Public

Common Criteria
Information Technology
Security Evaluation

Security Target Lite of

Samsung S3FW9FV/FT/F9/F8 Secure 32-bit RISC Microcontroller for Smart Card

Version 1.1

29th March 2017

SAMSUNG ELECTRONICS

REVISION HISTORY

UPDATES:

Version	Date	Modification
1.0	27th March 2017	Creation
1.1	29th March 2017	Figure 1 and FDP_SDI.2 is updated

CONTENTS

1 ST INTRODUCTION

2 This introductory chapter contains the following sections:

1.1 Security Target and TOE Reference

1.2 TOE Overview and TOE Description

1.3 Interfaces of the TOE

1.4 TOE Intended Usage

1.1 Security Target and TOE Reference

3 The Security Target Lite version is 1.1 and dated 29th March 2017

4 The Security Target Lite are built on *Common Criteria version 3.1*.

- Title: Security Target Lite of Samsung S3FW9FV/FT/F9/F8 Secure 32-Bit RISC Microcontroller for Smart Cards
- TOE Revision: 0
- Target of Evaluation: S3FW9FV/FT/F9/F8
- Provided by: Samsung Electronics Co., Ltd.
- Common Criteria version :

  [1] Common Criteria, Part 1: Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, Version 3.1, Revision 4, September 2012, CCMB-2012-09-001
  [2] Common Criteria, Part 2: Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, Version 3.1, Revision 4, September 2012, CCMB-2012-09-002
  [3] Common Criteria, Part 3: Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, Version 3.1, Revision 4, September 2012, CCMB-2012-09-003
  [4] Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1, Revision 4, September 2012, CCMB-2012-09-004

1.2 TOE Overview and TOE Description

Introduction

5 The Target of Evaluation (TOE), the S3FW9FV/FT/F9/F8 microcontroller is a smartcard integrated circuit which is composed of a processing unit, security components, contact based I/O ports, hardware circuit for testing purpose during the manufacturing process and volatile and non-volatile memories (hardware). IC embedded software (user software) can be stored in FLASH memory and executed accordingly.

TOE Definition

6 The S3FW9FV/FT/F9/F8 single-chip CMOS micro-controller is designed and packaged specifically for "Smart Card" applications.

7 The SC000 CPU architecture of the S3FW9FV/FT/F9/F8 microcontroller follows the Harvard style, that is, it has separate program memory and data memory. Both instruction and data can be fetched simultaneously without causing a stall, using separate paths for memory access.

8 The main security features of the S3FW9FV/FT/F9/F8 integrated circuit are:

- Security sensors or detectors or filters
- Shields
- Dedicated tamper-resistant design based on synthesizable glue logic and secure topology
- Dedicated hardware mechanisms against side-channel attacks

9 The main hardware blocks of the S3FW9FV/FT/F9/F8 Integrated Circuit are described in **Figure 1** below:

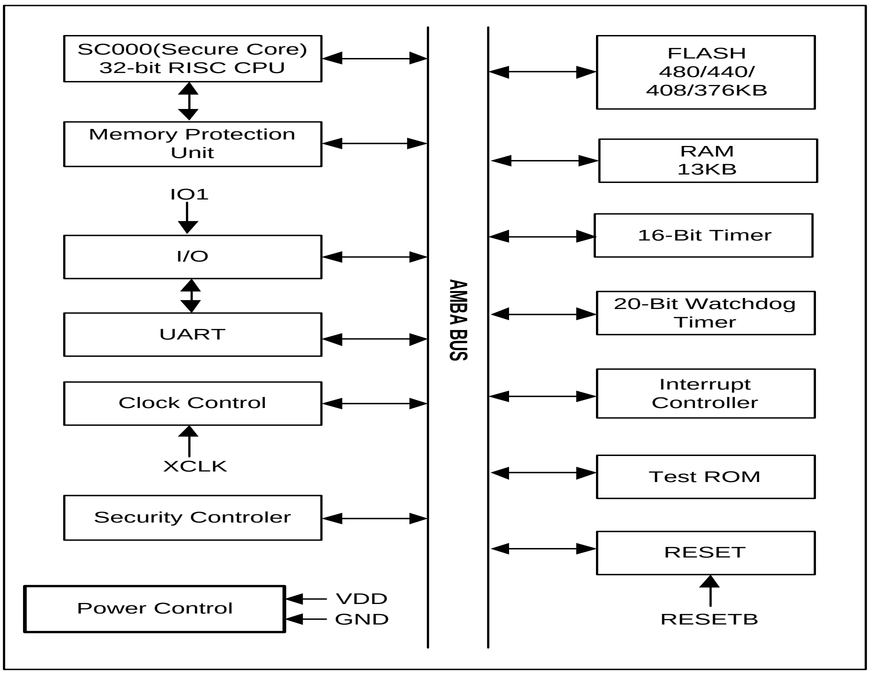


Figure 1. S3FW9FV/FT/F9/F8 Block Diagram
(Red blocks are out of evaluation scope even they are physically exist)

10 The TOE consists of the following Hardware and Software:

TOE Hardware

- 480Kbytes (S3FW9FV), 440Kbytes (S3FW9FT), 408Kbytes (S3FW9F9) and 376Kbytes (S3FW9F8) of FLASH with 13K bytes RAM for all devices in the device family.
- 32-bit Central Processing Unit (CPU)
- Internal Voltage Regulator (IVR)
- Detectors & Security Logic
- Filters
- Hardware UART for contact I/O modes
- Address & data buses
- Internal Clock
- Timers

- Power on Reset

TOE Software

11 The TOE configuration is summarized in table 1 below:

Item Type	Item	Version	Form of delivery
Hardware	S3FW9FV/FT/F9/F8 Secure 32-Bit RISC Microcontroller for Smart Card	0	Wafer
Document	Hardware User's manual	1.0	Softcopy
Document	Security Application Note	1.0	Softcopy
Document	Chip Delivery Specification	1.0	Softcopy

Table 1. TOE Configuration

TOE Features

CPU

- 32-bit SC000 core

Memory

- 480Kbytes (S3FW9FV), 440Kbytes (S3FW9FT), 408Kbytes (S3FW9F9) and 376Kbytes (S3FW9F8) for Data and Program Memory (FLASH)
- 13K-byte Data Memory (RAM)

FLASH Write Operations

Abnormal Condition Detectors

Filters

Interrupts

- Sources for IRQ: I/O1 buffer available
- Source for IRQ: Timer0, IO1 falling edge, Flash erase/write time end interrupts, security
- Software Interrupts

Serial I/O Interface

- T=0 and 1 (ISO 7816-3)

Reset and Power Down Mode

- Power-on reset and external reset
- Stop mode

Memory Encryption and Bus Scrambling

Timers

- 16-Bit Timer with 8 Bit prescaler

Clock Sources

- External clock: 1 MHz–10 MHz(Class A,B) 1MHz-7.5MHz(Class C)
- Internal clock

Operating Voltage Range

- 1.62 V - 5.5 V

Operating Temperature

- - 25°C to 85°C

Package

- Wafer

TOE Life cycle

12 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:

Site / Building	Phase
Hwasung Plant/ DSR Building	Phase 2
Giheung Plant/ Line 6,S1	Phase 3
Onyang Plant	Phase 4
PKL Plant	Phase 3

Table 4. Sites of each phase

13

- IC Development (Phase 2):
    - IC design,
    - IC Dedicated Software development,

- the IC Manufacturing (Phase 3):
    - integration and photomask fabrication,
    - IC production,
    - IC testing,
    - preparation 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.

14 In addition, three 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 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.

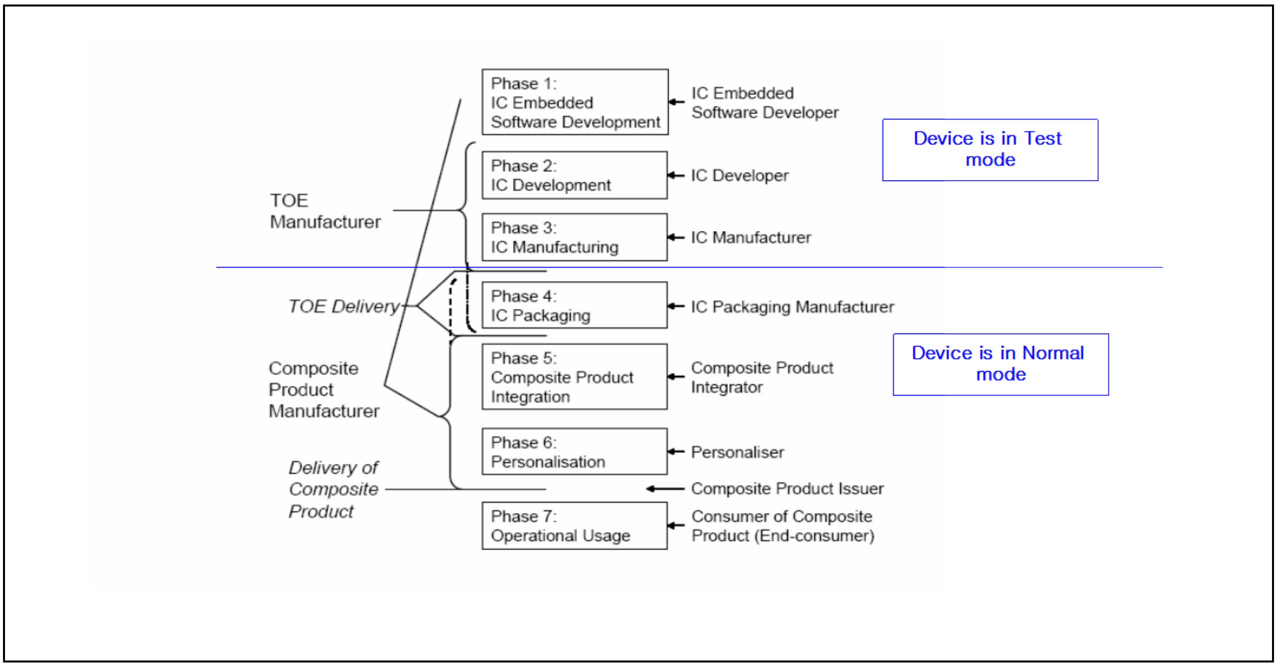


Figure 1: Definition of "TOE Delivery" and responsible Parties

15 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 is delivered in form of wafers. The IC Embedded Software is downloaded at phase 3 and no loader software is included within the TOE in phase 3 or phase 4 since involved engineers download the embedded software with only testing tools.

1.3 Interfaces of the TOE

- The physical interface of the TOE with the external environment is the entire surface of the IC
- The electrical interface of the TOE with the external environment is made of the chip's pads including the Vdd, RESETB, XCLK, GND, IO1.
- The data interface of the TOE is made of the Contact I/O pads.
- The software interface of the TOE with the hardware consists of Special Function Registers (SFR) and CPU instructions.

1.4 TOE Intended Usage

16 The TOE is dedicated to applications such as:

- Network based transaction processing such a mobile phones (GSM SIM cards)

2 CONFORMANCE CLAIMS

17 This chapter 2 contains the following sections:

2.1 CC Conformance Claim

2.2 PP Claim

2.3 Package Claim

2.4 Conformance Claim Rationale

2.1 CC Conformance Claim

18 This Security target claims to be conformant to the Common Criteria version 3.1 R4.

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

20 This *Security Target* has been built with the Common Criteria for Information Technology Security Evaluation; Version 3.1 which comprises

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

[2] Common Criteria, Part 2: Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, Version 3.1, Revision 4, Sept. 2012, CCMB-2012-09-002

[3] Common Criteria, Part 3: Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, Version 3.1, Revision 4, Sept. 2012, CCMB-2012-09-003

[4] Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1, Revision 4, Sept. 2012, CCMB-2012-09-004

21 has been taken into account.

2.2 PP Claim

22 This ST does not claim conformance to any other PP.

2.3 Package Claim

23 The assurance level for this Security Target is EAL4 augmented with AVA_VAN.4 and ALC_DVS.2.

3 SECURITY PROBLEM DEFINITION

24 This chapter 3 contains the following sections:

3.1 Description of Assets

3.2 Threats

3.3 Organizational Security Policies

3.4 Assumptions

3.1 Description of Assets

Assets regarding the Threats

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

- the User Data,
- the Security IC Embedded Software stored and in operation,
- the security services provided by the TOE for the Security IC Embedded Software.

26 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 and 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.

27 The Security IC may not distinguish between User Data which are public known or kept confidential. Therefore the security IC shall protect the confidentiality and integrity of the User Data, unless the Security IC Embedded Software chooses to disclose or modify it.

28 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. Though the Security IC Embedded Software (normally stored in the ROM) will in many cases not contain secret data or algorithms, it must be protected from being disclosed, since for instance knowledge of specific implementation details may assist an attacker.

29 Note that there are many ways to manipulate or disclose the User Data: (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 security functionality. The knowledge of this information enables or supports attacks on the assets. Therefore the TOE Manufacturer must ensure that the development and production of the TOE is secure so that no information is unintentionally made available for the operational phase of the TOE.

30 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.

31 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, Security IC Embedded Software, Initialisation Data and Pre-personalisation Data,
- specific development aids,
- test and characterisation related data,
- material for software development support, and
- photomasks and products in any form

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

3.2 Threats

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

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

33 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.

34 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 secret User Data stored in the TOE, and (iii) develop and build a functional equivalent of the Security IC using the input from the previous steps.

35 The Security IC is a platform for the Security IC Embedded Software which ensures that especially the critical User Data are stored and processed in a secure way (refer to below). The Security IC Embedded Software must also ensure that critical User Data are treated as required in the application context. In addition, the personalisation process supported by the Security IC Embedded Software (and perhaps by the Security IC in addition) must be secure. This last step is beyond the scope of the Security Target. 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 measures which split into those being evaluated according to the Security IC and those being subject to the evaluation of the Security IC Embedded Software or Security IC and the corresponding personalisation process. Therefore, functional cloning is indirectly covered by the security concerns and threats described below.

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

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

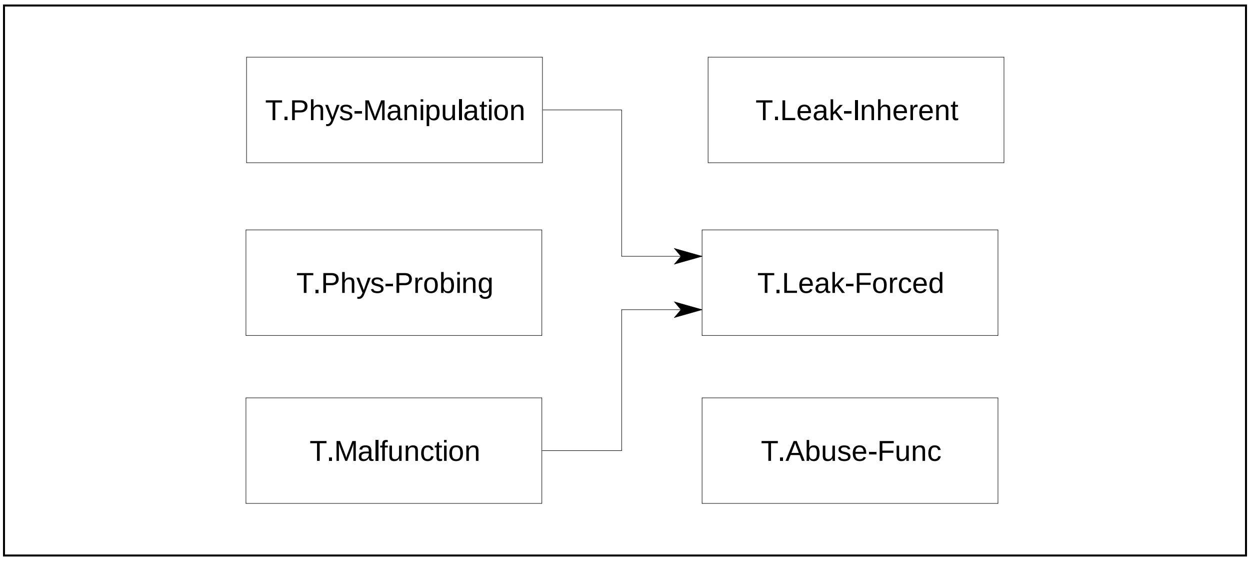


Figure 3: Standard Threats

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

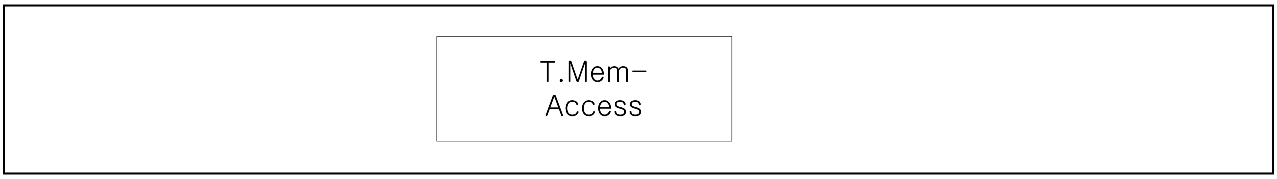


Figure 4: Threats related to security service

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

39 The above security concerns are derived from considering the end-usage phase (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.

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

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

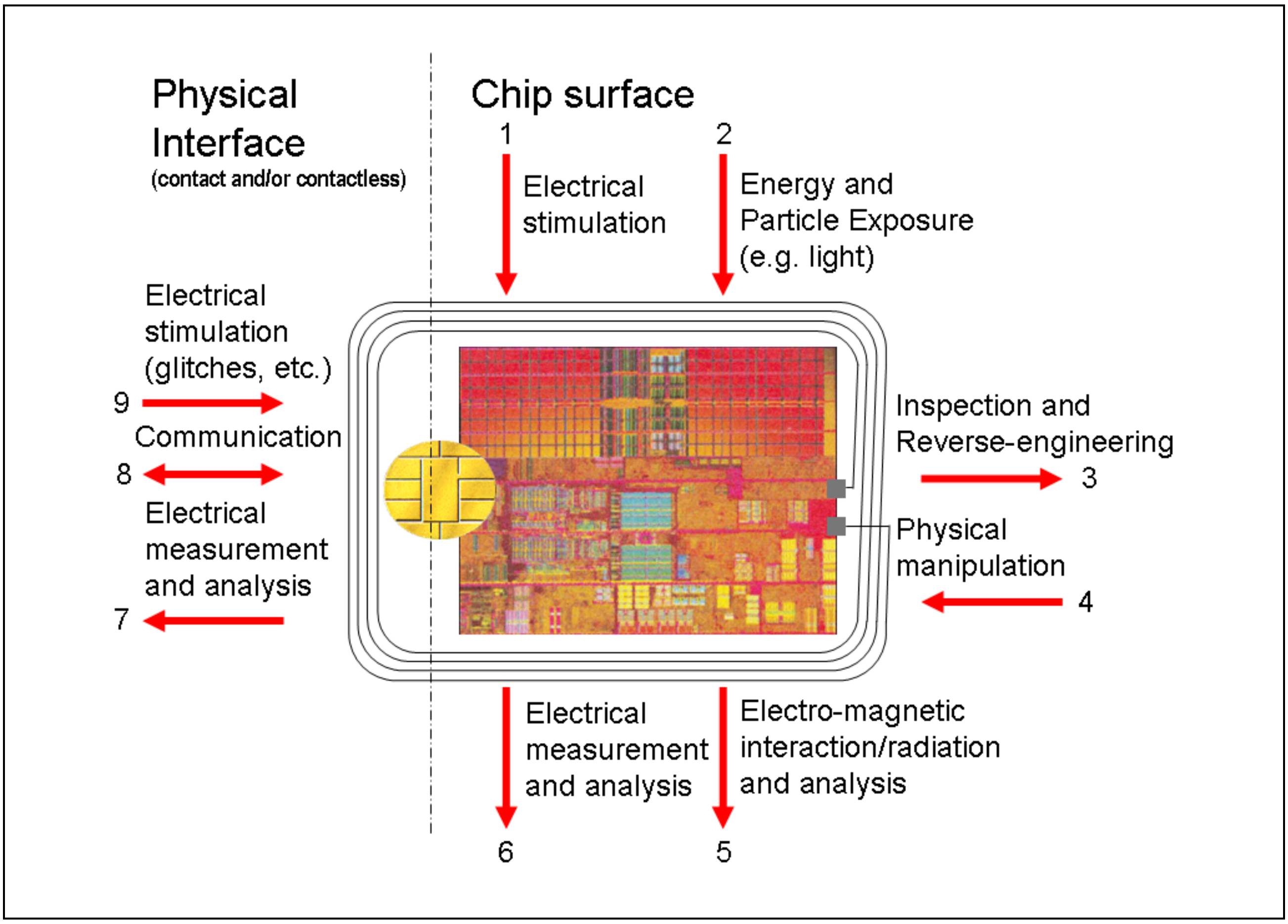

Physical Interface
(contact and/or contactless)
Chip surface
1
Electrical stimulation
2
Energy and Particle Exposure (e.g. light)
Electrical stimulation (glitches, etc.)
9
Communication
8
Electrical measurement and analysis
7
Inspection and Reverse-engineering
3
Physical manipulation
4
Electrical measurement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

Figure 5: Interactions between the TOE and its outer world

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

Standard Threats

43 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 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 the Differential Power Analysis (DPA). This leakage may be interpreted as a covert channel transmission but is more closely related to measurement of operating parameters, which may be derived either from direct (contact) measurements (Numbers 6 and 7 in Figure 5) or measurement of emanations (Number 5 in Figure 5) and can then be related to the specific operation being performed.

44 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, (ii) to disclose/reconstruct the Security IC Embedded Software or (iii) to disclose other critical information about the operation of the TOE to enable attacks disclosing or manipulating the User Data or the Security IC Embedded Software.

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

45 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 or the Security IC Embedded Software. This may be achieved by operating the Security IC outside the normal operating conditions (Numbers 1, 2 and 9 in Figure 5).

46 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, (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 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 5) and IC reverse engineering efforts (Number 3 in Figure 5). The modification may result in the deactivation of a security feature. Before that hardware security mechanisms and layout characteristics need to be identified. Determination of software design including treatment of User Data 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 gathering significant knowledge about the TOE's internal construction here (Number 3 in Figure 5).

47 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 data as part of the assets even if the information leakage is not inherent but caused by the attacker.

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

48 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, (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 or the Security IC Embedded Software..

Threats related to additional TOE Specific Functionality

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

T.Mem-Access Memory Access Violation

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

Clarification: This threat does not address the proper definition and management of the security rules implemented by the Security IC Embedded Software, this being software design and correctness issue. This threat addresses the reliability of the abstract machine targeted by the software implementation. To avert the threat, the set of access rules provided by this TOE should be undefeated if operated according to the provided guidance. The threat is not realized if the Security IC Embedded Software is designed or implemented to grant access to restricted information. It is realized if an implemented access denial is granted under unexpected conditions or if the execution machinery does not effectively control a controlled access.

Here the attacker is expected to (i) take advantage of flaws in the design and/or the implementation of the TOE memory access rules (refer to T.Abuse-Func but for functions available after TOE delivery), (ii) introduce flaws by forcing operational conditions (refer to T.Malfunction) and/or by physical manipulation (refer to T.Phys-Manipulation). This attacker is expected to have a high level potential of attack.

3.3 Organizational Security Policies

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

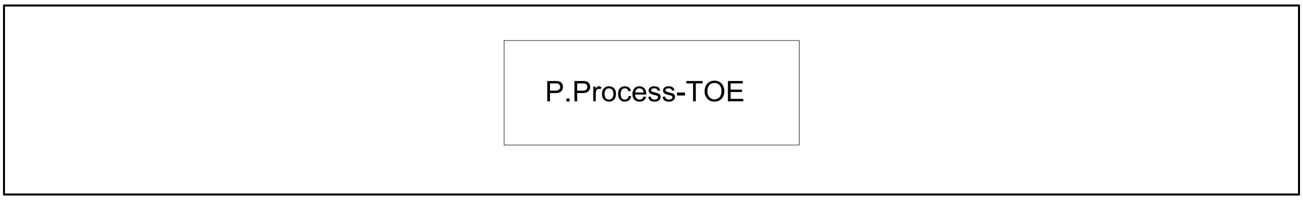


Figure 6: Policies

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

P.Process-TOE	Protection 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.

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

53 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, Security IC Embedded Software, Initialisation Data and Pre-personalisation Data,
- 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.

54 The TOE provides specific security functionality which can be used by the Smartcard Embedded Software. In the following specific security functionality is listed which is not derived from threats identified for the TOE's environment because it can only be decided in the context of the smartcard application, against which threats the Smartcard Embedded Software will use the specific security functionality.

3.4 Assumptions

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

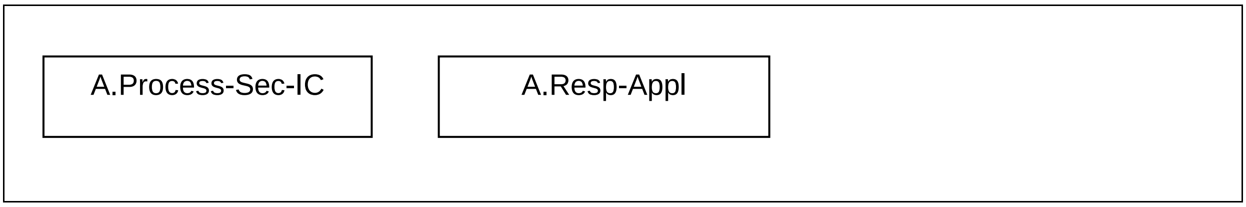


Figure 7: Assumptions

56 The intended usage of the TOE is twofold, depending on the Life Cycle Phase: (i) The Security IC Embedded Software developer use 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.

57 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.

58 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 consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use).

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

59 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 and personalisation data including specifications of formats and memory areas, test related data,
- the User Data and related documentation, and
- material for software development support

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

60 The developer of the Security IC Embedded Software must ensure the appropriate "Treatment of User Data (A.Resp-Appl)" while developing this software in Phase 1 as specified below.

A.Resp-Appl Treatment of User Data

All User Data are owned by Security IC Embedded Software. Therefore, it must be assumed that security relevant User Data (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 shall be handled and protected. The evaluation of the Security IC according to this Security Target is conducted on generalized application context. The concrete requirements for the Security IC Embedded Software shall be defined in the Security Target for the Security IC Embedded Software. The Security IC can not prevent any compromise or modification of User Data by malicious Security IC Embedded Software. The assumption A.Resp-Appl ensures that the Security IC Embedded Software follows the security rules of the application context.

4 SECURITY OBJECTIVES

61 This chapter Security Objectives contains the following sections:

4.1 Security Objectives for the TOE

4.2 Security Objectives for the IC Embedded Software development Environment

4.3 Security Objectives for the operational Environment

4.4 Security Objectives Rationale

4.1 Security Objectives for the TOE

62 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.

63 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.

64 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 8). Note that the integrity of the TOE is a means to reach these objectives.

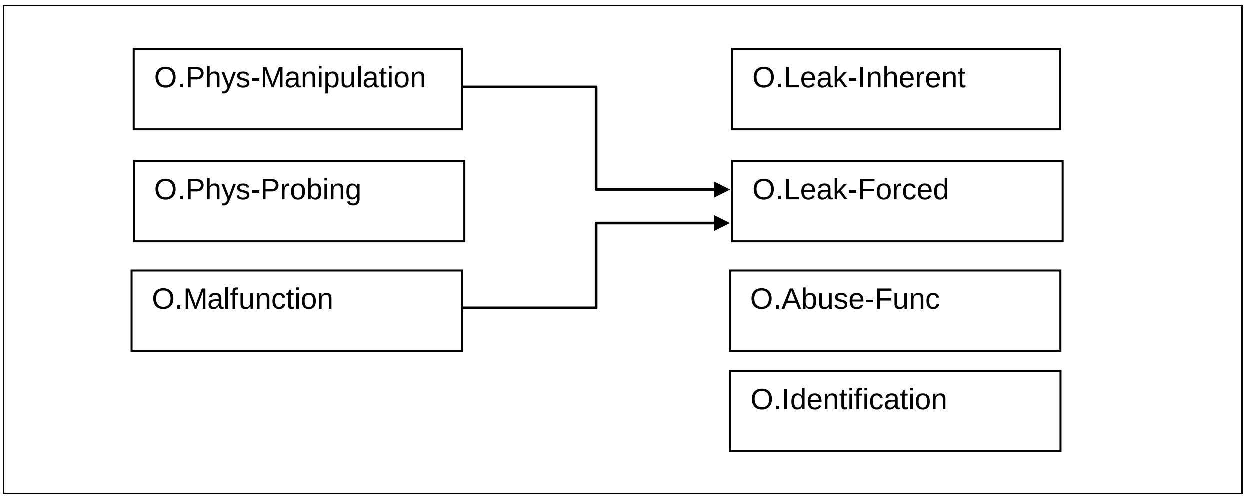


Figure 8: Standard Security Objectives

65 The additional high-level security considerations are refined below by defining security objectives as required by the Common Criteria (refer to Figure 9).

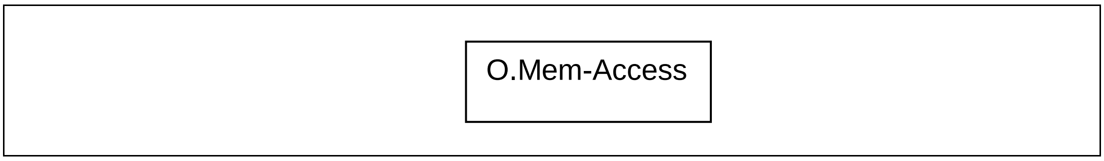


Figure 9: Security Objectives related to Specific Functionality

Standard Security Objectives

66 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 (User Data or TSF data) stored and/or processed in the Smartcard IC

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

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

67 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 of User Data, against the disclosure/reconstruction of the Smartcard Embedded Software or against the disclosure of other critical operational information. This includes protection against

- measuring through galvanic contacts which is direct physical probing on the chips surface except on pads being bonded (using standard tools for 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.

68 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 prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent errors. The environmental conditions may include voltage, clock frequency, temperature, or external energy fields.

Remark: A malfunction of the TOE 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.

69 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 Smartcard Embedded Software and the User Data. This includes protection against

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

The TOE 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.

70 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.

71 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 (i) to disclose critical User Data, (ii) to manipulate critical User Data of the Smartcard Embedded Software,

(iii) to manipulate Soft-coded Smartcard Embedded Software or (iv) bypass, deactivate, change or explore security features or functions 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.

72 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 for Memory Access Control

73 The TOE shall provide "Area based Memory Access Control (O.Mem-Access)" as specified below.

O.Mem-Access Area based Memory Access Control

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

4.2 Security Objectives for the Security IC Embedded Software

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

75 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.

76 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 Smartcard Embedded Software will not disclose security relevant user data 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

77 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 "consumer" to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use).

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

4.4 Security Objectives Rationale

78 Table 5 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	Phase 1
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.Mem-Access	O.Mem-Access	

Table 5: Security Objectives versus Assumptions, Threats or Policies

79 The justification related to the assumption "Treatment of User Data (A.Resp-Appl)" is as follows:

80 Since OE.Resp-Appl requires the developer of the Smartcard Embedded Software to implement measures as assumed in A.Resp-Appl, the assumption is covered by the objective.

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

82 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 41 (page14). 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.

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

84 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.

85 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)" is as follows:

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

87 The justification related to the threat "Memory Access Violation (T.Mem-Access)" is as follows:

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

89 The clarification of O.Mem-Access makes clear that it is up to the Smartcard Embedded Software to implement the memory management scheme by appropriately administrating the TSF. The TOE shall provide access control functions as a means to be used by the Smartcard Embedded Software. This is further emphasised by the clarification of Treatment of User Data (OE.Resp-Appl) which reminds that the Smartcard Embedded Software must not undermine the restrictions it defines. Therefore, the clarifications contribute to the coverage of the threat T.Mem-Access.

5 EXTENDED COMPONENTS DEFINITION

90 This chapter 5 Extended Components Definition contains the following sections:

5.1 Definition of the Family FMT_LIM

5.2 Definition of the Family FAU_SAS

5.3 Definition of the Family FDP_SDC

5.1 Definition of the Family FMT_LIM

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

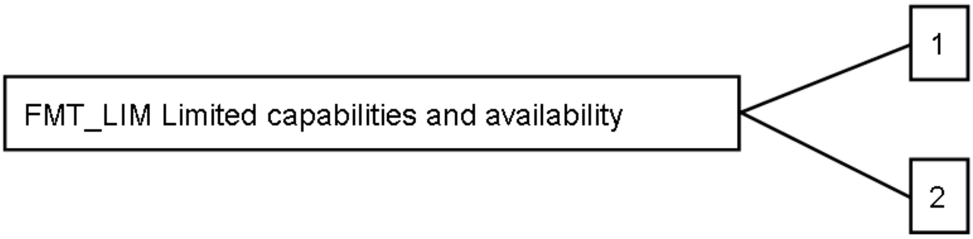
92 The family "Limited capabilities and availability (FMT_LIM)" is specified as follows.

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.

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

FMT_LIM.1 Limited capabilities

Hierarchical to: No other components.

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 [assignment: Limited capability and availability policy].

Dependencies: FMT_LIM.2 Limited availability.

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

FMT_LIM.2 Limited availability

Hierarchical to: No other components.

FMT_LIM.2.1 The TSF shall be designed in a manner that limits their availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced [assignment: Limited capability and availability policy].

Dependencies: FMT_LIM.1 Limited capabilities.

95 Application note: The functional requirements FMT_LIM.1 and FMT_LIM.2 assume that there are two types of mechanisms (limited capabilities and limited availability) which together shall provide protection in order to enforce the policy. This also allows 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.

The combination of both requirements shall enforce the policy.

5.2 Definition of the Family FAU_SAS

96 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.

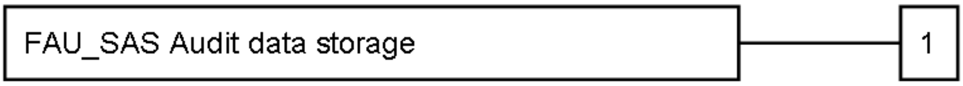
97 The family "Audit data storage (FAU_SAS)" is specified as follows.

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.

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].

Dependencies: No dependencies.

5.3 Definition of the Family FDP_SDC

98 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.

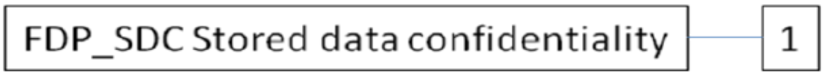
99 The family " Stored data confidentiality (FDP_SDC)" is specified as follows.

FAU_SDC.1 **Stored data confidentiality**

Family behaviour

100 This family defines 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*]

Application note: Evaluator may assess the FLASH/RAM 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.

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)".

6 IT SECURITY REQUIREMENTS

101 This chapter 6 IT Security Requirements contains the following sections:

6.1 Security Functional Requirements for the TOE

6.2 Security Assurance Requirements for the TOE

6.3 Security Requirements Rationale

6.1 Security Functional Requirements for the TOE

102 In order to define the Security Functional Requirements the Part 2 of the Common Criteria was used. However, some Security Functional Requirements have been refined. The refinements are described below the associated SFR. The operations completed in the ST are marked in italic font.

103 Please note that, the following conventions are used to state each Security Functional Requirement:

- Refinement operations are explicitly identified at the end of the SFR definition.
- Assignment operations are identified *italic*.
- Selection operations are identified by underline.

Malfunctions

104 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

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).*

Dependencies: FPT_FLS.1 Failure with preservation of secure state

Refinement: The term "failure" above means "circumstances". The TOE prevents failures for the "circumstances" defined above.

105 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.

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.*

Dependencies: No dependencies

Refinement: The term "failure" above also covers "circumstances". The TOE prevents failures for the "circumstances" defined above.

Application note: The secure state is maintained by TOE's detectors. The TOE's detectors are monitoring the failure occurs. *The failures are abnormal detectors that detect out of the specified range. If the failures are happen, the TOE goes into secure state. This satisfies the FPT_FLS.1 "Failure with preservation of secure state."*

Abuse of Functionality

106 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.

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

Dependencies: FMT_LIM.2 Limited availability.

107 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.

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

Dependencies: FMT_LIM.1 Limited capabilities.

108 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.

FAU_SAS.1.1 The TSF shall provide *the test process before TOE Delivery* with the capability to store the *Initialisation Data and/or Pre-personalisation Data and/or supplements of the Smartcard Embedded Software* in the *Test ROM area.*

Dependencies: No dependencies.

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

Physical Manipulation and Probing

109 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 *FLASH or RAM.*

Application note: Evaluator may assess the FLASH/RAM 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.

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)".

110 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: No other components.

Dependencies: No dependencies.

FDP.SDI.2.1 The TSF shall monitor user data stored in containers controlled by the TSF for *CRC* on all objects, based on the following attributes: *FLASH or RAM read operation*.

FDP.SDI.2.2 Upon detection of a data integrity error, the TSF shall *enforce adevice RESET or an interrupt (FIQ- Fast Interrupt)*

Application Note: This requirement is achieved by Security IC Embedded Software using CRC. Security IC Embedded Software should implement monitoring procedures for User Data stored on memory

111 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.

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

Dependencies: No dependencies.

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: This requirement is achieved by security feature as the Active shield must be removed and bypassed in order to perform physical intrusive attacks. The TOE makes appropriate secure reaction to stop operation if a physical manipulation or physical probing attack is detected. And also internal scrambling & encryption for memory and logic area make the reverse-engineering of the TOE layout unpractical. So these functionalities meet the security functional requirement of FPT_PHP.3: Resistance to physical attack.Leakage.

Leakage

112 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.

FDP_ITT.1.1 The TSF shall enforce the *Data Processing Policy* to prevent the disclosure of user data when it is transmitted between physically-separated parts of the TOE.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]

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.

113 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.

FPT_ITT.1.1 The TSF shall protect TSF data from disclosure when it is transmitted between separate parts of the TOE.

Dependencies: No dependencies.

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.

114 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.

FDP_IFC.1.1 The TSF shall enforce the *Data Processing Policy* on *all confidential data when they are processed or transferred by the TOE or by the Security IC Embedded Software.*

Dependencies: FDP_IFF.1 Simple security attributes

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

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

Memory access control

116 Usage of multiple applications in one Smartcard often requires separating code and data in order to prevent that one application can access code of another application. To support this, the TOE provides Area based Memory Access Control.

117 The security service being provided is described in the Security Function Policy (SFP) **Memory Access Control Policy**. The security functional requirement **"Subset access control (FDP_ACC.1)"** requires that this policy is in place and defines the scope were it applies. The security functional requirement **"Security attribute based access control (FDP_ACF.1)"** defines security attribute usage and characteristics of policies. It describes the rules for the function that implements the Security Function Policy (SFP) as identified in FDP_ACC.1. The decision whether an access is permitted or not is taken based upon attributes allocated to the software. The user software defines the attributes. The corresponding permission control information is evaluated "on-the-fly" by the hardware so that access is granted/effective or denied/inoperable.

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

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

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

Memory Access Control Policy

The TOE shall control *the execution of code stored in memory area.*

The TOE shall restrict the ability to define, to change or at least to finally accept the applied rules (as mentioned in FDP_ACF.1) to *software with memory area where the software is executed.*

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

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

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

FDP_ACF.1 Security attribute based access control

The attributes are the operations related to the code stored in memory, which are the *execute* operations.

Hierarchical to: No other components.

FDP_ACF.1.1 The TSF shall enforce the *Memory Access Control Policy* to objects based on *the* memory *area where the software is executed from.*

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *evaluate the corresponding permission control information before code executions so that executions to be denied cannot be utilised by the subject attempting to perform the operation*.

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

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialisation

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

FMT_MSA.3 Static attribute initialisation

Hierarchical to: No other components.

FMT_MSA.3.1 The TSF shall enforce the *Memory Access Control Policy* to provide *well defined (refer to S3FW9FV/FT/F9/F8 User's Manual – MASCON)* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow *access to RAM* to specify alternative initial values to override the default values when an object or information is created.

Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

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

FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.

FMT_MSA.1.1 The TSF shall enforce the *Memory Access Control Policy* to restrict the ability to modify the security attributes *permission control information* to *running at privilege MASCON (refer to S3FW9FV/FT/F9/F8 User's Manual – MASCON).*

Dependencies: [FDP_ACC.1 Subset access control or
FDP_IFC.1 Subset information flow control]
FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

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

FMT_SMF.1 Specification of management functions

Hierarchical to: No other components

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions*: access the control registers of the MASCON (refer to S3FW9FV/FT/F9/F8 User's Manual – MASCON).*

Dependencies: No dependencies

Summary of Security Functional Requirements

Security Functional Requirements
Limited fault tolerance (FRU_FLT.2)
Failure with preservation of secure state (FPT_FLS.1)
Audit storage (FAU_SAS.1)
Stored data confidentiality (FDP_SDC.1)
Stored data integrity monitoring and action (FDP_SDI.2)
Limited capabilities(FMT_LIM.1)
Limited availability (FMT_LIM.2)
Resistance to physical attack (FPT_PHP.3)
Basic internal transfer protection (FDP_ITT.1)
Basic internal TSF data transfer protection (FPT_ITT.1)
Subset information flow control (FDP_IFC.1)

Table 6. Security Functional Requirements

Security Functional Requirements
Subset access control (FDP_ACC.1)
Security attribute based access control (FDP_ACF.1)
Static attribute initialization (FMT_MSA.3)
Management of security attributes (FMT_MSA.1)
Specification of management functions (FMT_SMF.1)

Table 7. Augmented Security Functional Requirements

6.2 TOE Assurance Requirements

126 The Security Target will be evaluated according to

Security Target evaluation (Class ASE)

127 The TOE Assurance Requirements for the evaluation of the TOE and its development and operating environment are those taken from the

Evaluation Assurance Level 4 (EAL4)

and augmented by the following components

ALC_DVS.2 and AVA_VAN.4

128 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 activities		
	Operational User Guidance	(AGD_OPE.1)
	Preparative procedures	(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.1)
	Functional Tests	(ATE_FUN.1)
	Independent Testing	(ATE_IND.2)
Class AVA: Vulnerability assessment		
	Vulnerability Analysis	(AVA_VAN.4)

6.2.1 Refinements of the TOE Assurance Requirements

129 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.

130 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)

131 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

132 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.

133 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":

134 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

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

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

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

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

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

Application Note : Depending on whether the TOE comprises programmable non-volatile memory 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

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

137 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.

138 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 samples17, 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.

139 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:

140 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

141 **" 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 : 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

142 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 Refinements regarding CM scope (ALC_CMS)

Introduction

143 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.

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

145 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

146 **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

147 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. 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.

148 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 Refinements regarding CM capabilities (ALC_CMC)

Introduction

149 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.

150 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:

151 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

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

153 **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

154 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.

155 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

156 **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.**

157 **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

158 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.

159 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.

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

161 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

162 **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.**

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

164 **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.**

165 **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

166 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

167 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.

168 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."

169 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

170 **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 Refinements regarding Test Coverage (ATE_COV)

Introduction

171 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."

172 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 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

173 **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).**

174 **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

175 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.

6.2.1.7 Refinements regarding Implementation Representation (ADV_IMP)

Introduction

176 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.

177 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."

178 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

179 **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.**

6.2.1.9 Refinements regarding User Guidance (AGD_OPE)

Introduction

180 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.“

181 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.

182 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.

183 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 fulfill 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

184 **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.**

185 **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.**

186 **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

187 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 Refinements regarding Preparative User Guidance (AGD_PRE)

Introduction

188 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.

189 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

190 **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.**

191 **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.**

192 **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.**

6.2.1.10 Refinements regarding Vulnerability Analysis (AVA_VAN).

Introduction

193 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.”

194 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.

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

Developer action elements:

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

Content and presentation elements:

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

Evaluator action elements:

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

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

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

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

Refinement

196 **The vulnerability analysis shall include a justification for the rating of information on the TOE available to the attacker and the usage of Open Samples since the protection of such information is demanded according to refinement regarding "Development Security (ALC_DVS)", section 6.2.1.2.**

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

Application Note: The vulnerability analysis will assess that the functions provided by the IC Dedicated Test Software cannot be abused after TOE Delivery (refer to the security functional requirements FMT_LIM.1 and FMT_LIM.2 in section 6.1). 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

Rationale for the Security Functional Requirements

197 Table 8 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" - AVA_VAN.4 "Advanced methodical vulnerability analysis"
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" - ADV_ARC.1 "Architectural Design with domain separation and non-bypassability"
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, AVA_VAN.4 plus those listed for O.Malfunction and O.Phys-Manipulation - FRU_FLT.2, FPT_FLS.1, FPT_PHP.3, ADV_ARC.1
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, ADV_ARC.1
O.Identification	- FAU_SAS.1 "Audit storage"
OE.Resp-Appl	not applicable
OE.Process-Sec-IC	not applicable
O.Mem-Access	- FDP_ACC.1 "Subset access control" - FDP_ACF.1 "Security attribute based access control" - FMT_MSA.3 "Static attribute initialisation" - FMT_MSA.1 "Management of security attributes" - FMT_SMF.1 "Specification of Management Functions"

Table 8: Security Requirements versus Security Objectives

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

199 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 behavior of the TOE while data are transmitted between or processed by TOE parts.

200 Of course this has also to be supported by the Security IC Embedded Software. For example timing attacks were possible if the processing time of algorithms implemented in the software would depend on the content of secret variables.

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

202 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.

203 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). In this case the combination of the Security IC Embedded Software together with FPT_PHP.3 is suitable to meet the objective.

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

205 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. To support this, the functions implementing FRU_FLT.2 and FPT_FLS.1 must work independently so that their operation cannot affect by the Security IC Embedded Software (refer to the refinement). Therefore, there is no possible instance of conditions under O.Malfunction, which is not covered. The suitability of the implementation is subject of the evaluation of the assurance component ADV_ARC.1

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

207 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.

208 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). This support must be addressed in the Guidance Documentation. Together with this FPT_PHP.3 is suitable to meet the objective.

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

210 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 measures 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.

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

212 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.

213 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 7.

214 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.

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

216 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.

217 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.

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

219 The security functional requirement "Subset access control (FDP_ACC.1)" with the related Security Function Policy (SFP) "Memory Access Control Policy" exactly require the implementation of an area based memory access control, which is a requirement from O.Mem-Access. Therefore, FDP_ACC.1 with its SFP is suitable to meet the security objective.

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

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

222 Finally, the security functional requirement "Specification of Management Functions (FMT_SMF.1)" is used for the specification of the management functions to be provided by the TOE as required by O.MEM_ACCESS. Therefore, FMT_SMF.1 is suitable to meet the security objective O.Mem_Access.

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

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

Dependencies of Security Functional Requirements

225 Table 8 below lists the security functional requirements defined in this Security Target, their dependencies and whether they are satisfied by other security requirements defined in this Security Target. The text following the table discusses the remaining cases.

Security Functional Requirement	Dependencies	Fulfilled by security requirements
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
FDP_SDC.1	None	No dependency
FDP_SDI.2	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_ACC.1	FDP_ACF.1	Yes
FDP_ACF.1	FDP_ACC.1 FMT_MSA.3	Yes Yes
FMT_MSA.3	FMT_MSA.1 FMT_SMR.1	Yes See discussion below
FMT_MSA.1	FDP_ACC.1 or FDP_IFC.1 FMT_SMR.1 FMT_SMF.1	Yes See discussion below Yes
FMT_SMF.1	None	No dependency

Table 9: Dependencies of the Security Functional Requirements

226 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). Therefore the dependency is considered satisfied.

227 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 because they prevent an attacker from disabling or circumventing the latter. Together with the discussion of the dependencies above this shows that the security functional requirements build a mutually supportive whole.

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

Rationale for the Assurance Requirements

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

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

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

ALC_DVS.2 Sufficiency of Security Measures

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

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

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

AVA_VAN.4 Advanced Methodical Vulnerability Analysis

235 Due to the intended use of the TOE, it is intended to defend against attacks with moderate level of resistance. This assurance requirement is achieved by the AVA_VAN.4 component.

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

237 AVA_VAN.4 has dependencies to ADV_ARC.1 “Security Architectural Design”, ADV_FSP.4 “Complete functional specification”, ADV_TDS.3 “Basic modular design”, ADV_IMP.1 “Implementation representation of the TSF”, AGD_OPE.1 “Operational user guidance”, AGD_PRE.1 “Preparative procedures”, and ATE_DPT.1 “Testing: security enforcing modules”.

238 All these dependencies are satisfied by EAL4.

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

Security Requirements are Internally Consistent

240 The discussion of security functional requirements and assurance components in the preceding sections has shown that mutual support and 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.

241 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.

242 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. 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 and those implemented in the Security IC Embedded Software.

243 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. 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 and those implemented in the Security IC Embedded Software.

244 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 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).

245 Physical probing (refer to FPT_PHP.3) shall directly avert the disclosure of primary assets. 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.

246 Leakage (refer to FDP_ITT.1, FPT_ITT.1) shall directly avert the disclosure of primary assets. 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.

247 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.

248 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, (ii) to manipulate (explore, bypass, deactivate or change) security features or functions of the TOE or of the Security IC Embedded Software or (iii) to enable an attack. Hereby the binding between these two security functional requirements is very important:

249 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 function which limits the availability can be bypassed, deactivated or changed by physical manipulation or a malfunction caused by an attacker. Therefore, if Limited Availability (FMT_LIM.2) is vulnerable, it is important to limit the capabilities of the functions in order to limit the possible benefit for an attacker.

250 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, to manipulate security features or functions of the TOE or of the Security IC Embedded Software or to enable an attack. Therefore, if an attacker could benefit from using such functions, it is important to limit their availability so that an attacker is not able to use them.

251 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.

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

253 Parts of the Smartcard IC Embedded Software may cause security violations by accidentally or deliberately accessing restricted data (which may include code). In order to avert the memory access violation it is important to the security functional requirement defining the scope where the Memory Access Policy is applied (FDP_ACC.1) and the security functional requirement defining the Memory Access Policy(FDP_ACF.1), and the security functional requirement ensuring the default value of security attribute(FMT_MSA.3) and the security functional requirement managing security attribute (FMT_MSA.1) and the security functional requirement performing security management function(FMT_SMF.1) are effective and bind well.

7 TOE SUMMARY SPECIFICATION

254 This chapter 7 TOE Summary Specification contains the following sections:

7.1 List of Security Functional Requirements

7.1 List of Security Functional Requirements

SFR1: FPT_FLS.1: Failure with preservation of secure state

255 The detection thresholds of **TOE's detectors** are inside the operating range of the TOE. Therefore abnormal events/failures are detected before the secure state is compromised. This allows to take User's defined appropriate actions by software or to immediately RESET the TOE.

256 The secure state is maintained by TOE's detectors. The TOE's detectors are monitoring the failure occurs. The failures are happen, the TOE goes into RESET state. This satisfies the FPT_FLS.1 "Failure with preservation of secure state."

TOE's Detectors

257 These functions records in register the events notified by the detectors (refer to list below). The software configures the reaction in case of detection:

- The TOE is immediately reset when an event is detected.
- Or, a special function register bit is set.

SFR2: FRU_FLT.2: Limited fault tolerance

258 All operating signals are filtered/regulated in order to prevent malfunction.

TOE's Filters

259 These filters are used for preventing noise, glitches and extremely high frequency in pad from causing undefined or unpredictable behavior of the chip.

260 TOE's filters and detectors are implemented by the hardware. The filtering and detection cannot be affected or bypassed by Smartcard Embedded Software. The reaction to the detection can be configured by the software. The influence on security and the way how to configure it is described in details in the S3FW9FV/FT/F9/F8 *User's Manual*. Therefore, FRU_FLT.2 is implemented by TOE.

261 Security domains are maintained since accesses to the access-prohibited area are trapped by this access control function.

SFR3: FPT_PHP.3: Resistance to physical attacks

262 This requirement is achieved by security feature as the shield must be removed and bypassed in order to perform physical intrusive attacks. The TOE makes appropriate secure reaction to stops operation if a physical manipulation or physical probing attack is detected. And also scrambling and mechanisms make reverse-engineering of the TOE layout unpractical. So these functionalities meet the security functional requirement of FPT_PHP.3: Resistance to physical attack.

SFR4: FDP_ACC.1: Subset access control

263 This requirement is achieved by security register access control, access right for the code executed in RAM.

1) Security registers access control: This security function manages access to the security control registers through access control security attributes.

2) Access rights for the code executed in RAM

3) Access control for operating state: This security function select booting memory area. User can select FLASH-Boot or FLASH bootloader Boot.

SFR5: FDP_ACF.1: Security attributes based access control.

264 This is covered by the policy defined for FDP_ACC.1, specifically (Access rights for the code executed in RAM).

SFR6: FMT_MSA.3: Static attribute initialization.

265 All Special Function Registers have DEFAULT values after Power on Reset..

SFR7: FMT_MSA.1: Management of security attributes.

266 This is achieved with the MASCON feature. The MASCON enables user to execute code and set register access.

SFR8: FMT_SMF.1: Specification of management functions.

267 This is achieved via access to Special Function Registers.

SFR9: FAU_SAS.1: Audit Storage

268 This requirement is fulfilled by the traceability/identification data written once and for all during the TEST mode of the manufacturing process.

269 During the TEST mode of manufacturing process, traceability data are written in the non-volatile memory of the TOE. Once the TOE is switched from TEST to NORMAL mode, those traceability data are READ ONLY and cannot be modified anymore. This enables to identify and track the TOE during the rest of its life.

SFR10: FMT_LIM.1: Limited capabilities

270 TEST mode can be accessed only by the TEST administrator by supplying an authentication password through a proprietary protocol. Once the TOE is changed to NORMAL mode, TEST mode functions are no more available for NORMAL mode.

SFR11: FMT_LIM.2: Limited availabilities

271 TEST mode can be accessed only by the TEST administrator by supplying an authentication password through a proprietary protocol. Once the TOE is changed to NORMAL mode, TEST mode commands are no more available for NORMAL mode. Functional test during manufacturing process is only available for TEST mode only.

SFR12: FDP_IFC.1: Subset information flow control

272 This requirement is achieved by the function protects the memory contents of the TOE from data analysis on the stored data as well as on internally transmitted data.

SFR13: FDP_ITT.1: Basic internal transfer protection

273 This requirement is achieved by the combination of the TOE security features TOE features 1) to 5) as it is unpractical to get access to internal signals and interpret them.

1) Static Address/Data scrambling for bus and memory: This function protects memory and address/data bus from probing attacks.

2) Memory encryption: This security function protects the memory contents of the TOE from data analysis on the stored data as well as on internally transmitted data.

3) Synthesizable processor core: The Central Processing Unit (CPU) of the TOE is synthesizable with glue logic, which makes reverse engineering and signal identification more difficult.

4) De-synchronization : The TOE operations can be made asynchronous. They make a full range of intrusive (e.g. probing attacks) and non-intrusive attacks (e.g. side-channel attacks) more complex and difficult.

SFR14: FPT_ITT.1: Basic internal TSF data transfer protection

274 This requirement is achieved by the combination of the TOE security features TOE features 1) to 4) as it is unpractical to get access to internal signals and interpret them.

SFR15: Stored data confidentiality (FDP_SDC.1)

275 This requirement is achieved by the combination of the TOE security features TOE features 1) to 3) as it is unpractical to get access to internal signals and interpret them.

1) Static Address/Data scrambling for bus and memory: This function protects memory and address/data bus

2) Data encryption for bus: This function protects data bus

2) Memory encryption: This security function protects the memory contents of the TOE from data analysis on the stored data.

SFR16: Stored data integrity monitoring and action (FDP_SDI.2)

276 This requirement is achieved by following functions.

1) CRC (Cyclic Redundancy Check)

8 ANNEX

8.1 Glossary

Application Data

All data managed by the Security IC Embedded Software in the application context. Application data comprise all data in the final Security IC.

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.

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).

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.

Security IC

Composition of the TOE, the Security IC Embedded Software, User Data 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

TOE Delivery

The period when the TOE is delivered which is 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 and its development and production environment are fulfilled. 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 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 specific circuitry, in non-volatile programmable memories (for instance E2PROM) or a combination thereof.

User data

All data managed by the Smartcard Embedded Software in the application context. User data comprise all data in the final Smartcard IC except the TSF data.

8.2 Abbreviations

CC

Common Criteria

EAL

Evaluation Assurance Level

IT

Information Technology

PP

Protection Profile

ST

Security Target

TOE

Target of Evaluation

TSC

TSF Scope of Control

TSF

TOE Security Functionality

TSFI

TSF Interface

TSP

TOE Security Policy

8.3 Literature

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

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

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

[4] Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1, Revision 4, September 2012, 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