commonly employed in IC failure analysis and IC reverse engineering efforts may be used. Before that hardware security mechanisms and layout characteristics need to be identified (Number 3 in Figure 8). 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.

84 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 8).

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.

85 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 failure analysis (Numbers 1, 2 and 4 in Figure 8) and IC reverse engineering efforts (Number 3 in Figure 8). 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 pre-requisite. 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 8).

86 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 8) which normally do not contain significant information about secrets.

87 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

88 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

89 The following Figure 9 shows the policies applied in this Protection Profile.

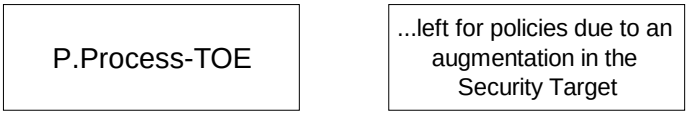


Figure 9: Policies

Application Note 5: The TOE may provide further security functions or security services which can be used by the Security IC Embedded Software. Particular specific security functionality may not necessarily be derived from threats or policies identified in the TOE's environment because it can only be decided in the context of the Security IC application. The specific security objectives and their security functionality can be provided according to security policies may be added in the Security Target if this Protection Profile needs to be augmented, cf. to annex for examples.

90 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.

91 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

92 The following Figure 10 shows the assumptions applied in this Protection Profile.

Figure 10: Assumptions

Application Note 6: The TOE may provide specific security services which can be used by the Security IC Embedded Software. In this case it can be required to add additional assumptions in the Security Target (cf. CEM [4] para 361).

93 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 clock) and (at least) mediates the

communication with the Security IC Embedded Software.

94 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.

95 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 (refer to Sections 1.2.2 and 7.1) are assumed to be protected appropriately. For a preliminary list of assets to be protected refer to paragraph 96 (page 29).

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

- the Security IC Embedded Software including specifications, implementation and related documentation,
- Pre-personalisation Data and Personalisation Data including specifications of formats and memory areas, test related data,
- the user data of the Composite TOE and related documentation, and
- material for software development support

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

97 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.

98 Note that particular requirements for the Security IC Embedded Software are often not

clear before considering a specific attack scenario during vulnerability analysis of the Security IC (AVA_VAN). A summary of such results is provided in the document "ETR for composite evaluation" (ETR-COMP), cf. [11]. This document will be provided for the evaluation of the composite product, cf. [10]. The ETR-COMP may also include guidance for additional tests being required for the combination of hardware and software. The TOE evaluation must be completed before evaluation of the Security IC Embedded Software can be completed. The TOE evaluation can be conducted before and independently from the evaluation of the Security IC Embedded Software.

99 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 Protection Profile is conducted on generalized application context. The concrete requirements for the Security IC Embedded Software shall be defined in the Protection Profile respective Security Target for the Security IC Embedded Software. The Security IC cannot prevent any compromise or modification of user data of the Composite TOE by malicious Security IC Embedded Software.

Application Note 7: Further assumptions might be required if the TOE provides specific additional security services for the Security IC Embedded Software (cf. [4], para. 178 and 361 for rules to obey if assumptions are added).

4 Security Objectives

100 This chapter Security Objectives contains the following sections:

Security Objectives for the TOE (4.1)

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

Security Objectives for the operational Environment *(4.3)*

Security Objectives Rationale (4.4)

4.1 Security Objectives for the TOE

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

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

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

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

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

102 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 11). Note that the integrity of the TOE is a means to reach these objectives.

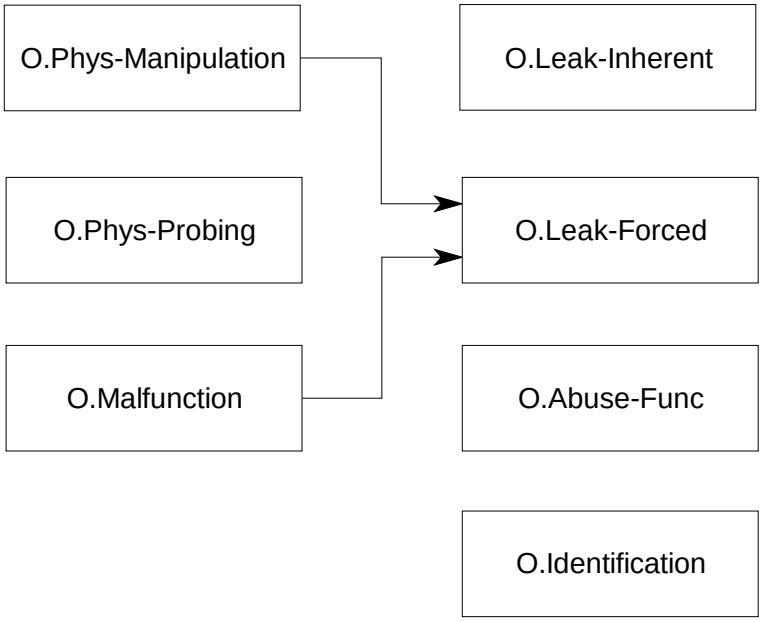


Figure 11: Standard Security Objectives

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

SG4 provide true random numbers.

104 The additional high-level security considerations are refined below by defining security objectives as required by the Common Criteria (refer to Figure 12).

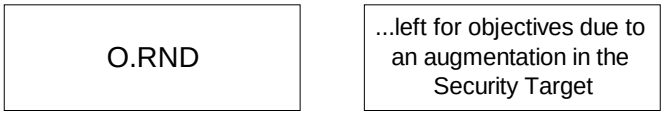


Figure 12: Security Objectives related to Specific Functionality

Application Note 8: If the TOE provides further functions or services to the Security IC Embedded Software (such as loader or cryptographic functions, cf. to the annexes) this may result in having additional high-level security goals in the Security Target which must also be refined.

Standard Security Objectives

105 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).

106 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.

107 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.

108 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. Examples of 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.

109 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.

110 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.

111 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 an information channel for a leakage attack.

112 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.

113 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 SG4)

114 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.

Application Note 9: If the TOE provides further services to the Security IC Embedded Software (such as cryptographic functions) this may result in having additional security objectives in the Security Target. The annex 7 provides packages for additional security services the TOE may provide.

4.2 Security Objectives for the Security IC Embedded Software

115 The development of the Security IC Embedded Software is outside the development and manufacturing of the TOE (cf. section 1.2.3). The Security IC Embedded Software defines the operational use of the TOE. This section describes the security objective for the Security IC Embedded Software.

116 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.

117 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

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

OE.Process-Sec-IC	Protection during composite product manufacturing Security procedures shall be used after TOE Delivery up to delivery to the end-consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use). This means that Phases after TOE Delivery up to the end of Phase 6 (refer to Section 1.2.3) must be protected appropriately. For a preliminary list of assets to be protected refer to paragraph 96 (page 29).

4.4 Security Objectives Rationale

119 Table 1 below gives an overview, how the assumptions, threats, and organisational security policies are addressed by the objectives. The text following after the table justifies this in detail.

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

Table 1: Security Objectives versus Assumptions, Threats or Policies

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

121 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.

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

123 O.Identification requires that the TOE has to support the possibility of a unique identification. The unique identification can be stored on the TOE. Since the unique identification is generated by the production environment the production environment must support the integrity of the generated unique identification. The technical and organisational security measures that ensure the security of the development environment and production environment are evaluated based on the assurance measures that are part of the evaluation. For a list of material produced and processed by the TOE Manufacturer refer to paragraph 69 (page 21). 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.

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

125 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.

126 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:

127 For all threats the corresponding objectives (refer to Table 1) are stated in a way, which directly corresponds to the description of the threat (refer to Section 3.2). It is clear from the description of each objective (refer to Section 4.1), that the corresponding threat is removed if the objective is valid. More specifically, in every case the ability to use the attack method successfully is countered, if the objective holds.

5 Extended Components Definition

5.1 Definition of the Family FCS_RNG

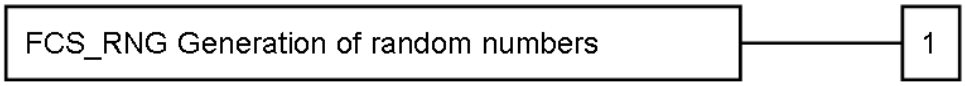
128 To define the IT security functional requirements of the TOE an additional family (FCS_RNG) of the Class FCS (cryptographic support) is defined here. This family describes the functional requirements for random number generation used for cryptographic purposes.

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*].

Application Note 10: A physical random number generator (RNG) produces the random number by a noise source based on physical random processes. A non-physical true RNG uses a noise source based on non-physical random processes like human interaction (key strokes, mouse movement). A deterministic RNG uses an random seed to produce a pseudorandom output. A hybrid RNG combines the principles of physical and deterministic RNGs where a hybrid physical RNG produces at least the amount of entropy the RNG output may contain and the internal state of a hybrid deterministic RNG output contains fresh entropy but less than the output of RNG may contain.

5.2 Definition of the Family FMT_LIM

129 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 (refer to Section 6.1) appropriate to address the specific issues of preventing the abuse of functions by limiting the capabilities of the functions and by limiting their availability.

130 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 (refer to Limited capabilities (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.

131 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*].

132 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 11: 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 user environment 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

133 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.

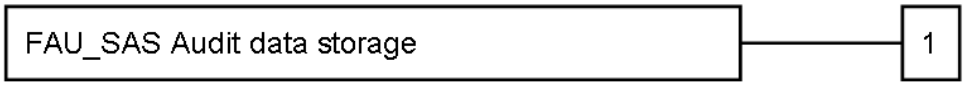
134 The family “Audit data storage (FAU_SAS)” is specified as follows.

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

135 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.

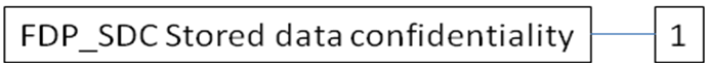
136 The family “Stored data confidentiality (FDP_SDC)” is specified as follows.

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. It complements the family Stored data integrity (FDP_SDI) which protects the user data from integrity errors while being stored in the memory.

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

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

Security Functional Requirements for the TOE (6.1)

Security Assurance Requirements for the TOE (6.2)

Refinements of the TOE Assurance Requirements (6.2.1)

Security Requirements Rationale (6.3)

Rationale for the security functional requirements (6.3.1)

Dependencies of security functional requirements (6.3.2)

Rationale for the Assurance Requirements (6.3.3)

Security Requirements are Internally Consistent (6.3.4)

138 Note that Section 6.2.1 is not mandatory according to the Common Criteria. The Refinements of the TOE Assurance Requirements take into account the peculiarities of the Security IC development and production process (Security IC's life-cycle).

139 The standard Security Requirements are shown in Figure 13. These security components are listed and explained below.

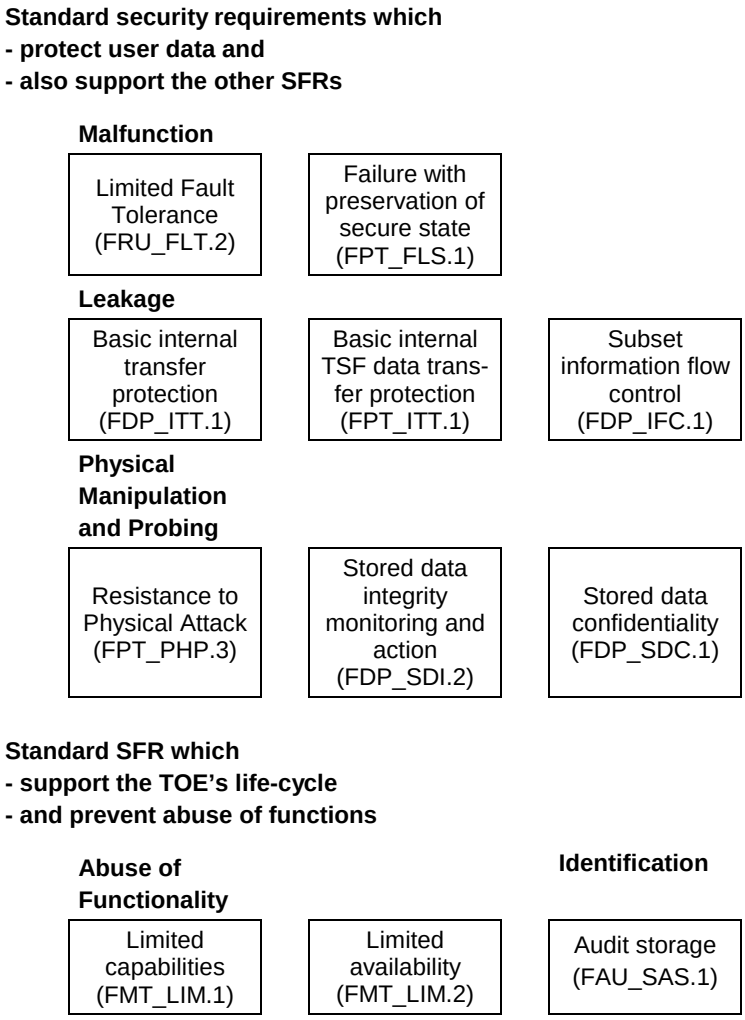


Figure 13: Standard Security Requirements

140 The Security Functional Requirements related to Specific Functionality are shown in Figure 14. These security functional components are listed and explained below.

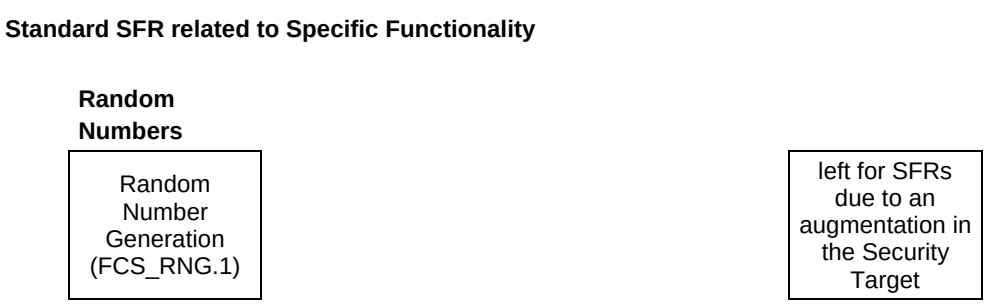


Figure 14: Security Functional Requirements related to Specific Functionality

Application Note 12: If the TOE provides further functions or services to the Security IC Embedded Software (e.g. such as described in the packages) this would result in having additional Security Functional Requirements in the Security Target.

6.1 Security Functional Requirements for the TOE

141 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.

142 The refinement operation is used to add detail to a requirement, and, thus, further restricts a requirement. Refinements of security requirements are denoted in such a way that added words are in bold text and removed words are crossed out. In some cases a interpretation refinement is given. In such a case a extra paragraph starting with “Refinement” may be given.

143 The selection operation is used to select one or more options provided by the CC in stating a requirement. Selections having been made by the PP author are denoted as underlined text. Selections to be filled in by the ST author appear in square brackets with an indication that a selection is to be made [selection:] and are italicised.

144 The assignment operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments having been made by the PP author are denoted by showing as underlined text. Assignments to be filled in by the ST author appear in square brackets with an indication that an assignment is to be made [assignment:] and are italicised. In some cases the assignment made by the PP authors defines a selection to be performed by the ST author. Thus this text is underlined and italicised like this.

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

Malfunctions

146 There are different ranges of operating conditions such as supply voltage, external frequency and temperature. The TOE can be operated within the limits visualised as the inner dashed rounded rectangle in Figure 15 and must operate correctly there. The limits have been reduced to ensure correct operation. This is visualised by the outer dotted rounded rectangle in the figure.

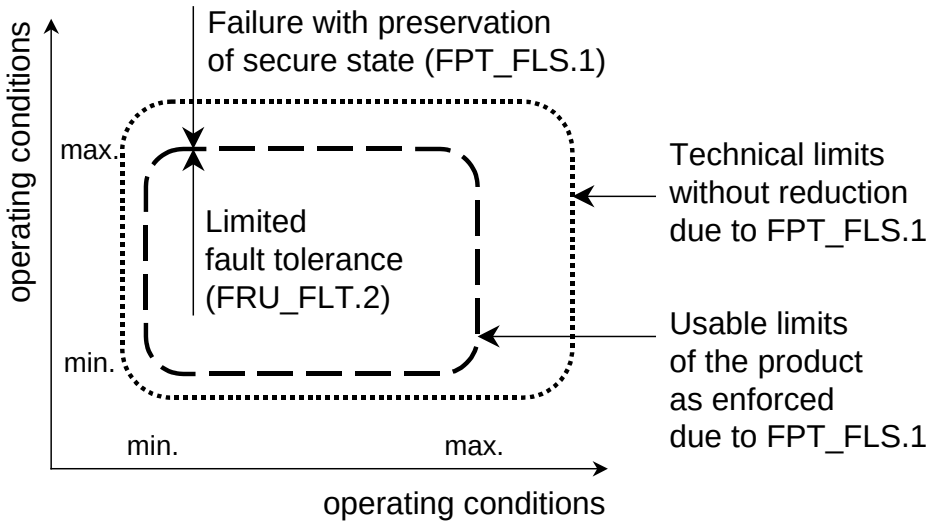


Figure 15: Paradigm regarding Operating Conditions

147 Figure 15 must not be understood as being two-dimensional and defining static limits only. Reality is multi-dimensional and includes a variety of timing aspects. Note that the limit of the operating conditions visualised by the inner dashed rounded rectangle in Figure 15 is not necessarily exactly reflected by the limits identified in the TOE's data sheet. Instead this limit marks the boundary between the "tolerance reaction" of the TOE and the "active reaction" of sensors (and perhaps other circuitry).

148 The security functional component has been selected in order to address the robustness within some limit (as shown by the inner dashed rectangle in Figure 15) before active reaction takes place to reach a failure with preservation of secure state. Note that the TOE does not (in most cases) actually detect faults or failures and then correct them in order to guarantee further operation of all the TOE's capabilities. This is the way software would implement Limited fault tolerance (FRU_FLT.2). Instead the TOE will achieve exactly the same by (i) stable functional design within the limits of operational conditions (e.g. temperature) and (ii) eliminating the cause for possible faults and by being resistant against influences (e.g. robustness against glitches of the power supply by means of filtering). In the case of the TOE the "reaction to a failure" is replaced by the "reaction to operating conditions" which could cause a malfunction without the reaction of the TOE's countermeasure addressed by the security functional component FPT_FLS.1.

149 If the TOE is exposed to other operating conditions this may not be tolerated. Then the TOE must detect that and preserve a secure state (use of detectors and cause a reset for instance). The security functional component (FPT_FLS.1) has been selected to ensure that. The way the secure state is reached depends on the implementation. Note that the TOE can monitor both external operating conditions and other internal conditions and then react appropriately. Exposure to specific "out of range" external operating conditions (environmental stress) may actually cause failure conditions internally which cannot be tolerated by FRU_FLT.2. Referring to external operating conditions the TOE is expected to respond if conditions are detected which may cause a failure. Examples for implementations of the security functional requirement FPT_FLS.1 are a voltage detector (external condition) and a circuitry which detects

accesses to address areas which are not used (internal condition).

150 Those parts of the TOE which support the security functional requirements “Failure with preservation of secure state (FPT_FLS.1)” and “Limited Fault tolerance (FRU_FLT.2)” shall be protected from misconfiguration of and by-passing by means of the Security IC Embedded Software. These aspects are addressed by the security assurance requirements Architectural design (ADV_ARC.1).

151 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)[3].
Refinement:	**The term “failure” above means “circumstances”. The TOE prevents failures for the “circumstances” defined above.**

Application Note 13: Environmental conditions include but are not limited to power supply, clock, and other external signals (e.g. reset signal) necessary for the TOE operation.

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

FPT_FLS.1	**Failure with preservation of secure state**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_FLS.1.1	The TSF shall preserve a secure state when the following types of failures occur: exposure to operating conditions which may not be tolerated according to the requirement Limited fault tolerance (FRU_FLT.2) and where therefore a malfunction could occur[4].
Refinement:	**The term “failure” above also covers “circumstances”. The TOE prevents failures for the “circumstances” defined above.**

[3] [assignment: *list of types of failures*]

[4] [assignment: *list of types of failures in the TSF*]

Application Note 14: The Security Target shall describe the secure state. In addition the author of the Security Target should give some rationale together with a clear definition of the secure state here.

Application Note 15: The Common Criteria suggest that the TOE generates audit data for the security functional requirements Limited fault tolerance (FRU_FLT.2) and Failure with preservation of secure state (FPT_FLS.1). This may be advantageous or even required for the application context. The author of the Security Target should consider this especially for FPT_FLS.1.

Abuse of Functionality

153 During testing at the end of Phase 3 before TOE Delivery, the TOE shall be able to store some data (for instance about the production history or identification data of the individual die or other data to be used after delivery). Therefore, the security functional component Audit storage (FAU_SAS.1) has been added. The security functional component FAU_SAS.1 has been newly created (refer to Section 5.3) and is used instead of FAU_GEN.1 which is too comprehensive to be applicable in this context.

154 The requirement FAU_SAS.1 shall be regarded as covering the injection of Initialisation Data, Pre-personalisation Data or other data as described in Section 7.1.1. After TOE Delivery the identification data (injected as part of the Initialisation Data) and the Pre-personalisation Data are available to the Security IC Embedded Software. These data are protected by the TOE as all other user data of the Composite TOE. It's up to the Security IC Embedded Software to use these data stored and provided by the TOE.

155 Each instantiation of the TOE has to undergo exhaustive testing at clearly defined stages of the production process where the correct functioning and properties are ascertained and also if necessary information might be stored in the EEPROM/Flash. This task is done by a specialised group of people of the TOE manufacturer called "test-personnel". The test-personnel is the first user of the TOE and their identity may be assumed as default user for FAU_SAS.1. If the Initialisation Data, Pre-personalisation Data or assigned other data can be written only once the test-personnel will be the only user able to store these data.

156 The TOE shall prevent functions (provided by the IC Dedicated Test Software or by hardware features) from being abused after TOE Delivery in order to compromise the TOE's security. (All such functions are called "Test Features" below.) This includes but is not limited to: disclose or manipulate user data of the Composite TOE and bypass, deactivate, change or explore security features or functions of the TOE. Details depend on the capabilities of the Test Features provided by the IC Dedicated Test Software and/or the hardware.

157 This can be achieved (i) by limiting the capabilities of these Test Features after Phase 3, (ii) by limiting the availability of these Test Features after Phase 3 or (iii) by a combination of both. The extended security functional components Limited capabilities (FMT_LIM.1) and Limited availability (FMT_LIM.2) have been defined (refer to Section 5.2) to address this. The Limited capability policy defined for SFR FMT_LIM.1

and the Limited availability policy defined for FMT_LIM.2 are identical.

158 Examples of the technical mechanisms used in the TOE are user authentication ("passwords"), non-availability (for instance through removal or disabling by "fusing") or a combination of both. A detailed technical specification would unnecessarily disclose details and is beyond the scope of a specification of requirements.

159 The TOE is tested after production in Phase 3 (refer to Section 7.1.1) using means provided by the IC Dedicated Software and/or specific hardware. The IC Dedicated Software is considered as being a test tool delivered as part of the TOE and used before TOE Delivery only. It does not provide functions in later phases of the Security IC's life-cycle. Therefore, no security functional requirement is mandatory according to this Protection Profile regarding these testing capabilities except FPT_LIM.1 and FPT_LIM.2.

160 All necessary information about the capabilities of the Test Features (including the IC Dedicated Software) must be provided by TOE Design (ADV_TDS). The TOE Design (ADV_TDS) shall describe the mechanisms and the Security Architecture (ADV_ARC) shall describe the security architecture design and implementation to limit the availability of the Test Features. The Vulnerability Assessment (AVA) shall analyse the effectiveness of the security mechanisms to enforce FMT_LIM.1 and FMT_LIM.2. For further information on how to handle the Test Features refer to Section 6.2.1.

161 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[5].

162 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.

[5] [assignment: *Limited capability and availability policy*]

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[6].

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

FAU_SAS.1 Audit storage

Hierarchical to: No other components.

Dependencies: No dependencies.

FAU_SAS.1.1 The TSF shall provide the test process before TOE Delivery [7] with the capability to store [selection: *the Initialisation Data, Pre-personalisation Data, [assignment: other data]*] [8] in the [assignment: *type of persistent memory*].

Application Note 16: The integrity and uniqueness of the unique identification of the TOE must be supported by the development, production and test environment. For details refer to section 6.2.1.1.

Application Note 17: The test process is running under control of the test-personnel. The ST writer shall perform the operation in the element FAU_SAS.1.1 by assignment data and of the type of persistent memory provided for the storage of Initialisation Data and/or Pre-personalisation Data and/or other data e.g. like supplements of the Security IC Embedded Software. If the TOE provides specific functions to protect these data or to process them, appropriate security functional requirements can be specified in the Security Target. Then the above paragraph needs to be revised in addition.

Physical Manipulation and Probing

164 The TOE can be subject to "tampering" which here pertains to (i) manipulation of the chip hardware and its security features with (ii) prior reverse-engineering to

[6] [assignment: *Limited capability and availability policy*]

[7] [assignment: *list of subjects*]

[8] [assignment: *list of audit information*]

understanding the design and its properties and functions), (iii) determination of critical data through measuring using galvanic contacts, (iv) determination of critical data not using galvanic contacts and (v) calculated manipulation of memory contents. Refer to paragraph 70 (on page 22) for further explanations.

165 The TSF protects the user data stored in specified memory areas against compromise and undetected manipulation. The TSF provides access to the data in the memory through the specified interfaces only. The TSF protects the information of the user data stored in specified memory areas against compromise by physical access to the stored data bypassing these interfaces. Therefore, the security functional component Stored data confidentiality (FDP_SDC.1) has been selected The TSF may not prevent any manipulation of the memory content but shall monitor the memory for integrity errors and react on detected errors as required by Stored data integrity monitoring and action (FDP_SDI.2).

166 The TOE is not always powered and therefore not able to detect, react or notify that it has been subject to tampering. Nevertheless, its design characteristics make reverse-engineering and manipulations etc. more difficult. This is regarded as being an “automatic response” to tampering. Therefore, the security functional component Resistance to physical attack (FPT_PHP.3) has been selected. The TOE may also provide features to actively respond to a possible tampering attack which is also covered by FPT_PHP.3.

167 The TOE may also leave it up to the Security IC Embedded Software to react when a possible tampering has been detected. Comprehensive guidance (refer to Common Criteria assurance class AGD) will be given for the developer of the Security IC Embedded Software in this case. 10

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

FDP_SDC.1 Stored data confidentiality

Hierarchical to: No other components.

Dependencies: No dependencies.

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

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

FDP_SDI.2 Stored data integrity monitoring and action

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring

Dependencies: No dependencies.

FDP_SDI.2.1 The TSF shall monitor user data stored in containers controlled by the TSF for [assignment: *integrity errors*] on all objects, based on the following attributes: [assignment: *user data attributes*].

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall [assignment: *action to be taken*].

Application Note 18: The Security Target writer shall perform the open operations. It may assign the monitored memory areas as user attributes in the element FDP_SDI.2.1.

170 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 19: The Security Target shall describe the automatic response of the TOE. All security functional requirements are derived from security objectives to protect the user data stored and processed on the Security IC or to provide secure security services. Therefore, the security functional requirements are enforced if the TOE stops operation or does not operate at all if a physical manipulation or physical probing attack is detected and the security cannot be ensured in another way.

[9] [assignment: *physical tampering scenarios*]

[10] [assignment: *list of TSF devices/elements*]

Application Note 20: The TOE might ".. provide unambiguous detection of physical tampering that might compromise the TSF and … provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred" as requested by the elements FPT_PHP.1.1 and FPT_PHP.1.2. However the notification of the tampering is subject to the Security IC Embedded Software. The ST writer can highlight security features that support the detection of physical tampering so that the writer of a composite ST is able to define an associated security functional requirement.

Leakage

171 When the Security IC processes user data of the Composite TOE and/or TSF Data, information about these data may be leaked by signals which can be measured externally (e.g. the ISO contacts of the Smartcard). An attacker may also cause malfunctions or perform manipulations of the TOE in order to cause the TOE to leak information. The analysis of those measurement data can lead to the disclosure of user data of the Composite TOE and other critical data. Examples are given in Section 7.2.

172 The security functional requirements “Basic internal transfer protection (FDP_ITT.1)” and “Basic internal TSF data transfer protection (FPT_ITT.1)” have been selected to ensure that the TOE must resist leakage attacks (both for user data of the Composite TOE and TSF data). The corresponding security policy is defined in the security functional requirement “Subset information flow control (FDP_IFC.1)”. These security functional requirements address inherent leakage. With respect to forced leakage they have to be considered in combination with the security functional requirements “Limited fault tolerance (FRU_FLT.2)” and “Failure with preservation of secure state (FPT_FLS.1)” on the one hand and “Resistance to physical attack (FPT_PHP.3)” on the other.

173 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.

[11] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]

[12] [selection: *disclosure, modification, loss of use*]

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

174 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 cryptographic co-processor) are seen as separated parts of the TOE.**

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

175 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].

176 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

---

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]

confidential data only but without the distinction of attributes controlled by the Security IC Embedded Software."

Random Numbers

177 The TOE generates random numbers. To define the IT security functional requirements of the TOE an additional family (FCS_RNG) of the Class FCS (cryptographic support) is defined in chapter 5.1. This family FCS_RNG Generation of random numbers describes the functional requirements for random number generation used for cryptographic purposes.

178 The TOE shall meet the requirement "Quality metric for random numbers (FCS_RNG.1)" as specified below (Common Criteria Part 2 extended).

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, hybrid physical, hybrid deterministic*][16] 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*].

Application Note 21: The ST writer shall perform the open operations. The operation performed in the element FCS_RNG.1.1 selects RNG types based on physical random number generators as typical provided by Security IC. The chapter 7.5 provides examples for the security capabilities and quality metrics used in some national certification schemes.

6.2 Security Assurance Requirements for the TOE

179 The Security Target to be developed based upon this Protection Profile will be evaluated according to Security Target evaluation (Class ASE).

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

Evaluation Assurance Level 4 (EAL4)

[16] [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*]

and augmented by taking the following components:

ALC_DVS.2, and AVA_VAN.5.

181 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)
Security problem definition	(ASE_SPD.1)
TOE summary specification	(ASE_TSS.1)
Class ATE: Tests	
Coverage	(ATE_COV.2)
Depth	(ATE_DPT.2)
Functional tests	(ATE_FUN.1)
Independent testing	(ATE_IND.2)
Class AVA: Vulnerability assessment	
Vulnerability analysis	(AVA_VAN.5)

Application Note 22: This Protection Profile requires EAL4 augmented but allows to add higher hierarchical components. To support this most parts of the Protection Profile are, whenever possible, formulated independently from possible augmentations (for instance those to reach EAL5 augmented): Therefore, this Protection Profile often refers to "the Common Criteria assurance component of the family XY" instead of referring to the specific components listed above. If the Security Target uses further augmentations this must be identified in this section (and possibly in Section 2.3). The authors of the Security Target shall also review the rationale of this Protection Profile and extend it as appropriate.

6.2.1 Refinements of the TOE Assurance Requirements

182 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. [5], [6], [7], [8], [9] and [10]. 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.

183 The following refinements shall support the comparability of evaluations according to this Protection Profile. 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)

184 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 23: 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.

6.2.1.1 Refinements regarding Delivery procedure (ALC_DEL)

Introduction

185 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.

186 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":

187 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

188 **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 24: 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 25: 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 26: Depending on whether the TOE comprises programmable non-volatile memory and/or ROM, 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.

6.2.1.2 Refinements regarding Development Security (ALC_DVS)

Introduction

189 The JILWG published the document “Joint Interpretation Library: Minimum Site Security Requirements (For trial use), 2013” [12].

190 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.

191 The TOE Manufacturer must ensure that the development and production of the TOE (refer to Section 1.2.3) is secure so that no restricted, sensitive, critical or very critical information is unintentionally made available for the operational phase of the TOE which enables or support attacks (cf. [9] for details). Therefore confidentiality and integrity of design information and test data must be guaranteed, access to samples[17], 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.

[17] This may comprise so called open samples that are only used for evaluation purposes

192 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:

193 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

194 **“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, paragraph 65 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 27: 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

195 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.

6.2.1.3 Refinement regarding CM scope (ALC_CMS)

Introduction

196 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.

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

198 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

199 **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 (as implemented in the ROM or 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

200 Depending on the product type with programmable non-volatile memory and/or ROM 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.

201 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.

6.2.1.4 Refinement regarding CM capabilities (ALC_CMC)

Introduction

202 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.

203 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:

204 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

205 **“Configuration items” comprise all items defined and refined under ALC_CMS (see above) to be tracked under CM.**

206 **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.**

6.2.1.5 Refinements regarding Security Architecture (ADV_ARC)

Introduction

207 The "Supporting Document Guidance Security Architecture requirements (ADV_ARC) for smart cards and similar devices" [7] provides further guidance on how to apply the assurance requirements for the security architecture to security integrated circuits.

208 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

209 **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.**

210 **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.**

6.2.1.6 Refinements regarding Functional Specification (ADV_FSP)

Introduction

211 The Common Criteria assurance component of the family ADV_FSP (functional specification) 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.

212 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.

213 The intended user of the TOE is the Developer of the Security IC Embedded Software and the Composite TOE Manufacturer, refer to paragraph 188.

214 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

215 **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.**

216 **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.**

217 **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.**

218 **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

219 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 6.2.1.5. 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.

6.2.1.7 Refinements regarding Implementation Representation (ADV_IMP)

Introduction

220 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.

221 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."

222 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

223 **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.**

6.2.1.8 Refinement regarding Test Coverage (ATE_COV)

Introduction

224 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."

225 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

226 **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 EEPROM writing).**

227 **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

228 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.

6.2.1.9 Refinement regarding User Guidance (AGD_OPE)

Introduction

229 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.“

230 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.

231 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.

232 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

233 **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, refer also to paragraph 188.**

234 **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.**

235 **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

236 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.

6.2.1.10 Refinement regarding Preparative User Guidance (AGD_PRE)

Introduction

237 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.

238 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

239 **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.**

240 **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.**

241 **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.**

6.2.1.11 Refinement regarding Vulnerability Analysis (AVA_VAN).

Introduction

242 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."

243 Since [4] 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.

244 The following text reflects the requirements of the selected component AVA_VAN.5:

Developer action elements:

AVA_VAN.5.1D The developer shall provide the TOE for testing.

Content and presentation elements:

AVA_VAN.5.1C The TOE shall be suitable for testing.

Evaluator action elements:

AVA_VAN.5.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.5.2E The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.5.3E 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.5.4E 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 High attack potential.

Refinement

245 **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 this Protection Profile (refer to refinement regarding "Development Security (ALC_DVS)", section 6.2.1.2).**

Application Note 28: Evaluator may assess the ROM 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 29: 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 [8]. 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 30: 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 31: 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 6.1). 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.

6.3 Security Requirements Rationale

6.3.1 Rationale for the security functional requirements

246 Table 2 below gives an overview, how the security functional requirements are combined to meet the security objectives. The detailed justification follows after the table.

Objective	TOE Security Functional and Assurance Requirements
O.Leak-Inherent	- FDP_ITT.1 "Basic internal transfer protection" - FPT_ITT.1 "Basic internal TSF data transfer protection" - FDP_IFC.1 "Subset information flow control"

Objective	TOE Security Functional and Assurance Requirements
O.Phys-Probing	- FDP_SDC.1 “Stored data confidentiality” - FPT_PHP.3 “Resistance to physical attack”
O.Malfunction	- FRU_FLT.2 “Limited fault tolerance - FPT_FLS.1 “Failure with preservation of secure state”
O.Phys-Manipulation	- FDP_SDI.2 “Stored data integrity monitoring and action” - FPT_PHP.3 “Resistance to physical attack”
O.Leak-Forced	All requirements listed for O.Leak-Inherent - FDP_ITT.1, FPT_ITT.1, FDP_IFC.1 plus those listed for O.Malfunction and O.Phys-Manipulation - FRU_FLT.2, FPT_FLS.1, FPT_PHP.3
O.Abuse-Func	- FMT_LIM.1 “Limited capabilities” - FMT_LIM.2 “Limited availability” plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced - FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1
O.Identification	- FAU_SAS.1 “Audit storage”
O.RND	- FCS_RNG.1 “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

Table 2: Security Requirements versus Security Objectives

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

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

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

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

251 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.

252 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.

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

254 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.

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

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

257 It is possible that the TOE needs additional support by the Embedded Software (for instance by implementing FDP_SDI.1 to check data integrity with the help of appropriate checksums, refer to Section 6.1). This support must be addressed in the Guidance Documentation. Together with this FPT_PHP.3 is suitable to meet the objective.

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

259 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.

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

261 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.

262 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 2.

263 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.

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

265 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.

266 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.

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

268 FCS_RNG.1 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.

269 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.

270 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.

271 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.

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

6.3.2 Dependencies of security functional requirements

273 Table 3 below lists the security functional requirements defined in this Protection Profile, their dependencies and whether they are satisfied by other security requirements defined in this Protection Profile. The text following the table discusses the remaining cases.

Security Functional Requirement	Dependencies	Fulfilled by security requirements in this PP
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
FDP_SDI.2	None	No dependency
FCS_RNG.1	None	No dependency

Table 3: Dependencies of the Security Functional Requirements

274 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).

275 As Table 3 shows, all other dependencies of functional requirements are fulfilled by security requirements defined in this Protection Profile.

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

6.3.3 Rationale for the Assurance Requirements

277 The assurance level EAL4 and the augmentation with the requirements ALC_DVS.2, and AVA_VAN.5 were chosen in order to meet assurance expectations explained in the following paragraphs.

278 An assurance level of EAL4 with the augmentations AVA_VAN.5 and ALC_DVS.2 are required for this type of TOE since it is intended to defend against sophisticated

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

ALC_DVS.2 Sufficiency of security measures

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

280 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.

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

AVA_VAN.5 Advanced methodical vulnerability analysis

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

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

284 AVA_VAN.5 has dependencies to ADV_ARC.1 “Security architecture description”, ADV_FSP.2 “Security enforcing functional specification”, ADV_TDS.3 “Basic modular design”, ADV_IMP.1 “Implementation representation of the TSF”, AGD_OPE.1 “Operational user guidance”, and AGD_PRE.1 “Preparative procedures”.

285 All these dependencies are satisfied by EAL4.

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

287 Note that detailed refinements for assurance requirements are given in Section 6.2.1.

6.3.4 Security Requirements are Internally Consistent

288 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.

289 The security functional requirements FDP_SDC.1 and FDP_SDI.2 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.

290 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, and those implemented in the Security IC Embedded Software.

291 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, and those implemented in the Security IC Embedded Software.

292 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).

293 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.

294 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.

295 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.

296 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.

297 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.

298 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 functions19F19, it is important to limit their availability so that an attacker is not able to use them.

---

[18] or, in the extreme case, not being provided

[19] the capabilities are not limited in a perfect way (FMT_LIM.1)

299 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.

300 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 Annex

301 Note that Section 7.1 contains additional information which is used for the refinements of the standard assurance requirements (refer to Section 6.2) defined in the separate Section 6.2.1.

7.1 Development and Production Process (life-cycle)

302 The following section emphasises two different life-cycles for the hardware platform. The first life-cycle applies to hardware platforms which are customised by the IC Embedded Software implemented in the ROM. The second life-cycle applies to hardware platforms without customisation where the IC Embedded Software is downloaded to the programmable non-volatile memory (e.g. Flash products).

303 Note that the Protection Profile is also applicable for products where both life cycles are combined. In this case the hardware platform is customised by an initial IC Embedded Software part which is supplemented by further IC Embedded Software parts downloaded to the programmable non-volatile memory. This may be applicable for Java Cards.

7.1.1 Life-Cycle Description

304 The Security IC product life-cycle is visualised in Figure 16 for products with customised ROM.

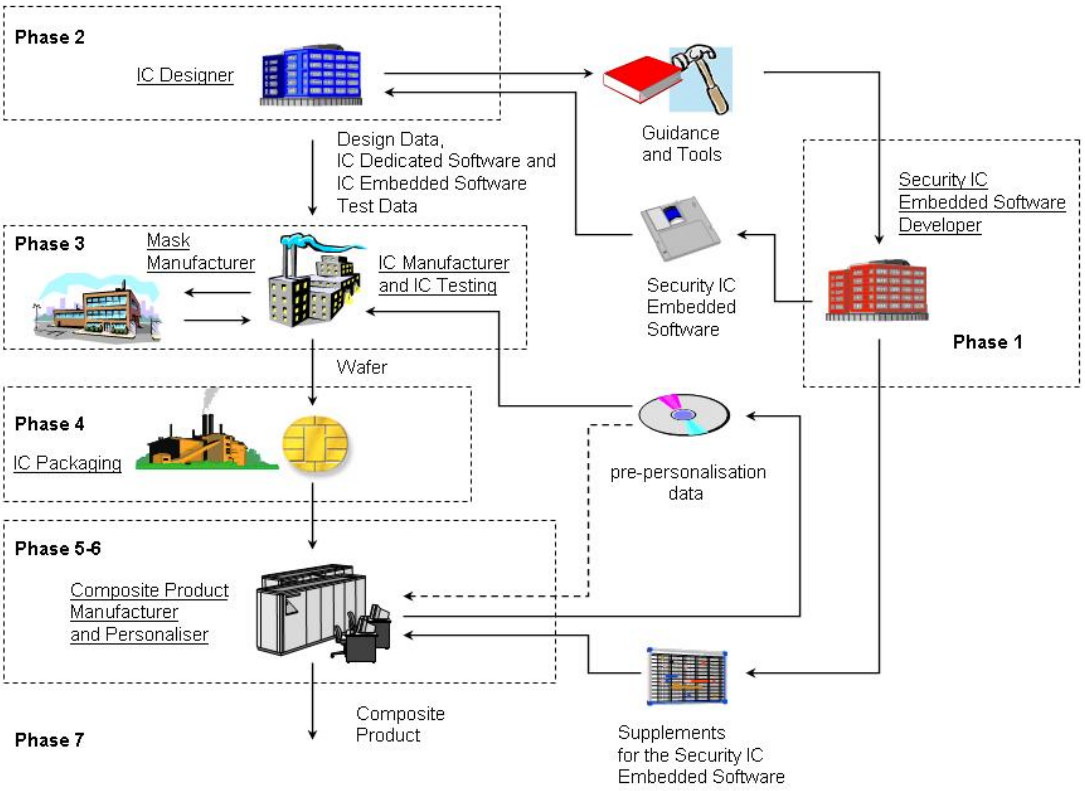


Figure 16: Security IC Life-Cycle if Security IC Embedded Software is implemented in ROM and EEPROM only

305 The Security IC product life-cycle for products without customisation of the hardware platform is visualised in Figure 17. In this case the Security IC Embedded Software is stored in programmable non-volatile memory.

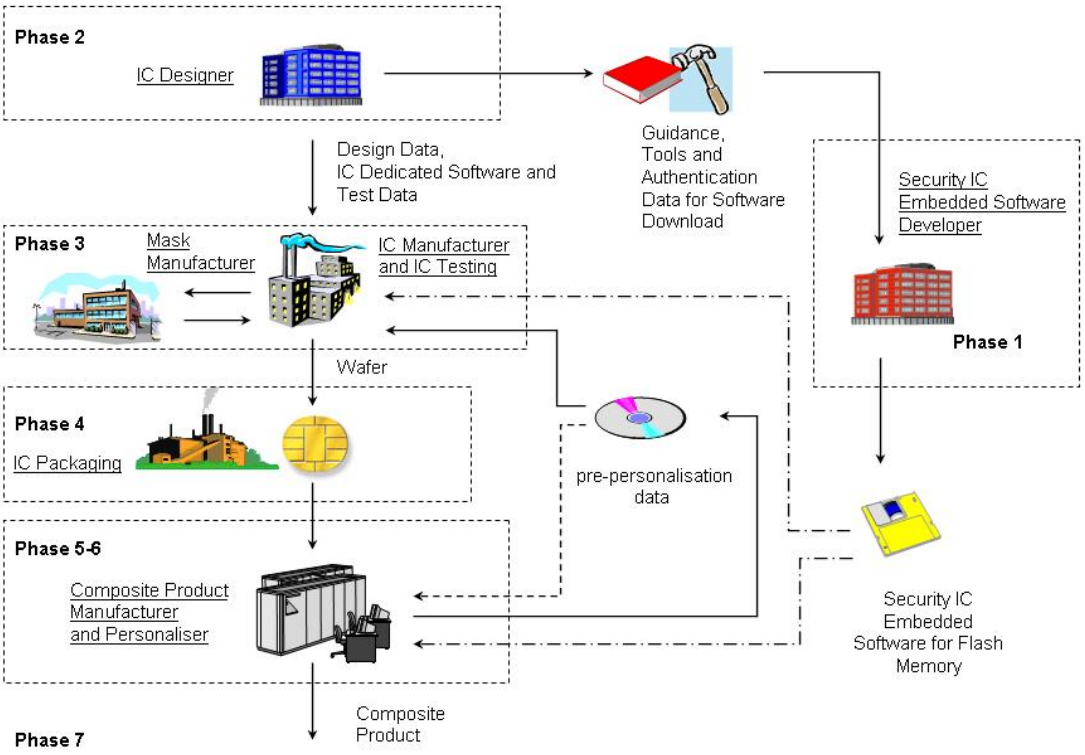


Figure 17: Security IC Life-Cycle if Security IC Embedded Software is loaded by Security IC Dedicated Software into the programmable non-volatile Memory

306 The Security IC product life-cycle is decomposed into seven phases where the following authorities are involved. For the main differences between the two life cycles depicted above refer to the foot notes in the table.

Phase 1	Security IC Embedded Software Development	The **Security IC Embedded Software Developer** is in charge of • the Security IC embedded software development and • the specification of IC pre-personalisation requirements, though the actual data for IC pre-personalisation come from Phase 6 (or Phase 4 or 5)[20].

[20] For Flash products this includes also requirements for the secured download of the Security IC Embedded Software.

Phase 2	IC Development	The **IC Designer** • designs the IC, • develops IC Dedicated Software, • provides information, software and tools to the Security IC Embedded Software Developer, and • receives the Security IC embedded software from the developer, through trusted delivery and verification procedures.[21] From the IC design, IC Dedicated Software and Security IC Embedded Software, the **IC Designer** • constructs the Security IC database, necessary for the IC photomask fabrication.
Phase 3	IC Manufacturing and Testing	The **IC Manufacturer** is responsible for • producing the IC through three main steps: IC manufacturing, IC testing, and IC pre-personalisation. The IC Mask Manufacturer • generates the photomasks for the IC manufacturing based upon an output from the Security IC database.

Application Note 32: If the Security IC Embedded Software is stored in a ROM the development of the software must be finished in Phase 1 and delivered to the TOE Manufacturer. If the Security IC Embedded Software is stored in a programmable non-volatile memory the TOE comprises a loader as part of the IC Dedicated Software and the Security IC Embedded Software can be downloaded. The download may be performed as service provided by the IC Manufacturer or IC Packaging Manufacturer for the Composite Product Integrator before TOE delivery or by the Composite Product Integrator after the TOE delivery. In the latter case the delivery of the Security IC Embedded Software to the TOE Manufacturer is not required and Phase 1 can be parallel to phase 2 to phase 4.

Phase 4	IC Packaging	The **IC Packaging Manufacturer** is responsible for • the IC packaging and testing.

[21] This item is not required if the TOE is a Flash product. In this case the TOE Manufacturer must provide the information for the download of the Security IC Embedded Software.

Application Note 33: Phase 4 can be part of the evaluation process, refer to section 1.2.3. Whether phase 4 is subject of the evaluation must be defined in the Security Target.

Phase 5	Security IC Product Finishing Process	The **Composite Product Manufacturer** is responsible for • the Security IC product finishing process and testing.
Phase 6	Security IC Personalisation	The **Personaliser** is responsible for • the Security IC personalisation and final tests.
Phase 7	Security IC End-usage	The **Security IC Issuer** is responsible for • the Security IC product delivery to the Security IC consumer, and the end of life process.

307 If the TOE comprises programmable non-volatile memory the Security IC Embedded Software may be loaded onto the chip in phase 3, 4, 5 or 6.

308 The relation between the semiconductor industry (TOE Manufacturer, refer to section 1.2.3, in particular comprising the roles IC Designer / IC Manufacturer and Mask Manufacturer) and the other parties being involved in the Security IC development and production (especially the Security IC Embedded Software Developer) are visualised in Figure 18.

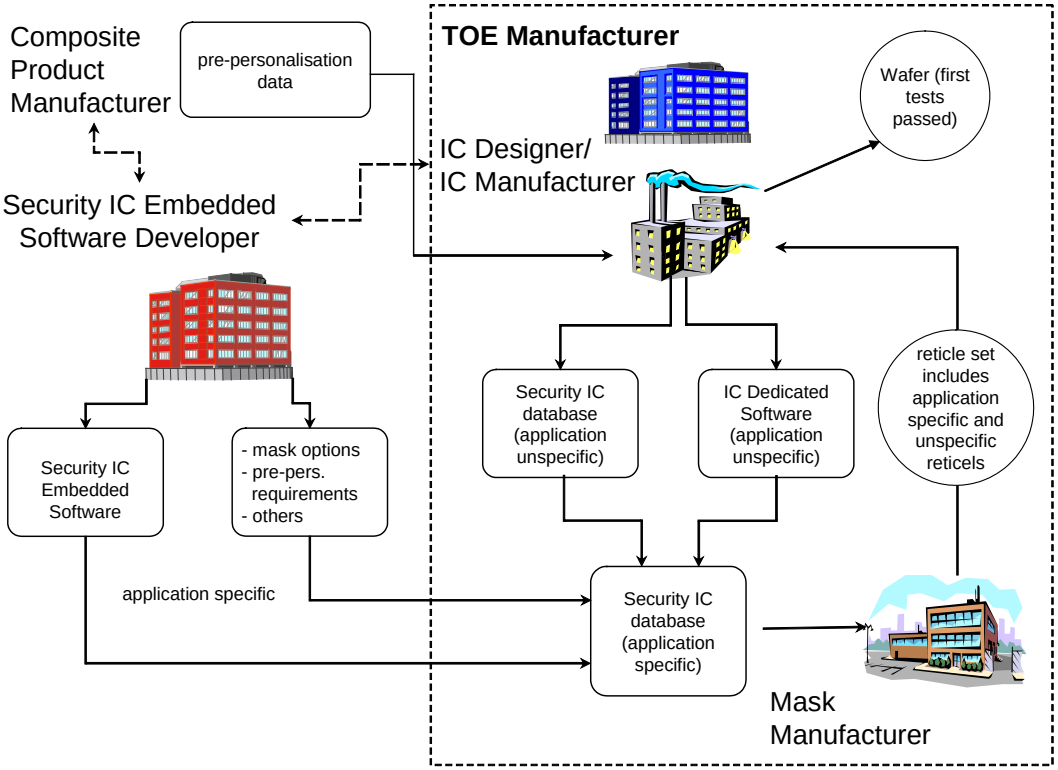


Figure 18: Development and Wafer Production including Testing in case of Embedded Software in ROM and EEPROM only

309 For Flash products and similar TOE the design of the hardware platform is not customised and the Security IC Embedded Software may not be delivered to the TOE Manufacturer. In this case the Security IC Embedded Software is loaded in a later phase. To ensure the control of the software download, sufficient authentication mechanisms must be implemented by the IC Dedicated Support Software. Associated authentication data and/or keys must be exchanged between the TOE Manufacturer and the developer of the Security IC Embedded Software.

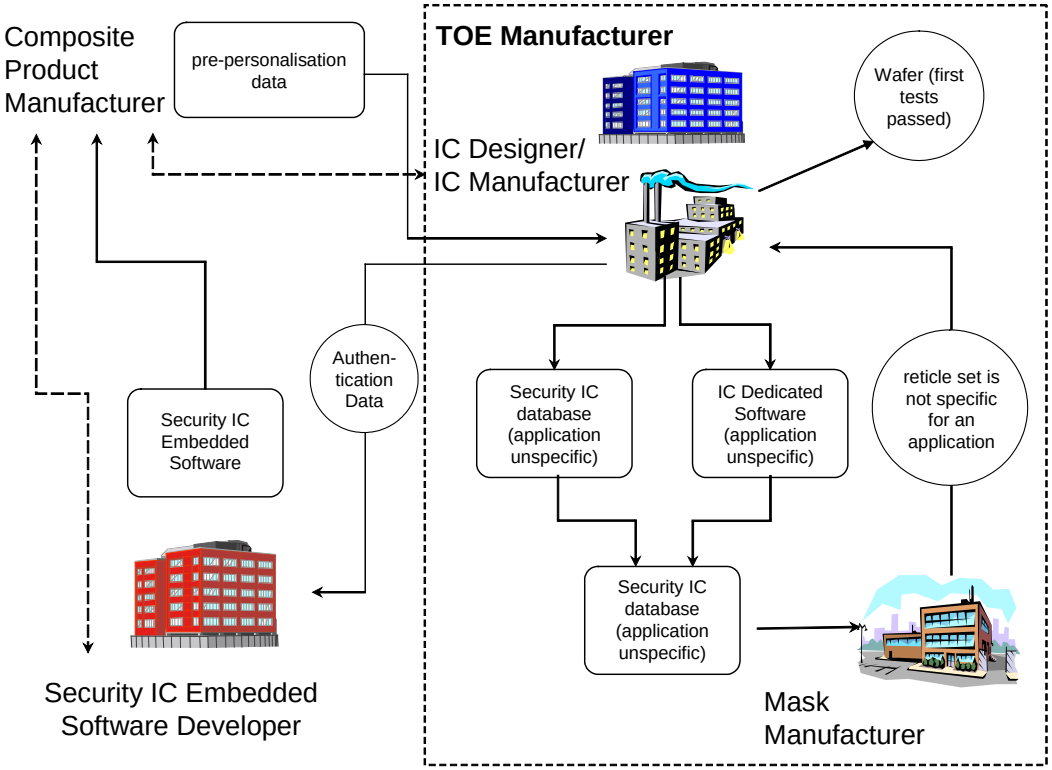


Figure 19: Development and Wafer Production including Testing in case of Embedded Software in programmable non-volatile memory only

310 The development process of the TOE starts with a process qualification. In parallel the concept of the TOE and the corresponding logical design is developed. The design uses standard library elements (circuitry and layout) which could be used for other (non security) integrated circuits but may include full custom elements specially designed for the TOE as well. Some cells have parameters: For instance the concrete layout of a ROM cell is determined by its contents which in turn is determined by the software or the data to be stored within.

311 All these “cells” not only differ in their logical or physical behaviour but also in their structure size which may range from very few elements such as simple gates up to physical units or sub-circuitry which may represent whole independent logical processing units. The physical “cells” (physical layout information is used) are placed on the chip area and then connected by wires (routing). Information about the physical layout of “cells”, about their position, about the shape of connecting wires and other process information define the physical layout of the chip.

312 These development steps are very complex. Only the development of the logical design might be similar to standard software development. However, technological constraints (such as timing) make this process more complicated and require for instance simulations which take technological and layout information into account. So, logical and physical design are developed in close relation.

313 The development of the information which defines the physical layout of an integrated circuit is a very complex matter. The photomasks or reticles required for wafer

production are basically produced based upon this information. However, a bunch of technology related parameters (possible even some depending on the wafer foundry) are taken into account in addition.

314 The photomasks or reticles are used to realise the integrated circuitry on/in a substrate. This again comprises tens of process steps each effecting the final result. Not only layout principles but process information is proprietary to IC Designers / IC Manufacturers. Each single chip (die or dice) is being tested after production.

315 The development and production is based upon a well established process of the manufacturer of the TOE. The processes are continuously developed and improved mainly in order to increase yield and reliability.

316 During integrated circuit development and production many information and material is produced as summarised in Section 7.1.2. The evaluator must concentrate on the security critical assets and exactly assess their storage and handling. It is not sufficient to assess a company as a whole, arguing that personnel is trustworthy and exchange of information and material with external partners is properly controlled.

7.1.2 Description of Assets of the Integrated Circuits Designer/Manufacturer

317 The assets of the manufacturer of the TOE to be protected during development and production of the TOE were already identified in paragraph 69 (page 21). Further explanatory text is given here.

318 The logical design data are those used to design the schematics of the chip (schematics or HDL sources and design documents). With the logical design data the functionality of the chip can be understood. The logical design data can be regarded as being independent from the actual implementation (layout) though they contain the timing characteristics of some functional units (circuitry blocks).

319 The physical design data comprises all topographic information (three dimensional) about parts of the chip or the whole chip. Topographic information is the absolute or relative position, form, thickness, length and size of any structures realised on the chip surface. These structures are pads, connecting wires, isolation layers, vias, and implants.

320 The IC Dedicated Software, Security IC Embedded Software (if delivered to the IC Designer/Manufacturer), Initialisation Data and Pre-personalisation Data comprises the source code including the related documents and the corresponding binaries as well as other data to be injected into the TOE before TOE Delivery.

Application Note 34: If the Security IC Embedded Software of the composite product is loaded by the Manufacturer of the composite product into the programmable non-volatile memory the IC Designer/Manufacturer and the Photomasks Manufacturer may not need to know this Embedded Software. In this case the pre-personalisation data will include authentication data to control the access provided by the loader as part of the IC Dedicated Software for loading the Embedded Software.

321 The specific development aids comprise all tools especially developed to produce the product. One important example is the “ROM translator” which produces the physical memory content from the software binaries.

322 The test and characterisation related data comprise all information, which is used for testing including test results (pre-layout, post layout and product) and the characterisation of the final chip.

323 The material for software development support comprises all information and material given to the Security IC Embedded Software Developer to support the development of the Security IC Embedded Software.

324 The photomasks and products comprises the photomasks or reticles (usable and scrap) and chips (usable and scrap) in different forms.

325 The requirements of the Common Criteria assurance family ALC_DVS apply to all the above items. This includes assessment of all sites being involved in the development and production of the product. Exceptions must be agreed with the certification body.

7.2 Package “Authentication of the Security IC”

326 This package enhances the unique identification of the TOE, with respect to authentication by external entities. The ST should include this package if the reliable identification of the TOE is required and masquerade of the genuine TOE is a threat in the operational environment, e.g. in the Security IC End-usage (Phase 7) respective Smartcard end-usage (Phase 7) according to [6].

7.2.1 Security Organisational Policy and Security Objective

327 The package adds the threat T.Masquerade_TOE in the security problem definition, the security objective O.Authentication for the TOE and the SFR FIA_API.1. The extended family FIA_API is defined in this section.

328 If this package “Authentication of the Security IC” is chosen the ST writer shall include the threat “Masquerade the TOE (T.Masquerade_TOE)” as specified below.

T.Masquerade_TOE **Masquerade the TOE**

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

329 The threat T.Masquerade_TOE may threaten the unique identity of the TOE as described in the P.Process-TOE or the property as being a genuine TOE without unique identity. Mitigation of masquerade requires tightening up the identification by authentication.

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

O.Authentication **Authentication to external entities**

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

331 The operational environment shall provide “External entities authenticating of the TOE (OE.TOE_Auth)”.

OE.TOE_Auth **External entities authenticating of the TOE**

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

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

7.2.2 Definition of the Family FIA_API

333 To describe the IT security functional requirements of the TOE a functional family FIA_API (Authentication Proof of Identity) of the Class FIA (Identification and authentication) is defined here. This family describes the functional requirements for the proof of the claimed identity by the TOE and enables the authentication verification by an external entity. The other families of the class FIA address the verification of the identity of an external entity by the TOE.

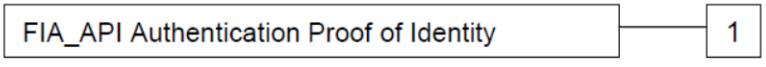
334 The other families of the Class FIA describe only the authentication verification of users’ identity performed by the TOE and do not describe the functionality of the user to prove their identity. The following paragraph defines the family FIA_API in the style of the Common Criteria part 2 (cf. [3], chapter “Extended components definition (APE_ECD)”) from a TOE point of view.

335 FIA_API Authentication Proof of Identity

Family Behaviour

This family defines functions provided by the TOE to prove its identity and to be verified by an external entity in the TOE IT environment.

Component levelling:



FIA_API.1 Authentication Proof of Identity, provides proof of the identity of the TOE, an object or an authorized user or role to an external entity.

Management	FIA_API.1 The following actions could be considered for the management functions in FMT: Management of authentication information used to prove the claimed identity.
Audit:	FIA_API.1 There are no actions defined to be auditable.

FIA_API.1	**Authentication Proof of Identity**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_API.1.1	The TSF shall provide a [assignment: *authentication mechanism*] to prove the identity of the [selection: *TOE*, *[assignment: object, authorized user or role]*] to an external entity.

7.2.3 Security Functional Requirement for Authentication of the TOE

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

FIA_API.1	**Authentication Proof of Identity**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_API.1.1	The TSF shall provide a [assignment: *authentication mechanism*] to prove the identity of the TOE[22] to an external entity.

[22] [selection: *TOE*, *[assignment: object, authorized user or role]*]

Application Note 35: The ST writer shall perform the open assignment in the element FIA_API.1.1. The authentication mechanism to be assigned could be a cryptographic mechanisms based on a key stored in protected memory of the TOE. The selection "TOE" defines the identity as authentic example of TOE (cf. O.Authentication) which is authenticated to an external entity (e.g. the Composite Product Manufacturer or the Personalisation agent). The proved identity depends on the set of TOE samples holding the same authentication verification data and the identity linked to the authentication reference data. E.g. the update of Security IC Embedded Software in life-cycle Phase 7 may need chip-individual private keys and chip-individual certificates of the corresponding public keys. In other use cases (e.g. electronic passports) the protection of personal data may be a concern and requiring many chip having the same private key.

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

7.3 Packages for Loader

338 This annex describes the security requirements for a Loader provided by the TOE. The Loader may be used to load data into the FLASH or EEPROM memory after delivery of the TOE. This Loader is considered as part of the TOE and is associated with the IC Dedicated Support Software (cf. para. 17).

339 The IC manufacturer may install Configuration Data, Initialisation Data and IC Dedicated Software and may be required by the Composite Product Integrator to install Security IC Embedded Software or other user data during the manufacturing process. The manufacturing tools and processes used are not available after TOE delivery and therefore they are not associated with these packages (but covered by the ALC class).

340 The IC Embedded Software may implement its own mechanism for loading data into EEPROM or Flash Memory but this functionality is out of scope of this PP.

341 The loaded data may be of different type and owner:

- IC Dedicated Support Software as part of the current TOE or a new TOE (cf. paragraph 346), or
- user data of the TOE as IC Embedded Software, TSF data or user data of the smartcard product.

342 The Loader may be used in different operational environments of the TOE:

- The Secured Environment maintains the confidentiality and integrity of the TOE as addressed by OE.Process-Sec-IC and the confidentiality and integrity of the IC Embedded Software, TSF data or user data associated with the smartcard product by security procedures of the smartcard product manufacturer, personaliser and other actors before delivery to the smartcard end-user depending on the smartcard life-cycle.
- The operational environments including "Phase 7 Security IC End-usage" (cf. also

"Phase 7 Smartcard end-usage" [6]) requiring self-protected TOE controlling access to the Loader and protecting the loaded data. The authorized user like IC manufacturer, the smartcard product manufacturer, personaliser or smartcard issuer need reliable identification of the TOE for loading or modification of IC Dedicated Support Software, IC Embedded Software, TSF data or user data of the smartcard product.

343 The packages address different functionality and method of use of the Loader:

- Package 1: Loader dedicated for usage in Secured Environment only
  - o limited capability of the Loader protecting user data in the writable memory areas,
  - o blocking of the Loader after intended usage (e.g. delivery to end-customer before Security IC End-usage phase) addressed by limited availability.
- Package 2: Loader dedicated for usage by authorized users only
  - o protection of the TOE user data against misuse of the Loader,
  - o trusted channel between Security IC and the authorised role to change the user data by means of the Loader,
  - o checking the integrity and the authenticity of the data provided by the authorized user to the Loader,
  - o access control on Loader usage.

344 The Package 1 comprise a base line set of security functionality of the Loader and assumes secure usage of the Loader in the Secured Environment. It requires the Composite Product Manufacturer to enable the protection against misuse of the Loader after intended usage and before delivery to the end-user in life-cycle Phase 7.

345 The Package 2 aims on use cases where the users have different security policies and authorization to load or modify data in the writable memory. E.g. the intended usage of the TOE may limit the loading of IC Dedicated Support Software to users authorized by IC manufacturer and the loading of Security IC Embedded Software to the Composite Product Integrator. The Loader may not distinguish between data as being IC Dedicated Support Software or Security IC Embedded Software but - as minimum – data as targeted to specific to memory areas. The Package 2 may be useful for Secured Environment as well.

346 The import and installation of user data, TSF data and IC Dedicated Support Software by means of the Loader are addressed by the security functional and assurance requirements in different way even if similar or the same mechanisms are used. If the Loader is used to import user data or TSF data the TOE is unchanged. The import of TSF data as management of TSF data does not change the TOE as well. The data of IC Dedicated Support Software will be loaded as user data but the installation of these data changes the TOE because the IC Dedicated Support Software is part of the Security IC TSF implementation.

347 The requirements of the Loader security functionality for loading user data of the composite product are described mainly by classes FDP: User Data Protection and FTP: Trusted Path/Channels and the family “Limited capabilities and availability (FMT_LIM)”.

348 The import of TSF data is not addressed by components of CC part 2 [2] but the management and the protection of TSF data. These specific security requirements shall be described by components of class FMT: Security Management (e.g. of family FMT_MTD: Management of TSF Data) and class FPT: Protection of TSF Data (e.g. of family Inter-TSF TSF data consistency (FPT_TDC) independent whether the Loader or any other mechanism is used.

349 The import of IC Dedicated Support Software will be loaded as user data. The installation of IC Dedicated Support Software changes the TOE itself and installs a new TOE. This new TOE comprises of the hardware of the previous TOE, parts of the IC dedicated Support Software which was left unchanged and is used by the new TOE, and the new IC Dedicated Support software (parts or complete). The certification of the new TOE shall be based on evaluation of the combination of all these parts. The update of the TOE may or may not include updated or new TSF data. The import of update data may be described as import of user data and the installation of the update shall be addressed by the assurance families Delivery (ALC_DEL) and Preparative procedures (AGD_PRE).

7.3.1 Package 1: Loader dedicated for usage in secured environment only

350 The “Package 1: Loader dedicated for usage in secured environment only” is intended for Loader only used in operational environments which are under control of the owner of the loaded data or its subcontractor. This is typically the Composite Product Manufacturer or more specifically the IC Packaging Manufacturer (cf. chapter 1.2.3 TOE life cycle and 7.1.1 Life-Cycle Description for details).

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

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

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

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

manipulation.

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

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

354 The organisational security policy Limitation of capability and blocking the Loader (P.Lim_Block_Loader) is directly implemented by the security objective for the TOE “Capability and availability of the Loader (O.Cap_Avail_Loader)” and the security objective for the TOE environment “Limitation of capability and blocking the Loader (OE.Lim_Block_Loader)”. The TOE security objective “Capability and availability of the Loader” (O.Cap_Avail_Loader)” mitigates also the threat “Abuse of Functionality “ (T.Abuse-Func) if attacker tries to misuse the Loader functionality in order to manipulate security services of the TOE provided or depending on IC Dedicated Support Software or user data of the TOE as IC Embedded Software, TSF data or user data of the smartcard product.

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

FMT_LIM.1/Loader	**Limited capabilities**
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.2 Limited availability.
FMT_LIM.1.1/Loader	The TSF shall 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: Deploying Loader functionality after [assignment: *action*] does not allow stored user data to be disclosed or manipulated by unauthorized user [23].
Application Note 36:	FMT_LIM.1 supplements FMT_LIM.2 allowing for non-overlapping loading of user data and protecting the TSF against misuse of the Loader for attacks against the TSF. The TOE Loader may allow for correction of already loaded user data before the assigned action e.g. before blocking the TOE Loader for TOE Delivery to the end-customer or any intermediate step in the life cycle of the Security IC or the smartcard.

[23] [assignment: *Limited capability policy*]

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

FMT_LIM.2/Loader	**Limited availability - Loader**
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.1 Limited capabilities.
FMT_LIM.2.1/Loader	The TSF shall be designed in a manner that limits its availability so that in conjunction with “Limited capabilities (FMT_LIM.1)” the following policy is enforced: The TSF prevents deploying the Loader functionality after [assignment: *action*][24].

Application Note 37: This is the easiest variant of Loader functionality relying on secure boot loading procedures in secure environment before TOE delivery to the assigned customer and preventing deploying the Loader of the Security IC after assigned action, e.g. after blocking of Loader for TOE delivery to the end-customer.

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

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

358 The Package 2: Loader dedicated for usage by authorized users only defines a more rigorous security functionality for the Loader. The ST shall include this package if the Loader is intended to be used in Phase 7: Operational usage of the life cycle. The package may be used also if the Loader is intended to be used after delivery by the TOE Manufacturer in Phase 4: IC Packaging, Phase 5: Composite Product Integration and Phase 6: Personalisation, e.g. by different authorized users.

359 This package is intended to support access control on usage of the Loader, mutual authentication of the TOE and the authorized user as end-points of a trusted channel and protection of integrity and confidentiality of the loaded data.

360 The ST may include additionally the Package “Authentication of the Security IC” for authentication of the TOE as end point of the trusted channel.

361 The ST may include additionally the “Package 1: Loader dedicated for usage in secured environment only” if limitation of capabilities and availability is provided by the TOE.

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

[24] [assignment: *Limited availability policy*]

P.Ctlr_Loader **Controlled usage to Loader Functionality**

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

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

O.Ctrl_Auth_Loader **Access control and authenticity for the Loader**

The TSF provides trusted communication channel with authorized user, supports confidentiality protection and authentication of the user data to be loaded and access control for usage of the Loader functionality.

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

OE.Loader_Usage **Secure communication and usage of the Loader**

The authorized user must support the trusted communication channel with the TOE by confidentiality protection and authenticity proof of the data to be loaded and fulfilling the access conditions required by the Loader.

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

366 The TOE Functional Requirement “Inter-TSF trusted channel (FTP_ITC.1)” is specified as follows.

FTP_ITC.1	**Inter-TSF trusted channel**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FTP_ITC.1.1	The TSF shall provide a communication channel between itself and **[assignment: *users authorized for using the Loader*]** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2	The TSF shall permit another trusted IT product[25] to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for deploying Loader [assignment: *rules*][26].

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

FDP_UCT.1	**Basic data exchange confidentiality**
Hierarchical to:	No other components.
Dependencies:	[FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_UCT.1.1	The TSF shall enforce the Loader SFP[27] to receive[28] user data in a manner protected from unauthorised disclosure.

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

FDP_UIT.1	**Data exchange integrity**
Hierarchical to:	No other components.
Dependencies:	[FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset

[25] [selection: *the TSF, another trusted IT product*]

[26] [assignment: *list of functions for which a trusted channel is required*]

[27] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]

[28] [selection: *transmit, receive*]

information flow control]

FDP_UIT.1.1	The TSF shall enforce the Loader SFP[29] to receive[30] user data in a manner protected from modification, deletion, insertion[31] errors.
FDP_UIT.1.2	The TSF shall be able to determine on receipt of user data, whether modification, deletion, insertion[32] has occurred.

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

FDP_ACC.1/ Loader	**Subset access control - Loader**
Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control.
FDP_ACC.1.1/ Loader	The TSF shall enforce the Loader SFP [33] on (1) the subjects [assignment: *authorized roles for using Loader*], (2) the objects user data in [assignment: *memory areas*], (3) the operation deployment of Loader [34]

Application Note 38: The TOE enforces the Loader SFP by FTP_ITC.1, FDP_UCT.1 and FDP_UIT.1 and FDP_ACF.1 to describe additional access control rules.

370 The TOE Functional Requirement "Security attribute based access control - Loader (FDP_ACF.1/Loader)" is specified as follows.

FDP_ACF.1/ Loader	**Security attribute based access control - Loader**
Hierarchical to:	No other components.
Dependencies:	FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1/	FDP_ACF.1.1 The TSF shall enforce the Loader SFP [35] to objects based on the following:

[29] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]

[30] [selection: *transmit, receive*]

[31] [selection: *modification, deletion, insertion, replay*]

[32] [selection: *modification, deletion, insertion, replay*]

[33] [assignment: *access control SFP*]

[34] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

Loader	(1) the subjects [assignment: a*uthorized roles for using Loader*] with security attributes [assignment: *SFP-relevant security attributes, or named groups of SFP-relevant security attributes*] (2) the objects user data in [assignment: *memory areas*] with security attributes [assignment: *SFP-relevant security attributes, or named groups of SFP-relevant security attributes*][36].
FDP_ACF.1.2/ Loader	FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*].
FDP_ACF.1.3/ Loader	FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*].
FDP_ACF.1.4/ Loader	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*].
Application Note 39:	The ST writer shall perform the open operations in the component of FDP_ACF.1/Loader in order to describe additional access control rules. The open assignment of security attributes may be empty. If the TOE supports blocking of the Loader as stated in the SFR FMT_LIM.2.1/Loader in Package 1 the ST may describe this blocking as rules for explicitly denying access in element FDP_ACF.1.4/Loader.

371 The SFR FMT_MSA.3 will not be necessary if the security attributes used to enforce the Loader SFP are fixed by the IC manufacturer and no new objects under control of the Loader SFP are created. The ST writer may be or may not be required to describe SFR by FMT_MSA.3 depending on any management of the relevant security attributes enforcing the Loader SFP.

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

- The SFR FDP_ACC.1/Loader defines the subjects, objects and operations of the Loader SFP enforced by the SFR FTP_ITC.1, FDP_UCT.1, FDP_UIT.1 and FDP_ACF.1/Loader.

---

[35] [assignment: *access control SFP*]

[36] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

- The SFR FTP_ITC.1 requires the TSF to establish a trusted channel with assured identification of its end points and protection of the channel data from modification or disclosure.

- The SFR FDP_UCT.1 requires the TSF to receive data protected from unauthorised disclosure.

- The SFR FDP_UIT.1 requires the TSF to verify the integrity of the received user data.

- The SFR FDP_ACF.1/Loader requires the TSF to implement access control for the Loader functionality.

7.4 Packages for Cryptographic Services

373 This chapter defines packages for optional cryptographic services provided by many but not all TOE.

374 The cryptographic security services described in these packages implement the same organizational security policy but in different extend.

P.Crypto-Service Cryptographic services of the TOE

The TOE provides secure hardware based cryptographic services for the IC Embedded Software.

375 The following packages describe how this organizational security policy P.Crypto-Services being part of these packages may be implemented by specific security objectives, are directly implemented by specific SFR of the class “Cryptographic Support”.

376 The dependency of the SFR for cryptographic operation shall be solved in the ST depending on the concrete TSF provided. This may be pure hardware implementation of the cryptographic algorithm or more complex combination of hardware and IC Dedicated Support Software.

7.4.1 Package “TDES”

377 The TOE shall provide “Cryptographic service Triple-DES (O.TDES)” as specified below.

O.TDES **Cryptographic service Triple-DES**

The TOE provides secure hardware based cryptographic services implementing the Triple-DES for encryption and decryption.

378 The security objective “Cryptographic service Triple-DES (O.TDES)” enforces the organizational security policy P.Crypto-Service.

379 The TOE shall meet the requirement “Cryptographic operation - TDES (FCS_COP.1/ TDES)” as specified below.

FCS_COP.1/ TDES	**Cryptographic operation – TDES**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1/ TDES	The TSF shall perform encryption and decryption[37] in accordance with a specified cryptographic algorithm TDES in [selection: *ECB mode, CBC mode*][38] and cryptographic key sizes [selection: *112 bit, 168 bit*][39] that meet the following NIST SP 800-67 [21], NIST SP 800-38A [22][40].

Application Note 40: Note FIPS 46-3 [14] was withdrawn in 2005. The Triple Data Encryption Algorithm with 112 bit and 168 bit keys is still an NIST approved cryptographic algorithm as defined in NIST SP 800-67 [21].

380 The TOE shall meet the requirement “Cryptographic key destruction – TDES FCS_CKM.4/TDES)” as specified below.

FCS_CKM.4/TDES	**Cryptographic key destruction - TDES**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4.1/ TDES	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: *cryptographic key destruction method*] that meets the following: [assignment: *list of standards*].

Application Note 41: The cryptographic key destruction may be provided by overwriting the internal stored key when a new key value is provided through the key interface or a key zeroize initiated by special signal.

[37] [assignment: *list of cryptographic operations*]

[38] [assignment: *cryptographic algorithm*]

[39] [assignment: *cryptographic key sizes*]

[40] [assignment: *list of standards*]

381 The FCS_COP.1/TDES and FCS_CKM.4/TDES meet the security objective “Cryptographic service Triple-DES (O.TDES)”.

382 The dependency of FCS_COP.1/TDES from FCS_CKM.4/TDES is fulfilled within the package.

383 The FCS_COP.1/TDES depends from [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]. This PP leave the decision to the ST writer not preferring one the alternative for user data import by including or excluding the respective SFR component FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes. This PP leave the decision to the ST writer for source of the keys to be used not preferring one the alternative including or excluding the respective SFR component key import by FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or for key generation by FCS_CKM.1 Cryptographic key generation.

384 The FCS_CKM.4/TDES depends from [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]. This PP leave the decision to the ST writer for the source of the keys to be destroyed not preferring one the alternative including or excluding the respective SFR component FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or for key generation by FCS_CKM.1 Cryptographic key generation.

7.4.2 Package “AES”

385 The TOE shall provide “Cryptographic service AES (O.AES)” as specified below.

O.AES	**Cryptographic service AES**
	The TOE provides secure hardware based cryptographic services for the AES for encryption and decryption.

386 The security objective “Cryptographic service AES (O.AES)” enforces the organizational security policy P.Crypto-Service.

387 The TOE shall meet the requirement “Cryptographic operation – AES (FCS_COP.1/AES)” as specified below.

FCS_COP.1/AES	**Cryptographic operation – AES**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/ AES	The TSF shall perform decryption and encryption[41] in accordance with a specified cryptographic algorithm AES in [selection: *ECB mode, CBC mode*][42] and cryptographic key sizes [selection: *128 bit, 192 bit, 256 bit*][43] that meet the following: FIPS 197 [16]], NIST SP 800-38A [22][44].

388 The TOE shall meet the requirement "Cryptographic key destruction (FCS_CKM.4)" as specified below.

FCS_CKM.4/AES	**Cryptographic key destruction**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4.1	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: *cryptographic key destruction method*] that meets the following: [assignment: *list of standards*].

Application Note 42: The cryptographic key destruction may be provided by overwriting the internal stored key when a new key value is provided through the key interface or a key zeroization initiated by special signal.

389 The FCS_COP.1/AES and FCS_CKM.4/AES meet the security objective "Cryptographic service AES (O.AES)".

390 The dependency of FCS_COP.1/AES from FCS_CKM.4/AES is fulfilled within the package.

391 The FCS_COP.1/AES depends from [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]. This PP leave the decision to the ST writer not preferring one the alternative for user data import by including or excluding the respective SFR component FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes. This PP leave the decision to the ST writer for source of the keys to be used not preferring one the alternative including or excluding the respective SFR component key import by FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or for key generation by FCS_CKM.1 Cryptographic key generation.

[41] [assignment: *list of cryptographic operations*]

[42] [assignment: *cryptographic algorithm*]

[43] [assignment: *cryptographic key sizes*]

[44] [assignment: *list of standards*]

392 The FCS_CKM.4/AES depends from [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]. This PP leave the decision to the ST writer for the source of the keys to be destroyed not preferring one the alternative including or excluding the respective SFR component FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or for key generation by FCS_CKM.1 Cryptographic key generation.

7.4.3 Package "Hash functions"

393 The TOE shall provide "Cryptographic service Hash function (O.SHA)" as specified below.

O.SHA	**Cryptographic service Hash function** The TOE provides secure hardware based cryptographic services for secure hash calculation.

394 The security objective "Cryptographic service Hash function (O.SHA)" enforces the organizational security policy P.Crypto-Service.

395 The TOE shall meet the requirement "Cryptographic operation – SHA (FCS_COP.1/SHA)" as specified below.

FCS_COP.1/SHA	**Cryptographic operation – SHA**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1/ SHA	The TSF shall perform hashing[45] in accordance with a specified cryptographic algorithm [selection: *SHA-1, SHA-224, SHA-256, SHA-384, SHA-512*][46] and cryptographic key sizes none[47] that meet the following FIPS 180-4 [15][48].
Application Note 43:	Note the keyed hash function use the same cryptographic base function but uses the concatenation of a key and other data which is normally performed in software.

[45] [assignment: *list of cryptographic operations*]

[46] [assignment: *cryptographic algorithm*]

[47] [assignment: *cryptographic key sizes*]

[48] [assignment: *list of standards*]

396 The FCS_COP.1/SHA meet the security objective “Cryptographic service SHA (O.SHA)”.

397 The FCS_COP.1/Hash depends from [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]. This PP leave the decision to the ST writer not preferring one the alternative for user data import by including or excluding the respective SFR component FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes. Because no key is used there is no need for key import as required by dependency to FDP_ITC.1, FDP_ITC.2 or key generation as required by dependency to FCS_CKM.1 or destruction as required by dependency to FCS_CKM.4.

7.5 Guidance for SFR for RNG (informative only)

398 This chapter provides informative examples of security requirements defined for RNG in some national certification schemes and how to perform the operations in the SFR FCS_RNG.1.

7.5.1 German Scheme

399 The Bundesamt für Sicherheit in der Informationstechnik (BSI) published mandatory evaluation requirements for the German Common Criteria certification scheme [17]. These documents describe predefined classes PTG.2, PTG.3 and DRG.4 of random number generators (cf. [18]) appropriate for the TOE of this protection profile.

400 If the ST selects the most commonly used pre-defined class PTG.2 the SFR “Random Number Generation – PTG.2 (FCS_RNG.1/PTG.2)” will look like this (without performed operation, cf. application note).

FCS_RNG.1/PTG.2 Random number generation – PTG.2

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1/PTG.2 The TSF shall provide a physical [49] random number generator that implements:

(PTG.2.1) A total failure test detects a total failure of entropy source immediately when the RNG has started. When a total failure is detected, no random numbers will be output.

(PTG.2.2) If a total failure of the entropy source occurs while the RNG is being operated, the RNG [selection: *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, generates*

[49] [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*]

the internal random numbers with a post-processing algorithm of class DRG.2 as long as its internal state entropy guarantees the claimed output entropy].

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

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

(PTG.2.5) The online test procedure checks the quality of the raw random number sequence. It is triggered [selection: *externally, at regular intervals, continuously, 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[50].

FCS_RNG.1.2/PTG.2 The TSF shall provide [selection: *bits, octets of bits, numbers [assignment: format of the numbers]*] that meet

(PTG.2.6) Test procedure A [assignment: *additional standard test* suites] does not distinguish the internal random numbers from output sequences of an ideal RNG.

(PTG.2.7) The average Shannon entropy per internal random bit exceeds 0.997[51].

Application Note 44: The ST writer shall perform the missing operation appropriate for cryptographic application of the random numbers in the elements FCS_RNG.1.1 and FCS_RNG_1.2. The ST writer shall perform the selections for specification of the security capabilities provided by the random number generator of the TOE. The assignment for additional standard statistical test suite in clause (PTG.2.6) maybe empty.
The evaluation of the random number generator shall follow a recognized methodology, e.g. AIS31 cf. [17].

7.5.2 NIAP

401 The following two informative examples show how FCS_RNG.1 may be used for SFR of physical RNG and hybrid deterministic RNG meeting the security requirements and designs of cryptographic post-processing in [19] and [20].

402 The National Institute of Standards and Technology (NIST) published NIST Special Publication 800-90A Recommendation for Random Number Generation Using

[50] [assignment: *list of security capabilities*]

[51] [assignment: *a defined quality metric*]

Deterministic Random Bit Generators, January 2012 [19] and NIST DRAFT Special Publication 800-90B Recommendation for the Entropy Sources Used for Random Bit Generation, August 2012 [20]. The draft recommendation for entropy sources [20] describes security requirements and test procedures that may be applied to the entropy source of a deterministic random number generator or a physical random number generator of the TOE. [19] defines hybrid deterministic RNG designs. Note [19] is currently under construction and only the designs based on block ciphers and hash functions should be used.

403 If the TOE shall implement a physical random number generator as entropy source compliant to [20] the ST writer may define a SFR "Random Number Generation – ES (FCS_RNG.1/ES)" like this:

FCS_RNG.1/ES Random number generation

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1/ES The TSF shall provide a ***physical***[52] random number generator that implements:

(ES.1) Failure or severe degradation of the noise source shall be detectable.

(ES.2) Continuous tests or other mechanisms in the entropy source shall protect against producing output during malfunctions.

(ES.3) [assignment: *list of additional security capabilities*][53].

FCS_RNG.1.2/ES The TSF shall provide [selection: *bits, octets of bits, numbers [assignment: format of the numbers]*] that meet

(ES.4) each output bit is independent of all other output bits,

(ES.5) [selection:

(ES.5a) *full entropy output*,

(ES.5b) *[assignment: bias and entropy rate of the output]*] [54].

404 Note [20] is still a draft under construction. The clause (ES.3) may describe conditioning components implementing NIST approved or non-approved cryptographic functions, which are optional in [20]. A full entropy source provides bit strings output containing at least $(1-\varepsilon)n$ bits entropy, where n is the length of each output string and $0 \le \varepsilon \le 2^{-64}$.

[52] [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*]

[53] [assignment: *list of security capabilities*]

[54] ([assignment: *a defined quality metric*]

405 If the TOE shall implement hybrid random number generator of the TOE complying to [19] seeded by a physical random number generator as entropy source described above the ST writer may define a SFR "Random Number Generation – Hybrid deterministic RNG (FCS_RNG.1/HD)" like this:.

FCS_RNG.1/HD Random number generation – Hybrid deterministic RNG

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1/HD The TSF shall provide a *hybrid deterministic*[55] random number generator that implements: [selection: *CTR_DRBG, Hash_DRBG, HMAC_DRBG*] as defined in NIST Special Publication 800-90A [19][56].

FCS_RNG.1.2/HD The TSF shall provide [selection: *bits, octets of bits, numbers [assignment: format of the numbers]*] that meet [assignment: *security bits*][57].

406 Please refer to NIST Special Publication 800-90A [19] for details about the security capabilities and the security bits as quality metric of the random number output.

7.6 Examples of Attack Scenarios

407 In this section background information is given to better understand the threats defined in section 3.2. The different types of influences on or interactions with the Security IC were already visualised in Figure 8. The contents of this section shall not be considered as being complete nor as a comprehensive guidance for the evaluation.

408 A standard tool used for electrical measurement (and application of voltage and injection of current) is the needle probe workstation. Often appropriate contact areas must be prepared before using the methods described above (refer to the threat T.Phys-Manipulation). The actual measurement is done using standard tools such as voltmeters, oscilloscopes and signal analysers.

409 In addition, there are indirect methods for measurements not requiring a direct (metallic) contact. Examples are voltage contrast imaging and electron probe microscopy. These methods are also referred to as physical probing since the Security IC must be prepared before using the methods described above (refer to the threat T.Phys-Manipulation).

410 The interface for the attack is (the Security IC carrier and then) the surface of the integrated circuit.

---

[55] [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*]

[56] [assignment: *list of security capabilities*]

[57] [assignment: *a defined quality metric*]

411 The application of appropriate combinations of such methods in order to reveal information (via a non-standard interface) are addressed by the threat T.Phys-Probing.

412 Malfunctions of the TOE may cause some of its TSF to fail to be effective. Often more critical, security functions (or mechanisms) of the Security IC Embedded Software may fail to be effective. This can be utilised by an attacker. The most straightforward way to cause malfunctions are irregular operating conditions in amplitude, shape, timing, occurrence etc. on the ISO interface (for instance such as glitches). Malfunctions can be due to errors or premature ageing.

413 The attacker stimulates the ISO interface (power supply, the external clock, reset and/or I/O). The attacker may also consider other types of influences on the Security IC or directly onto the surface of the integrated circuit. In the latter case it might be required to manipulate the Security IC (refer to the threat T.Phys-Manipulation). In addition, the attacker needs to observe the behaviour of the Security IC and immediately take advantage of a possible malfunction. This requires to have additional equipment such as a terminal and communication software, but may include other things depending on the application to be attacked.

414 The application of appropriate combinations of such methods in order to manipulate the Security IC Embedded Software (or the IC Dedicated Test Software) while being executed (via a standard interface) are addressed by the threat T.Malfunction.

415 Specific sorts of malfunctions are a means to reveal information about cryptographic keys or other critical data. Such methods are addressed by the threat T.Leak-Forced.

416 Standard tools used for the manipulation of circuitry are the Focused Ion Beam (FIB) and the laser cutter. The contents of programmable memories (such as EEPROM) may be modified for instance by manipulation of circuitry, by exposing cells to charged particle beams, by using electromagnetic waves or by electrical probing (application of voltage and injection of current).

417 Manipulations require prior extensive reverse-engineering. The methods being applied are for instance optical inspection, voltage contrast imaging, image processing and pattern matching. In order to analyse circuitry the chip hardware must be removed from its carrier and then de-layered using appropriate methods (wet etching, plasma etching, grinding).

418 The interface for the attack is (the Security IC carrier and then) the surface of the integrated circuit.

419 The application of appropriate combinations of such methods in order to perform manipulations are addressed by the threat T.Phys-Manipulation.

420 When the Security IC processes user data of the Composite TOE and other critical data information about these data may be contained in signals which can be measured on the ISO contacts of the Security IC using standard tools such as voltmeters, oscilloscopes and signal analysers. The Security IC may also produce emanation which can be received using an antenna and analysed. For the analysis of the measured data specific tools (software) are required.

421 The interface for the attack is the ISO interface (contacts of the Security IC) but other interfaces may also be used.

422 The application of appropriate combinations of such methods in order to reveal information (without affecting the TOE's operation or the TOE itself) are addressed by the threat T.Leak-Inherent. Public known attack scenarios are for instance the Simple Power Analysis (SPA) and the Differential Power Analysis (DPA).

423 An attacker may also apply methods in order to cause the TOE to leak information. For instance the attacker must in addition cause faults. The interface for the attack can be more complex in this case. The ISO interface (contacts of the Security IC), the Security IC itself and/or the surface of the integrated circuit may be used to cause faults (refer to the threat T.Malfunction for more detail). Physical manipulations may also be done (refer to the threat T.Phys-Manipulation).

424 The application of appropriate combinations of such methods in order to reveal information (by affecting the TOE's operation or manipulating the TOE itself) are addressed by the threat T.Leak-Forced not being related to attacks on cryptographic algorithms only. Public known attack scenarios are for instance the Differential Fault Analysis (DFA) and the Bellcore type of attacks.

425 The evaluation of the TOE will in many cases not lead to final results for Security IC products built using the TOE. Tests must be repeated with the actual Security IC Embedded Software.

426 Test Features (including other non-application related function) implemented in the TOE might be abused in order to disclose or manipulate user data and bypass, deactivate, change or explore security features or functions of the TOE. Details depend on the capabilities of the Test Features provided by the IC Dedicated Test Software which are not specified here.

427 If the IC Dedicated Test Software offers commands via the ISO I/O interface an attacker needs to communicate with the Security IC using a terminal and the communication software. If other interfaces are used and/or if the usage of such commands is protected, it can be necessary to manipulate the TOE (refer to the threat T.Phys-Manipulation for more detail) and/or to circumvent authentication mechanisms. An attacker may also reveal information by physical probing (refer to the threat T.Phys-Probing) or analysing data (refer to the threats T.Leak-Inherent and T.Leak-Forced). If the TOE provides a command interface it can be subject to manipulations as described under the threat T.Malfunction and the software must not be susceptible to invalid inputs and other types of logical attacks being specific for software. Details depend on the way the Test Features are provided and protected by the TOE which is not specified here.

428 The application of appropriate combinations of methods in order to reveal information or perform manipulations are addressed by the threat T.Abuse-Func.

7.7 Glossary of Vocabulary

Application Data	All data managed by the Security IC Embedded Software in the application context. Application data comprise all data in the final Security IC.
Authentication reference data	Data used to verify the claimed identity in an authentication procedure.
Authentication verification data	Data used to prove the claimed identity in an authentication procedure.
Composite Product Integrator	Role installing or finalising the IC Embedded Software and the applications on platform transforming the TOE into the unpersonalised Composite Product after TOE delivery. The TOE Manufacturer may implement IC Embedded Software delivered by the Security IC Embedded Software Developer before TOE delivery (e.g. if the IC Embedded Software is implemented in ROM or is stored in the non-volatile memory as service provided by the IC Manufacturer or IC Packaging Manufacturer).
Composite Product Manufacturer	The Composite Product Manufacturer has the following roles (i) the Security IC Embedded Software Developer (Phase 1), (ii) the Composite Product Integrator (Phase 5) and (iii) the Personaliser (Phase 6). If the TOE is delivered after Phase 3 in form of wafers or sawn wafers (dice) he has the role of the IC Packaging Manufacturer (Phase 4) in addition. The customer of the TOE Manufacturer who receives the TOE during TOE Delivery. The Composite Product Manufacturer includes the Security IC Embedded Software developer and all roles after TOE Delivery up to Phase 6 (refer to Figure 2 on page 11 and Section 7.1.1).
End-consumer	User of the Composite Product in Phase 7.
IC Dedicated Software	IC proprietary software embedded in a Security IC (also known as IC firmware) and developed by the IC Developer. Such software is required for testing purpose (IC Dedicated Test Software) but may provide additional services to facilitate usage of the hardware and/or to provide additional services (IC Dedicated Support Software).

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

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

User data of the Composite TOE	All data managed by the Smartcard Embedded Software in the application context.
User data of the TOE	Data for the user of the TOE, that does not affect the operation of the TSF. From the point of view of TOE defined in this PP the user data comprises the Security IC Embedded Software and the user data of the Composite TOE.

7.8 Literature

[1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; September 2012, Version 3.1, Revision 4, CCMB-2012-09-001,

[2] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; September 2012, Version 3.1, Revision 4, CCMB-2012-09-002

[3] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; September 2012, Version 3.1, Revision 4, CCMB-2012-09-003

[4] Common Methodology for Information Technology Security Evaluation (CEM), Evaluation Methodology; September 2012, Version 3.1, Revision 4, CCMB-2012-09-004

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

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

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

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

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

[10] Supporting Document: Composite product evaluation for Smart Cards and similar devices, April 2012, Version 2.1, CCDB-2012-04-001

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

[12] Joint Interpretation Library: Minimum Site Security Requirements (For trial use), 2013

[13] Eurosmart Smartcard IC Platform Protection Profile, Version 1.0, July 2007, BSI-PP-0035

[14] Federal Information Processing Standards Publication FIPS PUB 46-3, DATA ENCRYPTION STANDARD (DES), Reaffirmed 1999 October 25, U.S. DEPARTMENT OF COMMERCE/National Institute of Standards and Technology

[15] Federal Information Processing Standards Publication 180-4 SECURE HASH STANDARD U.S. DEPARTMENT OF COMMERCE/National Institute of Standards and Technology, 2011 February, 11

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

[17] Evaluation of random number generators, Version 0.8, Bundesamt für Sicherheit in der Informationstechnik

[18] Killmann, W. Schindler, „A proposal for: Functionality classes for random number generators“, Version 2.0, September 18, 2011

[19] NIST Special Publication 800-90A Recommendation for Random Number Generation Using Deterministic Random Bit Generators, January 2012

[20] NIST DRAFT Special Publication 800-90B Recommendation for the Entropy Sources Used for Random Bit Generation, August 2012

[21] NIST SP 800-67, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, revised January 2012, National Institute of Standards and Technology

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

7.9 List of Abbreviations

CC	Common Criteria.
CCDB	Common Criteria Development Board
EAL	Evaluation Assurance Level.
IC	Integrated circuit.
IT	Information Technology.
JILWG	Joint Interpretation Library Working Group (SOG-IS)
PP	Protection Profile.
RNG	Random number generator
SOG-IS	Senior Officer Group Information Security
ST	Security Target.
TOE	Target of Evaluation.

TSC	TSF Scope of Control.
TSF	TOE Security Functionality.